Paper CD10

Creating ADaM Friendly Analysis Data from SDTM Using Meta Data

Erik Brun, H. Lundbeck A/S, Valby, Denmark
Rico Schiller, H. Lundbeck A/S, Valby, Denmark

ABSTRACT

This paper introduces how it is possible to create standardised ADaM friendly analysis data with SDTM (3.1.x) as source using meta-data. The validated system, SADs4, fully incorporates CDISC standards and principles in the dataflow process at Lundbeck. The system requires the Lundbeck SADs data model, the Lundbeck meta-data dictionary, and a number of control tables specifying the study setup. The control tables contain information such as windowing, imputation methods, study specific derivations, and additions to the data model. The system has proven to be flexible, fast, and fairly easy in use. It will work with a wide variaty of indications. An additional gain is that we are able to make data integrations very fast, due to the consistent structure of the output data. The generated datasets can be used by our validated standard package of analysis and reporting programs.

INTRODUCTION

SDTM data is an organised raw data format in which every piece of data has its own single and unique designated place. There are no redundancy apart from the key variable that links the different SDTM data domains, the unique subject identifier.

To facilitate analysis and reporting, data enrichment is needed. Data enrichment means that data points can be repeated in multiple places and in different formats (for example both as character and numeric). Data enrichment also means that data points can be added that were not in the raw data. The SADs (Statistical Analysis Data sets) are information enriched data structures for statistical analyses.

The SADs contain a large number of derived variables. Derived variables can be copied from a specific observation (for example baseline blood pressure and pulse rate values), computed (such as body mass index) or derived (like imputation flags for derived data according to the method of Last Observation Carried Forward).

Having a fully validated and documented system that works well and is continuously kept in its validated state is of great value to Lundbeck. The validation and documentation of the system assists greatly during audits, but also enables junior programmers to start working very quickly with the system with only marginal assistance from experienced programmers. Additional thoughts and effort has been put into the error handling process, making error messages much more intuitive and understandable. The standardised structure of the data has also made it easy to integrate across studies.

HISTORICAL OVERVIEW:

In preparation for use on a submission, Lundbeck initiated a project to create a standard set of analysis datasets. This project resulted in the development of SADs v.1, and was validated for general use in 2000. Based on the experience gained during the submission, the SADs System was further enhanced and Version 2 was validated for general use in 2001.

SADs System Version 3 was developed to accommodate more complex study structures (multiple periods for different treatments, multiple baselines, and extension studies). In addition an in-house ETL-tool was developed; this allowed data-mapping and transformation of data from different data sources into the SADs structure. Version 3 was validated for general use in 2004.

All versions of SADs have a matching set of standard data analysis and reporting programs.

As Lundbeck moved into new therapeutic areas (acute indications such as stroke) the system has been challenged, e.g. by requiring a higher resolution in time and the ability to deal with local laboratories. Furthermore, the data model was embedded in the source code, and this made it extremely difficult to update the data model.

A large project was initiated. The project would touch upon all areas of the data handling process in the clinical areas. As part of this project it was decided to not only update the SADs, but to also update the standard reporting program package.

The objectives for a new system were:

- Create the basis upon which the automated and validated production of consistent and standardised statistical analysis reports and listings for safety and efficacy data is possible within Lundbeck's Division of International Clinical Research
- The system should allow for clear documentation of the configuration settings applied in a single study

- Standardise the structure and interpretability of detailed clinical data delivered to regulatory authorities, affiliates, strategic alliance partners based on CDISC terminology and concepts. Ever since the development of the SADs System Version 3, Lundbeck has pursued a strategy of applying CDISC standards in all scientific data models.
- Provide a validated and controlled environment for the collection and integration of clinical data across studies within a drug project

THE SOLUTION

SADs4 uses SDTM v 3.1.x as source data.

The SADs system, illustrated in Figure 1, consists of following components:

- a generic SAS® master job specification file
- a library of generic SAS macros
- a set of study specific configuration tables, in essence an Microsoft Excel work-book
- the SADs data model
- Data Capture Dictionary (DCD) – Lundbeck's meta-data dictionary

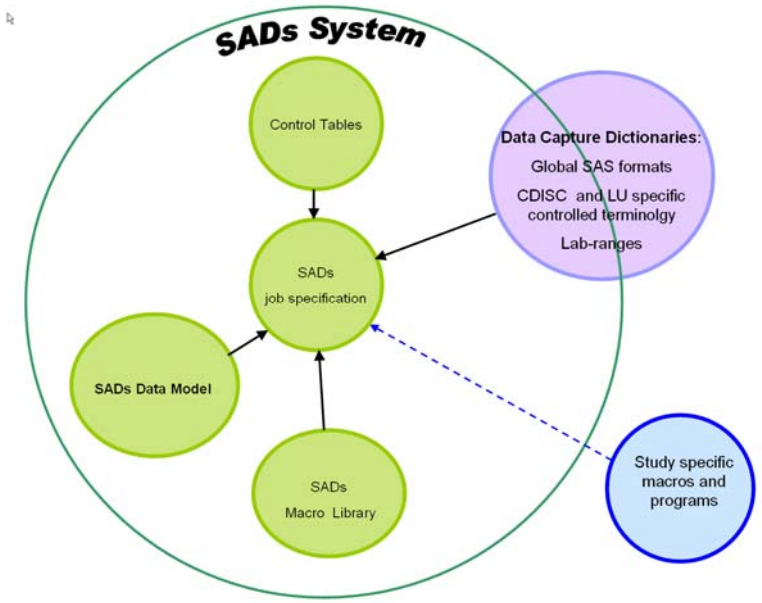

Figure 1 SADs System internal components and external supporting features

MACRO LIBRARY

The SADs System is written in SAS Language, and the environment is SAS 9.2 in CDR (Clinical Data Repository "CDC Waban SCE 2.5.1"). The macro library consists of 35 SAS-macros, some of these will be invoked multiple times during the SADs creation.

CONTROL TABLES

Configuration of the SADs output by means of control tables (Study metadata) allows great flexibility when tailoring the datasets for reporting. At the same time, the process is based on generic validated code that can be reused across studies and projects.

In the 19 control tables the user amongst other configures sitegroup assignment, date imputations, derivations (totals, AVISIT, AVISITN), general settings (windowing method(s), which baseline values are added to *Core_pat* (ADSL)), laboratory ranges used, windowing rules, period assignment, sort order of output datasets, generation of indices, study specific additions to the data model, definitions of PARAM, which periods a certain param is imputed at missing visits

The macros in the macro library have a substantial amount of check ensuring that input from the control tables is sensible, if not the macro will terminate the SADs run.

Example 1: Date imputations

In SDTM, dates (and date-times) are stored in a character variable such as AESTDTC. In SADs, a date is represented by three variables; AESTDTC - the raw SDTM character, AESTDTN – the numerical representation (date-time), AESTDT_CD – an imputation flag, indicating which imputation method has been applied and the accuracy of the raw value.

In figure 2 the first column tells which dataset, the second the name of the numerical date variable. The third column is controlling which imputation rule should be applied when an incomplete date is encountered (EARLY or LATE). The fourth column "Expected" is expected accuracy (DAY or MINUTE)

	A	B	C	D	E
1					
2	Data Set	Variable	Rule	Expected	Limit
3	ADVERSE_EVENT	AEENDTN	LATE	DAY	
4	ADVERSE_EVENT	AESTDTN	EARLY	DAY	DOSE_STDTN
5	CORE_PAT	BRTHDTN	EARLY	DAY	
6	CORE_PAT	CDBRDTN	EARLY	DAY	DOSE_STDTN
7	CORE_PAT	DOSE_ENDTN	LATE	DAY	
8	CORE_PAT	DOSE_STDTN	EARLY	DAY	
9	CORE_PAT	DSENDTN	EARLY	DAY	
10	CORE_PAT	DSDECAS_LEVDTN	LATE		

Figure 2 – Date imputation rules

Input	Settings	Output
AESTDTC = ”2011-08-07”	Rule=”EARLY” Expected=”DAY”	AESTDTN = 07AUG2011:00:00:00
AESTDTC=”2011-08”	Rule=”EARLY” Expected=”DAY”	AESTDTN = 01AUG2011:00:00:00
AEENDTC=”2011-08”	Rule=”LATE” Expected=”DAY”	AESTDTN=31AUG2011:00:00:00
AEENDTC=”2011-08”	Rule=”LATE” Expected=”MINUTE”	AESTDTN=31AUG2011:23:59:00
AESTDTC=”2011-08”	Rule=”EARLY” Expected=”DAY” Limit=DOSE_STDTN (DOSE_STDTN=07AUG2011)	AESTDTN=07AUG2011:00:00:00

Example 2 Windowing of visits

Fig. 3 is a sample from the control table that is used when windowing visits, for assigning treatment periods, setting baseline flags and several other data operations.

During the creation of the SADs, each patient will have all attended visits assigned to a study period in accordance with the specifications provided in the control table The study period is equivalent, but not identical, to the term EPOCH in SDTM.

AVISITs are windowed to a nominal day (analysis day) when the date of visit is inside the window described by the columns WFROM and WTO.

Points for Baselines are defined by what triggers the period. The baseline observation is the last observation in time preceding the baseline definition

Several window models (First column) will be used in most studies as it rarely occurs that all measurements are made at all visits during a study. This way it is avoided that unscheduled measurements are windowed into visits where the measurements were not planned.

Window_model	PERIODN	PERIOD	AVISIT	TRIGGER	NOMINAL_DAY	WFROM	WTO	BASELINE
default	1	SCREEN	SCREEN		-7			
default			BASELINE		0			
default	3	TREATMENT	Visit 3	DOSE_STDTN	7	1	10	PRIMARY
default			Visit 4		14	11	17	
default			Visit 5		21	18	24	
default			Visit 6		28	25	35	
default			COMPLETION		42	36	56	
default	4	WASH OUT	Down Taper		49	0		WASHOUT
default	5	FOLLOW UP	Follow Up	DHMS(DATEPART(EXENDTN+7), HOUR(TIMEPART(EXENDTN)), MINUTE(TIMEPART(EXENDTN)), SECOND(TIMEPART(EXENDTN)))	70	0		

Figure 3 Timing control table (Columns omitted for simplicity and readability)

DATA MODEL

The user can add new variables, but neither delete nor change attributes of existing variables. Windowing methods, multiple baselines and imputation methods such as LOCF and BOCF, all result in structurally identical datasets. Consequently, many datasets may be generated from one source (depending on the settings). The SADs model meta-data structure contains a total of 13 distinct data structures: one for each dataset (AdverseEvents, Demographics etc.)

efficacy data (scales) are, though, described in two tables in a generic way. The use of the SADs Data Model ensures consistent data structures across projects and studies, even when extensive study-specific configuration takes place.

DATA CAPTURE DICTIONARIES (DCD)

DCD includes:

- controlled terminology
- global SAS formats
- applicable laboratory ranges

This borderline component of the SADs System is mandatory to the SADs creation process. Both SAS formats and laboratory-ranges were an integrated part of the previous versions of the SADs System. As part of the redesign in SADs System v.4.0, the SAS formats and laboratory range management have been promoted to a global level of the Data Capture and Analysis process to enhance governance and harmonisation across data processes, projects and studies.

The SADs System offers the flexibility of including study-specific local macros or local programs from a designated location in the study folder structure of the CDR. Such local components are not considered an integrated part of the validated generic system itself and will, of course, always be subject to QC procedures.

CDISC STANDARDS

CDISC standards are applied whenever possible, though longer (more meaningful) variable names are occasionally desirable in daily use.

As SDTM data is the source for SADs, variable names are retained for the traceability when the data is unmodified and in the same context. (For example, USUBJID, LBCAT, CMSTDTC, AETERM, and VSSEQ). Appendix D (CDISC variable-naming fragments) in the SDTMig has been used when naming variables whenever possible.

Examples of ADaM concepts applied in the system:

AVAL that is equal to --STRESN for values that are not derived, a set of qualifying variables like the baseline, derivation type, and “evaluability” flags, change from baseline values.

Analysis visit (AVISIT/AVISIT) is assigned as well as the original values are kept in the VISIT/VISITN variables.

PARAM/PARAMCD is assigned using a control table. This table tells the program how to make the look-up in the DCD. The use of the DCD ensures consistency across studies and projects.

The SADs System does not attempt to compensate for incorrect input data. Dirty input data will result in dirty analysis datasets. Input data, which is incomprehensible as SDTM data, will result in no output. The program issues alerts when data seems to be dubious.

DATASET CREATION SEQUENCE:

The quite busy figure 4 shows how and in which sequence the datasets are created.

- The initial process, Process I, creates an intermediate dataset that is used by most of the following processes. This is also the first of two processes creating the ADSL (demographics) dataset, called Core_Pat.
- Process II creates the exposure dataset (Exposure)
- Process III creates the Study_Ranges dataset. This is a dataset contains an overview of ranges (laboratory, vital signs, and ECG) applied in the study.
- What is often perceived as BDS in ADaM, the efficacy (SDTM QS) and examinations datasets are created in process IV. This process is by far the most demanding in terms of resources.
- Process V that creates the visits dataset (Core_Vis).
- The “events” datasets; Adverse Event (Adverse_Events), Concomitant Medication (Medication) and Medical History (History) are subsequently created in process VI.
- In Process VII the creation of Core_pat is finalised with the population of information from the processes II-VI.
- The finalisation of the SADs creation job is process VIII that creates the comments dataset.

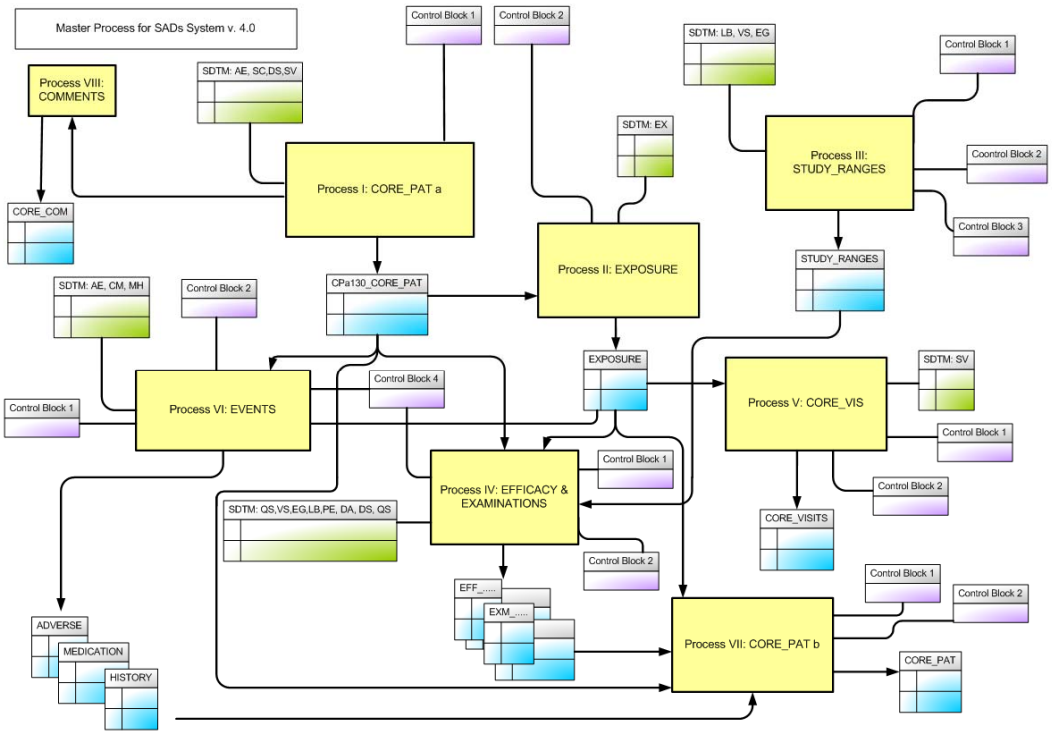

Figure 4 – Dataset creation sequence

Each of these processes consists of a number of steps (Figure 5). A step corresponds to the invocation of a SAS macro. The macro may read one or more control tables (Lavender boxes). The control table input is checked for consistency. When an inconsistency is detected, the program issues a fail message and halts the SADs generation. Next, the program checks the input data (the green boxes and/or the output from the previous step). Checks whether the dataset exists and whether it contains the expected variables and only then it makes the actual data manipulation.

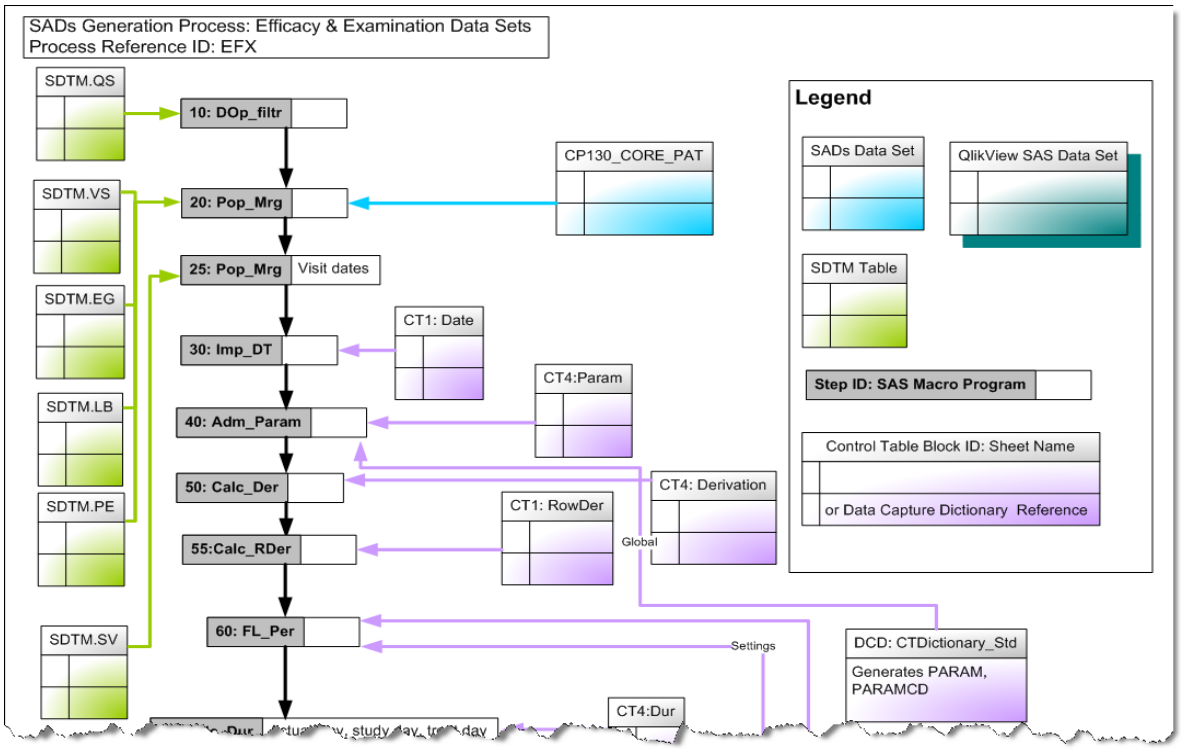

Figure 5 – Sample from a process map

CONCLUSIONS

We have succeeded creating a system that works. It can handle most study designs, except for some very complex designs, like studies with a not pre-defined number of treatment cycles. Though complex, the system is still transparent.

The generated datasets are used by our validated standard package of analysis and reporting programs.

A junior programmer can make a first draft of a normal study within a couple of days. The downside of this is that it is not challenging for experienced programmers as little programming is required for standard studies.

Some may ask: Why not ADaM? The design of the SADs system was the art of the possible. This project started before the ADaM model officially went live, we had talks with a member of the CDISC ADaM team, which enabled us to build in ADaM concepts. And, just as important, the SADs data structures should also be able to be accepted by the statisticians. For example ADaM would require a greater number of datasets (considering the “one proc away principle”). ADaM datasets could easily be generated based on SADs datasets. The main activity needed will be renaming of variables and a little typecasting.

A system like this makes data integration across studies quite easy – basically just drop variables not in the standard data model and append the datasets. Especially when a consistent naming for periods and the like has been applied in a project.

Initially, it was attempted to create the SADs 4 system using SAS DI. We can not recommend using such a tool for this sort of development as the systems needed much more flexibility than we could achieve with SAS DI.

The control tables are fairly comprehensive; in the future an application of the PRM (Protocol Representation Model) that SADs could read would greatly reduce the amount of metadata that needs to be entered into the control tables. If the SAP got a human and machine readable design as well, the analysis data set generation could be far more automated.

REFERENCES

CDISC MODELS

SDTM ver 1.2 and SDTMig v3.1.2

ADaM ver 1.0 and ADaMig v2.1

PRM ver 1.0

ACKNOWLEDGEMENTS

Thanks to our co-developers Karina S, Sven J, Camilla T, Kaj N, Andrej D, and Jon R. Without their help the system would probably have remained sketchy ideas on colourful post-it notes.

Thanks to Ross B for proof-reading and sanity check of this article

CONTACT INFORMATION

Erik Brun, System & Process Specialist

H. Lundbeck A/S

Ottiliavej 9

2500 Valby

Denmark

erik@lundbeck.com

www.lundbeck.com

Rico Schiller, Head of Section

H. Lundbeck A/S

Ottiliavej 9

2500 Valby

Denmark

rico@lundbeck.com

www.lundbeck.com