

Metadata and Standard Programs

Marianne Caramés, Novo Nordisk A/S, Bagsværd, Denmark
Martin Lindhard, Novo Nordisk A/S, Bagsværd, Denmark

ABSTRACT

Novo Nordisk A/S is using a concept of metadata based on an enhanced CDISC SDTM model. The metadata are structured in a database and are used by the statistical programmers and statisticians to define the planned measurements for a trial, and in the end for producing the tables, listings and figures. A set of new standard programs is making use of these metadata and is ensuring a more intelligent and structured way of statistical reporting which will be a great benefit for later submissions where you typically pool data from a number of trials and make an integrated summary.

INTRODUCTION

Why just use the collected clinical data in your reporting, when the use of different types of metadata can make your data reporting both more intelligent and data-driven?

DATA IS NOT LIMITED TO CLINICAL DATA

The collection of data should not be limited to the clinical data coming from the CRFs. By using the different sources of metadata: from protocols, programming specifications and defined business rules, we can add an extra layer of data to our clinical data and make room for a more generalised and data-driven approach for analysing and reporting of the data. The generalised approach will facilitate the use of standard programs for the reporting.

In Figure 1 the metadata is divided into different categories in the data warehouse.

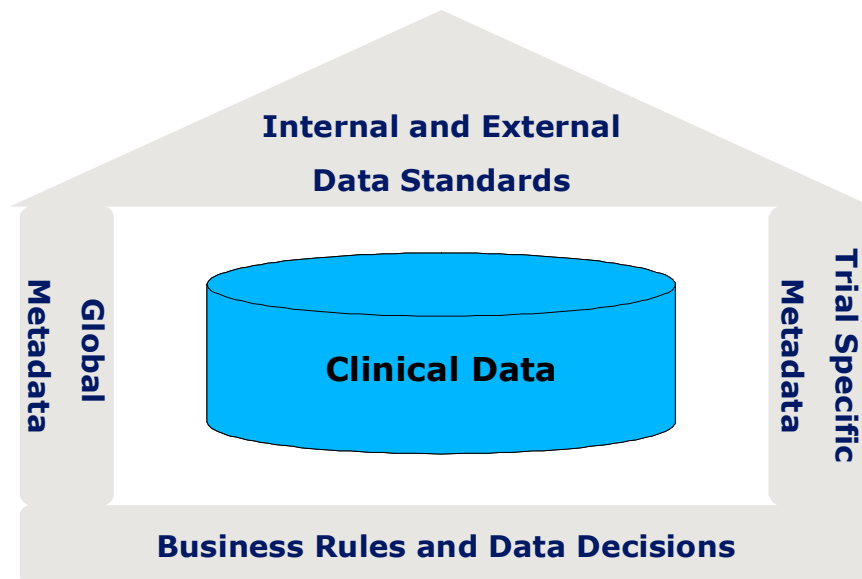


Figure 1 Clinical data and categories of metadata in the data warehouse

SOURCES FOR METADATA

The collection of metadata origins from several sources. Some of the metadata already exist but in a form not very suitable for data processing, such as the trial design and trial flowchart defined in the protocol. Some of the metadata exist in a more implicit form, such as data decisions and business rules in programs. To be able to use these metadata they need to be collected and stored in a generalised way as illustrated in Figure 2, so we can utilise them in our programming.

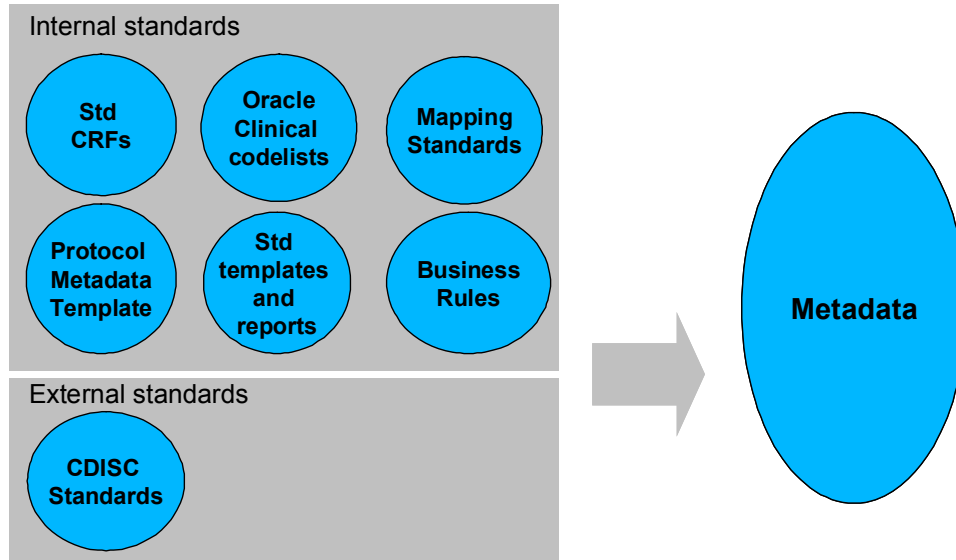


Figure 2 Sources for metadata

THE CLINICAL DATA WAREHOUSE

In order to facilitate collection of metadata in a standardised way a data model and a data warehouse holding all the metadata information together with all the clinical data information have been developed. An overview of the different systems that are supporting the data warehouse is given in Figure 3. The data warehouse applications are defined by three parts:

- The Metadata application - CDW Operations Application. This is the application used to enter trial metadata, global metadata and a mapping application to map between collected source data and the generalised data model.
- The Data Repository containing all data
- The Analysis Platform where the data is stored in an enriched version together with all standard programs. This is the environment where all analysis is done and statistical output is generated.

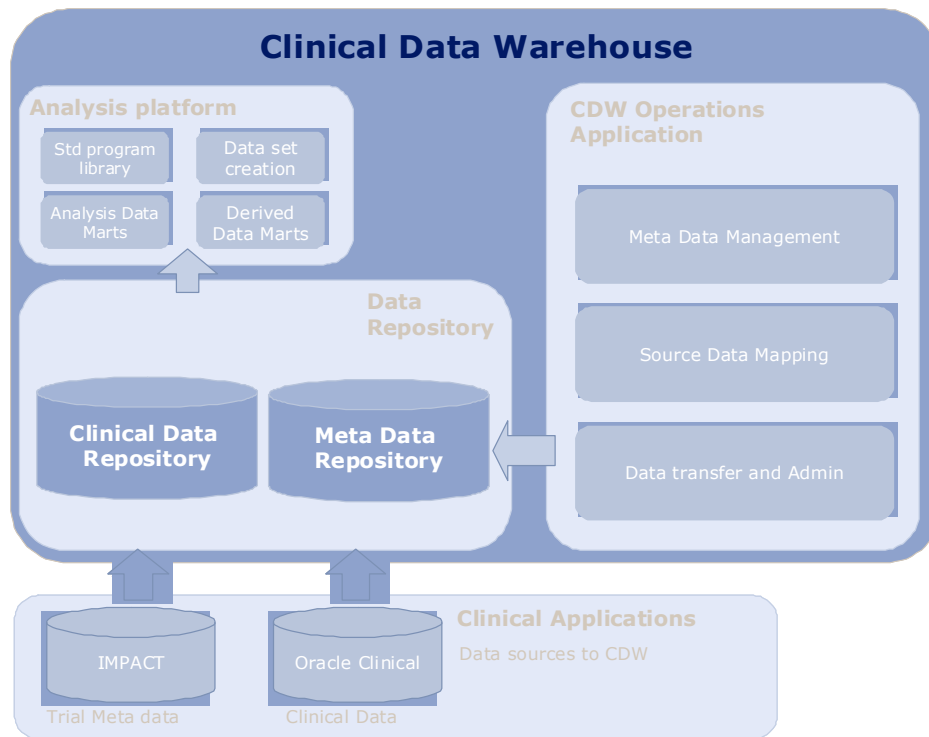


Figure 3 Overview of the Clinical Data Warehouse

In the data warehouse the metadata is divided into global metadata and trial specific metadata.

GLOBAL METADATA

Global metadata is defined across all projects and trials. Examples are the generic trial designs described through attributes such as:

- type of trial design
- numbers of treatment elements
- numbers of epochs
- numbers of trial arms

The code lists used for the clinical data are also global metadata. Here the categorical responses are defined, but also restricted values used for the different attributes in the trial specific metadata and business rules. In Figure 4 an example of a typical global metadata entry screen is seen where metadata on a particular topic code (measurement) can be updated or created. In this example a specific topic code is defined with the corresponding:

- output format
- labels
- description
- sorting sequence
- categories

Furthermore a standard unit and molecular weight for the topic code are defined enabling a standardised unit conversion. This is utilised in the standard programs producing tables since a column is created in the derived datasets holding the finding values converted to standard units.

Maintain Finding Definition

Define common Finding Definition attributes

Finding Topic Code: HBA1C_BLOOD
 Short Finding Topic Code: HBA1C_B
 Finding Label: Blood HbA1c
 Finding Short Label: HbA1c
 Description: CDISC submission value: HBA1C
 Hemoglobin A1C. A measurement of the glycosylated hemoglobin in blood.
 Sort Sequence: 810
 Status: Active
 Language: English (U.K.)
 Standard:
 External Terminology:
 Finding Category: Glucose metabolism
 Finding Sub Category:
 Finding Type: Numeric Finding
 Derived Variable:
 SAS Display Format: 5.1
 Default Flowchart Group: Efficacy

Figure 4 Example of a data entry screen for global metadata for a Topic Code

TRIAL SPECIFIC METADATA

Trial specific metadata defines the trial by use of the global metadata and by use of additional attributes. Examples are:

- the actual treatment elements in the trial
- the actual trial arms
- the blinding level
- the visits
- visit type and relation to the trial elements and epochs

Also the trial flowchart information is defined by attributes like:

- type of information (finding, event or intervention)
- class of finding
- collection information (singleton, by visit, repeated measurement)
- collection visits,

In Figure 5 an example of a setup for visits in a study is given with the trial specific time and windows, epochs, types and IDs.

Visit ID	Planned Trial Epoch	Planned Trial Time	Visit Type	Visit Type Alias
10 V1	Screening Epoch	Reference: Baseline Time: -1 Weeks Window: 0 / 7 Days	Screening visit	Visit 1
20 V2	Treatment Epoch	Reference: Baseline Time: 0 Weeks Window: 0 / 0 Days	Baseline/Randomisation visit	Visit 2 Randomisation
30 V3	Treatment Epoch	Reference: Baseline Time: 1 Weeks Window: -2 / 2 Days	Treatment visit	Standard site visit
40 V4	Treatment Epoch	Reference: Baseline Time: 2 Weeks Window: -2 / 2 Days	Treatment visit	Standard site visit
50 V5	Treatment Epoch	Reference: Baseline Time: 3 Weeks Window: -2 / 2 Days	Treatment visit	Standard phone contact
60 V6	Treatment Epoch	Reference: Baseline Time: 4 Weeks Window: -2 / 2 Days	Treatment visit	Special site visit
70 V7	Treatment Epoch	Reference: Baseline Time: 5 Weeks Window: -2 / 2 Days	Treatment visit	Standard phone contact

Figure 5 Example of trial metadata for visits

PhUSE 2009

BUSINESS RULES AND DATA DECISIONS

The business rules and data decisions are defined so they can be selected and applied to the data points – both as attributes to the flowchart information and as flags and populations defined for the trial. These rules are utilised by the standard programs to control which algorithm to use when e.g. performing visit reallocation or handling partial or missing dates and times.

In Figure 6 an example of a data entry screen for definition of business rules for collection of a topic code at a visit is shown. In this example it is, besides the pre and post visit date and first and last drug date windows, possible to specify which algorithm that should be utilised when a finding value is missing or a date or time value is missing or partial.

Flowchart Group:	Glucose metabolism
Assessment:	Blood HbA1c
Visit ID:	10
Visit Short Label:	V1
Treatment Allocation:	<input type="text" value="Current Visit"/>
Assessment Type:	<input type="text" value="Baseline Assessment"/>
Mandatory for PP Population:	<input type="checkbox"/>
Qualify for FAS Population:	<input type="checkbox"/>
Pre Visit Date Time Window:	<input type="text"/> Days
Post Visit Date Time Window:	<input type="text"/> Days
Pre First Drug Date Time Window:	<input type="text"/> Days
Post First Drug Date Time Window:	<input type="text"/> Days
Pre Last Drug Date Time Window:	<input type="text"/> Days
Post Last Drug Date Time Window:	<input type="text"/> Days
Retest Rule:	<input type="text"/>
Missing Finding Value Rule:	<input type="text"/>
Visit Reallocation Rule:	<input type="text"/>
Missing Finding Date Rule:	<input type="text"/>
Missing Finding Time Rule:	<input type="text"/>

Figure 6 Example of a data entry screen for setting trial metadata and business rules for a Topic Code at a visit

THE STANDARD PROGRAMS

The standard programs developed make use of the structured data collection using the data model, the global metadata in form of code lists and standard units, the structured and generalised information of the trial specific metadata and the defined business rules and data decisions.

The Standard Programs have been defined in two layers: In the Derived Variables and Enrichments (DVE) programs and in Statistical Analysis and Reporting (SAR) programs.

DVE LAYER

The Derived Data Marts (DDM) are defined using the business rules and generalised data-driven methods for:

- blinding
- treatment allocation
- unit conversions
- definition of planned findings
- setting of flags
- derived variables

The DVE layer consists of programs utilised to enrich data into the DDMs that are the basis for the programs that produces the statistical output. This layer consists of the programs that implement the business rules, enrichments, populations and conversions defined in the metadata.

PhUSE 2009

The DVE Layer is implemented as a framework consisting of one program that calls the necessary macro components in order to enrich the data as specified in the metadata. A central DVE framework program is kept up to date by the standard programming team with the newest DVE components and the newest version of the data model. This program implements all available business rules and populations however not all rules/populations are needed in all trials. Since only a subset of the business rules and populations are needed by trials each trial team develop their own DVE framework program based on the central DVE framework. In Figure 7 parts of a DVE framework program is given.

```
17
20 * Bmi;
21 %mk_dve_bmi_v02(path=ddm_temp
22                 ,topic_code_weight=BODY_WEIGHT
23                 ,topic_code_estimated_weight=WEIGHT_ESTIMATE
24                 ,topic_code_height=HEIGHT
25                 ,topic_code_estimated_height=HEIGHT_ESTIMATE
26                 ,assesment_types=BASELINE
27                 ,topic_code_bmi=BMI);
28
29 * Randomised;
30 %mk_dve_randomised_v02();
31
32 * Exposed;
33 %mk_dve_exposed_v02;
34
35 * Safety Analysis Set;
36 %mk_dve_safety_v02();
37
38 * ITT Analysis Set;
39 %mk_dve_itt_v02();
40
41 * Full Analysis Set;
42 %mk_dve_fas_v02();
43
44 * PK/PP Analysis Set;
45 %mk_dve_pkpd_v02();
46
47 * Withdrawn;
48 %mk_dve_withdrawn_v01();
49
50 * Completed;
51 %mk_dve_completed_v01();
```

Figure 7 DVE framework program executing specific macro components for defining population flags and other enrichments

In Figure 7 the calls for macros producing population flags (Intention to Treat, Safety, Full Analysis Set, etc.) are examples of calls for macros that may differ from trial to trial based on the type of trial and the contents of the Statistical Analysis Plan (SAP). The trial teams then substitutes the macro components with trial specific components. This however is a non-automated process that has to be done by the trial teams manually for now. In the future a code generator that based on trial metadata populates a DVE framework program with the correct macro calls could be developed.

SAR LAYER

The statistical analyses and reports (SAR) programs make use of the defined data in the metadata and the DDMS to make user friendly interfaces and easy choices in the extraction and analysing of the data. Also the output from the analysing and reporting has an output metadata layer to make easy collection into Word files and appendices possible.

The programs in the SAR Layer consist of two types of SAS® programs: building block macros (BBM) and standard reporting programs.

The BBMs are validated SAS macros that can be utilised by the trial teams in custom programs without validation of anything but the call for the macros. These macros are utilised in the standard reporting programs and used for extraction of data, calculation of statistics, printing of data etc.

The standard reporting programs are SAS programs that generate actual report output in .txt, .rtf and/or .html format. These programs utilise the BBMs to extract data, process the extracted data and to print the data. The standard

PhUSE 2009

reporting programs utilise the BBMs the same way that the DVE Framework utilises the DVE macro components by calling them one after another parameterising them for the needed functionality. An example is given in Figure 8.

```
1  %macro tsum_findgroup_v02();
2      %access_sdd_v02();
3
4      *****;
5      * Extract data and generate user selection specific sorting formats;
6      *****;
7      %extract_find_v02();
8
9      *****;
10     * Generate the statistics for the categorical findings;
11     *****;
12     %calculate_catgo_stats_v02();
13
14     *****;
15     * Generate the statistics for the numeric findings;
16     *****;
17     %calculate_num_stats_v02();
18
19     *****;
20     * Append the statistics and add the Number of Subjects;
21     *****;
22     %append_stats_v02();
23
24     *****;
25     * Generate the titles as the macro variable title_final;
26     *****;
27     %create_title_v02();
28
29     *****;
30     * Setup output destination;
31     *****;
32     %init_start_v02();
33
34     *****;
35     * Print the report;
36     *****;
37     %print_summary_v02();
38
39     %init_end_v02();
40
41 %mend tsum_findgroup_v02;
42 %tsum_findgroup_v02();
```

Figure 8: Example of Standard Reporting Program utilizing Building Block Macros

The standard reporting programs are built in an environment that converts macro variables to parameters in a user interface. When the programs are executed or placed in a job the User Interface pops up and the user can choose among metadata grouped in selection lists. This metadata is the metadata that has been entered in the Clinical Data Warehouse and transferred through the different stages until the last stage as selection list content in the User Interfaces.

Typical examples of metadata controlled content in the User Interfaces are selection lists containing:

- populations
- sub groups
- visits
- topic codes

An example of this is given in Figure 9. In this example the Population selection list is populated with the populations that have been setup by the trial team in the metadata and enriched by the macros called from the DVE framework program. This selection is used to filter data based on a flag variable generated by one of the DVE programs, the mk_dve_itt_v01 macro seen in Figure 7.

PhUSE 2009

The screenshot shows a software interface with several sections:

- Report Selections:** A tabbed menu at the top with options: Report Selections1 (selected), Report Selections2, Layout Selections, Statistics, EOT Specific, and 1.
- *Instance:** A text input field containing "/SDD/Test Data/Test0/Current from SDD", with "Browse..." and "Clear" buttons to its right.
- Trial Population:** A dropdown menu currently showing "ITT".
- Trial Sub Group:** A dropdown menu currently showing "Sex=M".
- Treatment Variable:** A tree view showing a folder "ui_arm_elem.sas7bdat" containing three items: "TRL_BRANCH_ARM_SHORT_LB", "TRL_ELEM_SHORT_LB" (highlighted in blue), and "TRL_ARM_SHORT_LB". Below this is a "User-generated columns" folder and a "Clear" button.
- Treatment:** Two lists: "available" and "selected". The "available" list contains: "<no selection>", "BIAsp 50", "BIAsp 70", "Follow-up", and "Screening". The "selected" list is empty. Between the lists are buttons ">", "<", and "<<". To the right of the "selected" list are buttons "A" and "V".

Figure 9: Example of typical User Interface with metadata based selection lists

In the example in Figure 9 the selection lists for Trial Population, Trial Sub Group and Treatment are based solely on Trial metadata entered by the trial teams in the metadata application and enriched by the DVE framework program.

In Figure 10 another example of the User Interface is given. The two selection lists for Visit ID and Change Visit ID depend on the choice of topic code. In the example BP_DIASTOLIC (the code for Systolic Blood Pressure) has been chosen and the two Visit selection lists are filled with the Visits IDs where the systolic blood pressure is planned to be measured. If the user chooses another topic code, e.g. BMI, the content in the two selection lists will change accordingly. This data is based on the trial specific metadata entered in the metadata application by the trial teams.

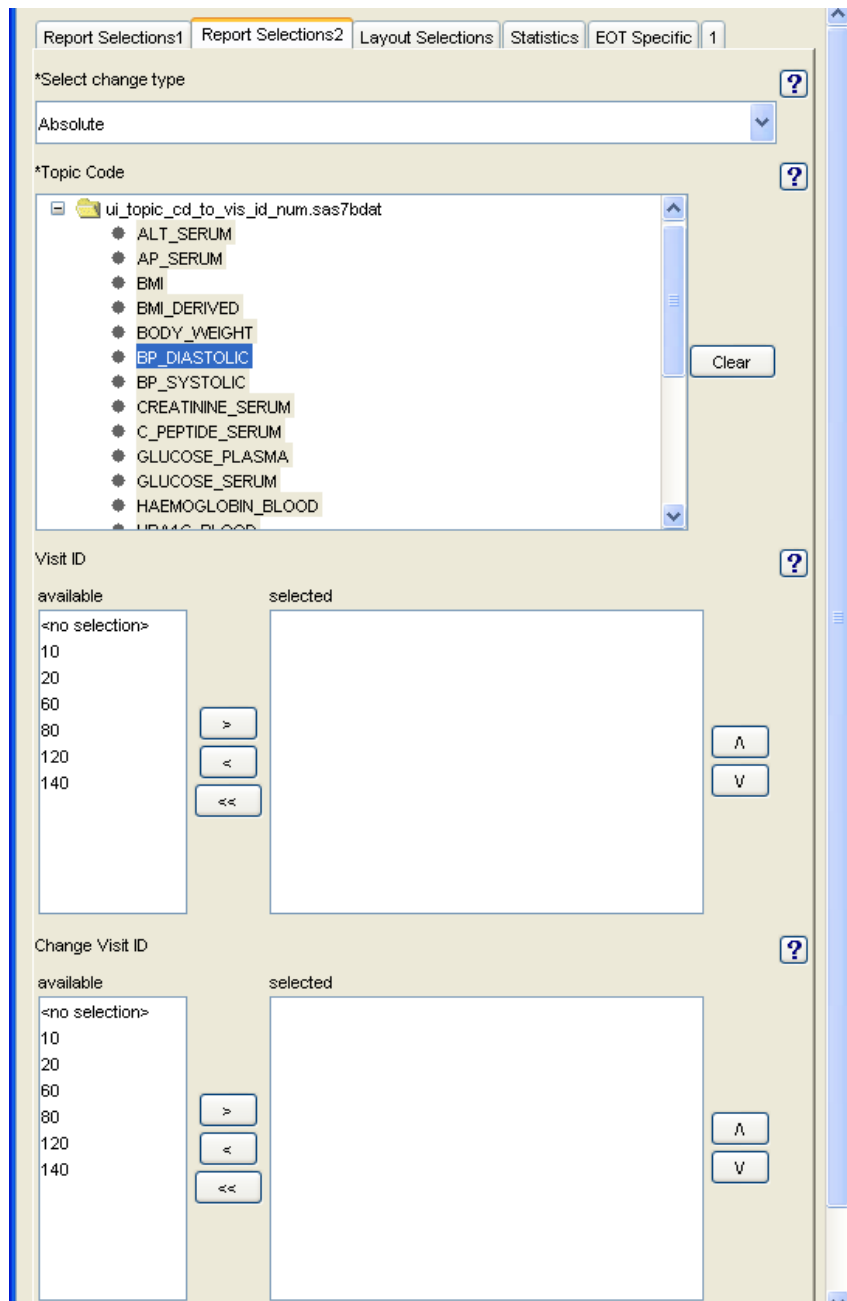


Figure 10 Metadata based selection lists

The final output is very dependent on both trial specific metadata and global metadata. The presentation of numbers in the output is metadata based since the format used to display different statistics are topic code dependent. Some topic codes must be displayed using 8.4, others using 8.1 etc. Figure 11 shows an example of output from a standard reporting program with indications of what parts of the output that are metadata controlled.

PhUSE 2009

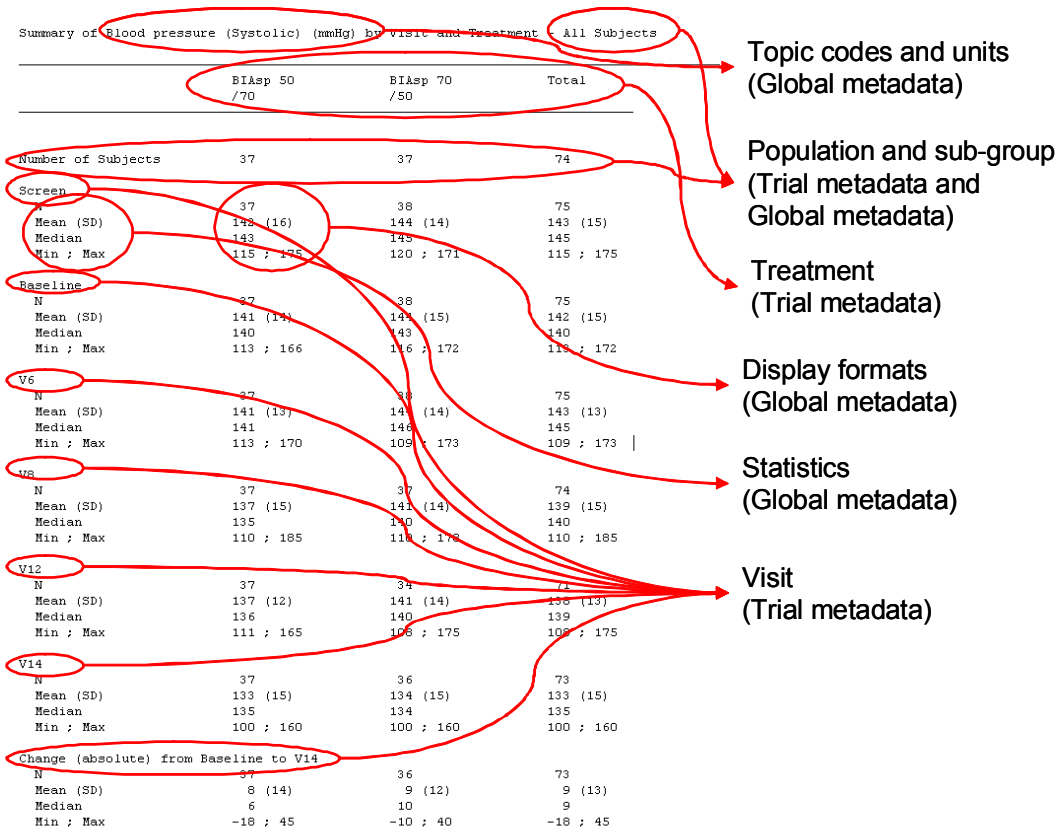


Figure 11 Metadata usages in the final output

In Figure 11:

- The topic code in the title, called BP_STYSTOLIC, has been converted using a format to Blood Pressure (Systolic) and the standard unit have been added, both based on global metadata.
- The Number of Subjects statistics are based on data filtered to only include the subjects in the chosen population and sub group, this is based on trial specific metadata.
- Headers (BIAsp 50/70, BIAsp 70/50) are based on labels defined in the trial metadata.
- The labels that specify the visit ids are based on the visit metadata entered in the metadata application.
- The numbers in the output have been formatted using the topic code specific SAS format as mentioned earlier.
- The title part containing “– All subjects” would have contained the global metadata based label for a population followed by the trial metadata based label for a sub group if any had been chosen, e.g. “Summary of Blood Pressure (mmHg) by Visit and Treatment – Safety Analysis Set, Sex = M”.

CONCLUSION

By making use of the data model, metadata and generalised explicit definitions of business rules and data decisions, it is possible to make an intelligent, data-driven creation of derived data marts, analysis datasets and reporting output. Standards are utilised and implemented and faster development facilitated.

CONTACT INFORMATION

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

Martin Lindhard and Marianne Caramés
 Novo Nordisk A/S
 Vandtårnsvej 114
 Søborg / 2860
 Denmark

Work Phone: +45 30 75 10 60 and +45 30 79 12 11

Email: mrli@novonordisk.com and maca@novonordisk.com

Brand and product names are trademarks of their respective companies.