

Developing Reporting & Analysis Methodologies

Amy Gillespie, Merck Research Laboratories, Upper Gwynedd, PA
Mary Varughese, Merck Research Laboratories, Upper Gwynedd, PA

ABSTRACT

SAS® programming for statistical analysis and reporting has been industry standard for many years. Currently when working on late stage development projects, programmers and statisticians often develop their SAS programs in a PC environment taking full use of available departmental SAS macros or by developing their own code. Reporting on safety data is often a standard and rigid process while reporting and analysis of efficacy data can require flexibility due to specific study needs. Programmers and statisticians have become experts utilizing the SAS system, often very creatively to continually strive for higher efficiency and quality.

Recognizing the impact of new industry data standards and compliance requirements on our programming, we led a team of individuals to analyze our current processes and recommend a new approach to analysis and reporting. This process utilizes the CDISC Study Data Tabulation Model (SDTM) as input for all analysis and reporting programs, as well as a 21 CFR compliant repository and a PC SAS programming environment. This paper presents our challenges and decisions when developing new methodologies for analysis and reporting and highlights the usage of new industry data standards.

1. INTRODUCTION

Our goal was to create a single process and methodology for all analysis and reporting. While our primary focus was on late stage development studies, the design is flexible enough to be used in early development studies as well. This single process applies to all programs to support study design, internal decision making, data set creation, exploratory analyses or results submitted to a regulatory agency. More specifically, programs supporting the statistical analysis and reporting for safety, efficacy, and immunogenicity will follow this new design.

Historically, our reporting and analysis programming approach was different for four types of reporting labeled as standard, project specific, traditional, and exploratory. Definitions for each of these reporting approaches are included below:

- **Standard Programming:** programming in support of tables required for WMA submissions which included adverse event, demographics, study status and therapy reporting.
- **Project Specific Programming:** programming in support of tables required for WMA submissions for which a standard program (as defined above) did not apply due to project or protocol specific logic and needs.
- **Traditional Analysis Programming:** programming in support of statistical analysis which includes efficacy, safety, and immunogenicity.
- **Exploratory Analysis:** Includes but not limited to analyses performed at the design stage of new studies, analyses checking the robustness of results, and analyses requested from clinical that are more ad hoc in nature.

These four reporting approaches use different data sources (live data versus derived data), environments for program development, test, validation, and production (PC versus UNIX), macro libraries, and program execution methodologies (GUI versus calling programs). These differences led to inefficiencies in our programming and a clear opportunity for improvement.

This paper will discuss a design for a single and consistent analysis and reporting approach for all types of programming. This design can be adopted for all phases of a clinical trial. Section 2 presents an overview of the new design methodology for analysis and reporting.

2. THE DESIGN

All programming for safety, efficacy, and immunogenicity is considered under one large umbrella called single analysis and reporting. Single analysis and reporting combines all programming historically referred to as standard, project specific, traditional, and exploratory. The design decisions for single analysis and reporting include the following:

- All development programming will be completed utilizing a local code development environment in PC SAS. All testing, validation, and execution of the programming code will also be completed using PC SAS. Refer to section 2.1.
- The data source for all single analysis and reporting programs will be SDTM+ and ADaM SAS V9 datasets. Refer to section 2.2.
- The structure for all analysis and reporting programs will follow a consistent look and feel with similar inputs and outputs. Refer to section 2.3.
- One global macro library will be utilized for all analysis and reporting. The global macro library is discussed more fully in another contributed PharmaSUG paper.
- The final archive for all programs, macros, tables, figures, datasets, and documents will be within a 21 CFR part 11 repository.

2.1 Development, Testing, Validation, and Execution in PC SAS

The programming environment for single analysis and reporting will be PC SAS. Options other than PC SAS were investigated as possible code development and execution environments. However none of the current available options met our requirements for editor functionality and performance. Since our ultimate goal is to be working in a server environment, the use of PC SAS will continue to be evaluated in the future.

2.2 SDTM+ and ADaM Data Source

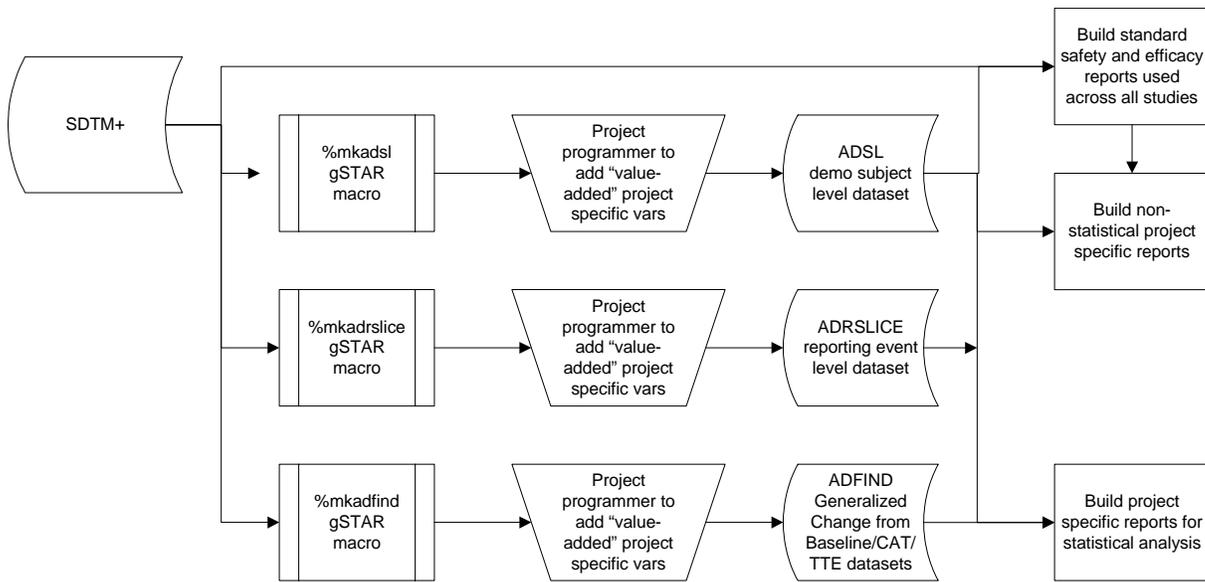
Our Clinical Data Repository (CDR) provides a single data repository for clinical patient data (CRFs) across all therapeutic areas. The CDR uses industry standard data models and study data tabulation model (SDTM) to facilitate collaboration with internal and external partners. During SDTM defining phase, a business need was identified to collect additional data to support the analysis and reporting. Since SDTM does not allow the addition of new variables, it was necessary to represent the metadata and data for each non-standard variable in SUPPQUAL. In order to support safety and efficacy programming in an efficient manner, a team developed CDR views. The CDR views consist of standard derivations, phasing and vocabulary leveling across trials, which provides consistency across trials. Data extracted from CDR contains standard (pure) and non-standard (plus) variables, and these are called SDTM+ datasets.

Figure 1 shows the data sources for single analysis and reporting. SDTM+ SAS datasets extracted from our CDR will be the "raw" data source for single analysis and reporting. ADaM datasets (Analysis Data Model) will be the "derived" data source for single analysis and reporting. ADaM datasets will be derived from the combination of SDTM+ datasets and project specific logic. The single analysis and reporting design utilizes three types of ADaM datasets as defined below. The ADRSLICE and ADFIND datasets are internally defined datasets inspired by the Analysis Data Model.

- ADSL: subject level demographics dataset
- ADRSLICE: reporting events parameters dataset; utilized to "slice" data by time points
- ADFIND: generalized findings or events domain analysis dataset used for change from baseline, categorical, and time to event analyses

The ADSL, ADRSLICE, and ADFIND datasets will be created by standard SAS macros stored in a global macro library. The macros will utilize SDTM+ data and project specific logic to create ADSL, ADRSLICE, and ADFIND as permanent SAS datasets. These datasets, as well as the SDTM+ "raw" datasets, will be the only data source for all safety, efficacy, and immunogenicity analysis and reporting programs.

Figure 1
Data Sources for Single Analysis and Reporting



2.3 Structure for Analysis and Reporting Programs

Details about the analysis and reporting programs are shown in Figure 2. The structure for all analysis and reporting programs in the Single Analysis and Reporting design will follow a consistent look and feel with similar inputs and outputs. Each analysis and reporting program will utilize the following:

- SDTM+ and ADaM SAS V9 formatted datasets
- A common initialization macro which defines the programming environment by declaring autocall macro libraries, libnames, and filenames
- Heavy usage of modular utility, statistical analysis, reporting, and graphical macros from a global macro library
- A parameter dataset instead of analysis and reporting calling programs to pass parameter values to reporting macros

The analysis and reporting macros will create rich text formatted (RTF) report tables and possibly SAS datasets for further analyses.

Figure 2
Analysis and Reporting Programs

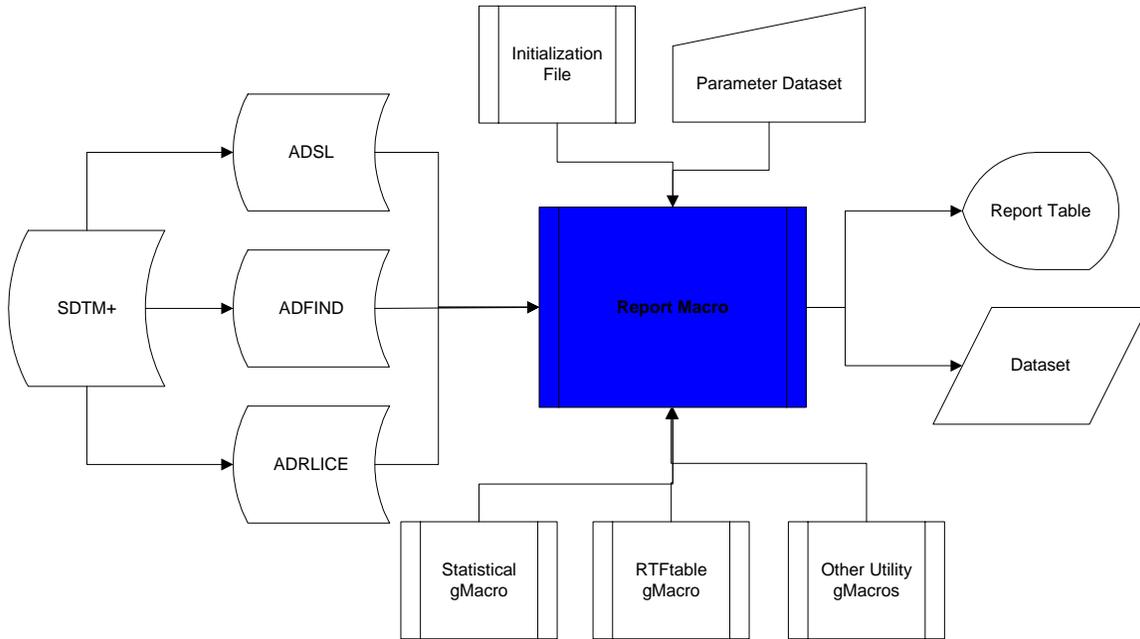
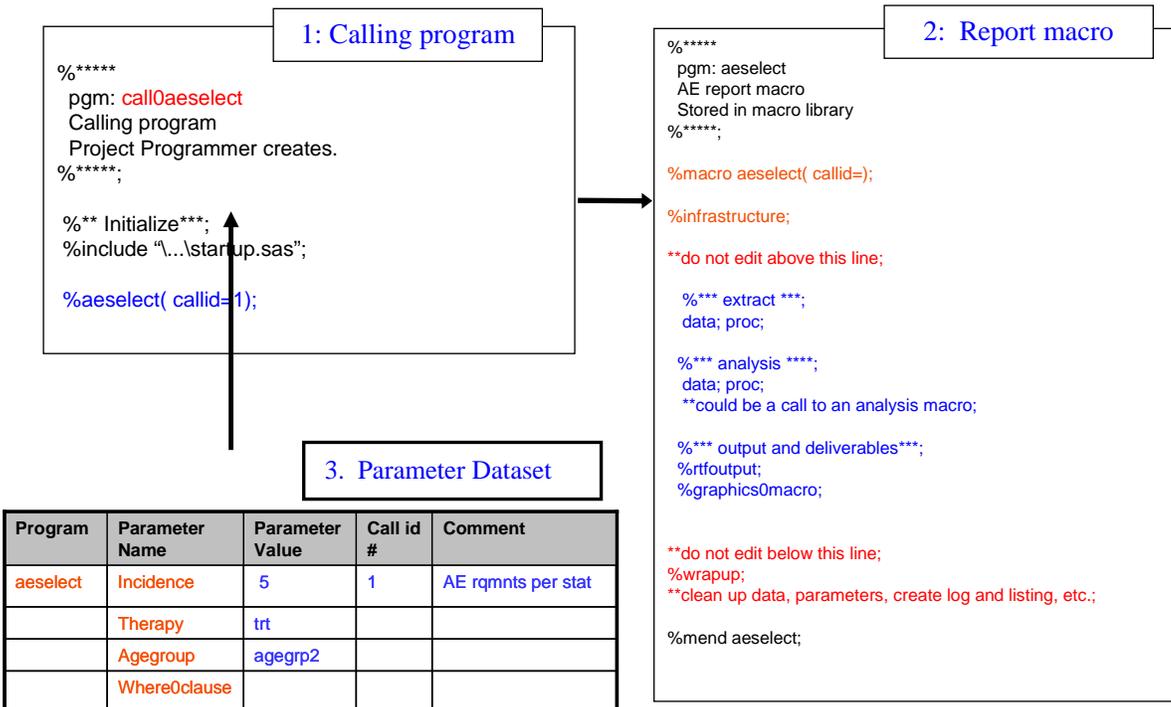


Figure 3 includes details about the calling program design. Each of the report macros will be executed using a calling program. The calling program will include a common initialization file, followed by a call to the report macro. The parameter values for the calling program will be stored in a SAS dataset.

Figure 3
Calling Program Design



3. CONCLUSION

The Single Analysis and Reporting design provides one common process for analysis and reporting. By utilizing one common process, analysis and reporting programs will be developed more efficiently, with higher quality, and more reusability.

4. TRADEMARK

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

5. ACKNOWLEDGEMENTS

The authors would like to acknowledge the following individuals who made valuable contributions to the work described in this paper as part of the Single Analysis & Reporting Team:

Frederic Coppin, Lisa Pyle, Bill Wilkins, John Troxell, Mark Tumelty, Fred Cieri, Christine Teng, Xingshu Zhu, Ginger Redner, Michael Senderak, Bipin Solanki, Carl Herremans, and Jennifer Shinomoto.

6. CONTACT INFORMATION

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

Amy Gillespie
Merck & Co., Inc.
P.O. Box 1000
Upper Gwynedd, PA 19454-2505
Work Phone: 267-305-6817
Email: amy_gillespie@merck.com

Mary Varughese
Merck & Co., Inc.
P.O. Box 1000
Upper Gwynedd, PA 19454-2505
Work Phone: 267-305-6891
Email: mary_varughese@merck.com