

SAS® Press Series

# **Validating Clinical Trial Data Reporting with SAS®**

**Carol I. Matthews  
Brian C. Shilling**





From *Validating Clinical Trial Data Reporting with SAS®*. Full book available for purchase [here](#).

---

## Contents

Preface ix

Acknowledgments xi

### Chapter 1 Pharmaceutical Industry Overview 1

1.1 Introduction 2

1.2 Regulations 2

1.2.1 Health Insurance Portability and Accountability Act 2

1.2.2 The Code of Federal Regulations 3

1.2.3 Guidance for Industry 4

1.2.4 International Conference on Harmonisation of Technical Requirements 5

1.2.5 Clinical Data Interchange Standards Consortium 6

1.3 Documentation 7

1.4 Standard Operating Procedures 7

1.4.1 Companywide Standard Operating Procedures 7

1.4.2 Department Standard Operating Procedures 8

1.4.3 Task Standard Operating Procedures 8

1.5 SAS Programming Guidelines 9

1.6 Quality Control versus Quality Assurance 9

1.7 Patient versus Subject 10

1.8 Conclusion 10

### Chapter 2 Validation Overview 11

2.1 Introduction 12

2.2 Validation versus Verification 12

2.3 Why Is Validation Needed? 13

2.3.1 Presenting Correct Information 13

2.3.2 Validating Early Saves Time 13

2.3.3 Developing a Positive Relationship 14

<b>2.4</b>	<b>How Do You Approach Validation?</b>	<b>14</b>
2.4.1	Start with All the Information	14
2.4.2	Have a Plan	15
2.4.3	Make the Code Do the Work	16
2.4.4	Ask Questions	16
2.4.5	Be Proactive	16
2.4.6	Validation Must Come First	17
<b>2.5</b>	<b>Validation Methods</b>	<b>17</b>
2.5.1	Independent Programming	18
2.5.2	Peer Review	19
<b>2.6</b>	<b>Validation Checklists</b>	<b>20</b>
<b>2.7</b>	<b>Software Development Life Cycle</b>	<b>21</b>
<b>2.8</b>	<b>Conclusion</b>	<b>22</b>

## **Chapter 3 Documentation and Maintenance 23**

<b>3.1</b>	<b>Introduction</b>	<b>24</b>
<b>3.2</b>	<b>Starting the Process</b>	<b>25</b>
3.2.1	Study Protocol	25
3.2.2	Annotated Case Report Form	26
3.2.3	Statistical Analysis Plan	29
3.2.4	Meeting Minutes	31
<b>3.3</b>	<b>Internal Program Documentation</b>	<b>32</b>
3.3.1	Program Header	32
3.3.2	Body Comments	34
3.3.3	Output Titles	35
<b>3.4</b>	<b>External Documentation</b>	<b>35</b>
3.4.1	Data Definition Tables	35
3.4.2	Program Directory	36
3.4.3	Validation Files	37
<b>3.5</b>	<b>Make Programs Maintainable</b>	<b>38</b>
3.5.1	Create and Follow Naming Conventions	38
3.5.2	Make It Easy to Read	38

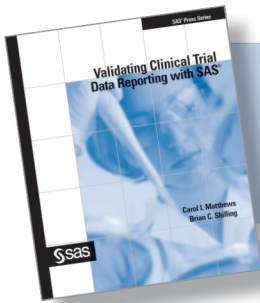
3.5.3	One Program, One Purpose	42
3.5.4	Comments, Comments, Comments	43
3.5.5	Use Macros Judiciously	44
3.6	Make Data Maintainable	44
3.6.1	Order Your Data	44
3.6.2	Label Everything	49
3.6.3	Attach Formats Sparingly	50
3.6.4	Consistency Is Key	51
3.6.5	Good Housekeeping	51
3.6.6	Look—but Don't Touch	53
3.7	Conclusion	56
Chapter 4	General Techniques to Facilitate Validation	57
4.1	Introduction	58
4.2	Validation Tools	58
4.2.1	Procedures	58
4.2.2	SAS Options and Language Elements	67
4.2.3	Using Macros Effectively	72
4.3	Techniques That Facilitate Validation	80
4.3.1	Start with a Clean Log	80
4.3.2	Print Only What You Need—When You Need It	81
4.3.3	Tracking Problems	82
4.3.4	Using PROC TRANSPOSE or an Alternative Solution	85
4.3.5	Tracking Dropped Data	89
4.4	Conclusion	93
Chapter 5	Data Import and Export	95
5.1	Introduction	96
5.2	Validating the Import Process	96
5.3	Validating the Export Process	98
5.4	General Items to Watch For When Transferring Data	99

<b>5.5 Working with SAS Files</b>	<b>100</b>
5.5.1 SAS Data Sets	100
5.5.2 SAS Transport Files	101
<b>5.6 Working with Other File Types</b>	<b>102</b>
5.6.1 Microsoft Excel Files	102
5.6.2 Flat Files	103
<b>5.7 Common Procedures Used for Validating Data Transfers</b>	<b>104</b>
5.7.1 PROC CONTENTS	104
5.7.2 PROC COMPARE	108
<b>5.8 Conclusion</b>	<b>112</b>

## **Chapter 6 Common Data Types 113**

<b>6.1 Introduction</b>	<b>114</b>
<b>6.2 Study Populations</b>	<b>114</b>
6.2.1 Safety	115
6.2.2 Intent-to-Treat	115
6.2.3 Efficacy	116
<b>6.3 Common Data Domains</b>	<b>116</b>
6.3.1 Subject Demographics	116
6.3.2 Inclusion/Exclusion Criteria	117
6.3.3 Subject Disposition	118
6.3.4 Medical History	118
6.3.5 Physical Examination	120
6.3.6 Vital Signs	120
6.3.7 Treatment Exposure	122
6.3.8 Concomitant Medications	123
6.3.9 Adverse Events	124
6.3.10 Clinical Laboratory Data	126
<b>6.4 Conclusion</b>	<b>137</b>

<b>Chapter 7 Reporting and Statistics</b>	<b>139</b>
7.1 Introduction	140
7.2 Pre-Output Validation Steps	140
7.2.1 Code Review	140
7.2.2 Log Review	141
7.3 Output Validation Steps	142
7.3.1 Understanding the Data	142
7.3.2 Understanding the Output	143
7.3.3 Checking the Result	143
7.3.4 Cross-Checking Related Output	146
7.3.5 Checking the Cosmetics	153
7.3.6 Updating the Specifications	157
7.3.7 Keeping What Is Important	157
7.4 Final QC Steps	158
7.5 Conclusion	158
<b>Appendix A Sample Quality Control Checklists</b>	<b>159</b>
<b>Appendix B Sample Statistical Analysis Plan</b>	<b>163</b>
<b>Appendix C Glossary</b>	<b>181</b>
<b>References</b>	<b>195</b>
<b>Index</b>	<b>197</b>



From *Validating Clinical Trial Data Reporting with SAS®*. Full book available for purchase [here](#).



# Chapter 1

## Pharmaceutical Industry Overview

---

- 1.1 Introduction 2
- 1.2 Regulations 2
  - 1.2.1 Health Insurance Portability and Accountability Act 2
  - 1.2.2 The Code of Federal Regulations 3
  - 1.2.3 Guidance for Industry 4
  - 1.2.4 International Conference on Harmonisation of Technical Requirements 5
  - 1.2.5 Clinical Data Interchange Standards Consortium 6
- 1.3 Documentation 7
- 1.4 Standard Operating Procedures 7
  - 1.4.1 Companywide Standard Operating Procedures 7
  - 1.4.2 Department Standard Operating Procedures 8
  - 1.4.3 Task Standard Operating Procedures 8
- 1.5 SAS Programming Guidelines 9
- 1.6 Quality Control versus Quality Assurance 9
- 1.7 Patient versus Subject 10
- 1.8 Conclusion 10

---

## 1.1 Introduction

The pharmaceutical industry, including clinical research organizations (CROs) and biotechnology companies, has adopted many industry standards and requirements. While these standards affect the entire clinical trial process, many have a direct impact on how SAS programmers work, and explain why validation is such a cornerstone of the programming process in this industry.

---

## 1.2 Regulations

There are many layers to the rules and regulations that govern the pharmaceutical industry. As a SAS programmer, you will be required to follow many of these regulations, which can be broken down into three major categories: federal laws, federal guidelines, and industry standards.

Federal laws (the Code of Federal Regulations) consist of legislation that is passed to control how things are done and how information is handled. Violation of these laws can lead to actions such as prosecution by the federal government. Federal guidelines are formal lists of suggestions that the federal government has issued to let the industry know the best way to conduct trials and submit the data in order to enable approval of a drug or device. These guidelines are simply that—guidelines. Unlike laws, failure to follow these guidelines does not carry as hefty a penalty, although it can lead the government to refuse to review a submission or approve a drug. Finally, with time and experience, companies have developed sets of standards that allow information and data to be shared more effectively. As the need for these industry standards has been recognized, organizations have been formed to determine the areas that need standards, to develop suitable standards, and to then document them to share information across companies.

The main source of information on industry standards and requirements is the Food and Drug Administration (FDA). Through various communication channels (primarily regulations and guidance documents published on the agency's Web site, [www.fda.gov](http://www.fda.gov)), the FDA defines the requirements and expectations for a New Drug Application (NDA). While many of the guidance documents and regulations that the FDA issues do not directly impact a SAS programmer's work, some do. Those most relevant to you are discussed here.

---

### 1.2.1 Health Insurance Portability and Accountability Act

As summarized by the U.S. Department of Labor ([www.dol.gov/dol/topic/health-plans/portability.htm](http://www.dol.gov/dol/topic/health-plans/portability.htm)), The Health Insurance Portability and Accountability Act of 1996 (HIPAA)

... provides rights and protections for participants and beneficiaries in group health plans. HIPAA includes protections for coverage under group health plans that limit exclusions for preexisting conditions; prohibit discrimination against employees and dependents based on their health status; and allow a special opportunity to enroll in a new plan to individuals in certain circumstances. HIPAA may also give you a right to purchase individual coverage if you have no group health plan coverage available, and have exhausted COBRA or other continuation coverage.

How does this impact you as a SAS programmer? It has little or no impact on day-to-day programming, but it is important to understand that the law exists and to have a general idea of its purpose. In simple terms, HIPAA serves to protect the information about a subject's identifying information. While this concept has only recently been so plainly articulated, it is the core reason that the most specific identifying information about each subject in every clinical trial conducted in the United States is limited to the subject's initials and date of birth. Any identifying information that is more specific is carefully protected by the investigating site. When validating data that may come to you as a programmer, it is important to understand that personal information should not be included—and if it is, it is your responsibility to point it out to have it removed.

---

### 1.2.2 The Code of Federal Regulations

Title 21 of the Code of Federal Regulations (CFR) pertains to food and drugs. Chapter 1 pertains to those components that identify the Food and Drug Administration (FDA) and the Department of Health and Human Services (DHHS). Within this set of regulations, Part 11, perhaps the most well-known and referenced section, specifically identifies electronic records and electronic signatures. It is important to note that any requirements listed under Title 21 in general are often referred to as *predicate rules*.<sup>1</sup> These rules can help determine when Part 11 rules apply to a specific situation, as well as how any aspect of a clinical trial is performed. On the subject of good clinical practice, 21 CFR 50, "Protection of Human Subjects," is one such predicate rule that requires clinical trial subjects to provide written informed consent to participate in a research trial. More indirectly, Part 820.70(i) addresses automated processes: "When computers or automated data processing systems are used as part of production or the quality system, the manufacturer shall validate computer software for its intended use according to an established protocol."<sup>2</sup> While this regulation directly applies to manufacturing, it is the predicate rule that is cited as the reason that SAS programs need to be validated. There are numerous topics within Title 21 that directly (Part 11 and Part 820) or indirectly (Part 50) affect programming. While you don't need to read each of these, it is helpful to understand what parts of the clinical trial and programming process are driven by these rules.

---

<sup>1</sup> [www.labcompliance.com/info/links/fda/regulations.aspx](http://www.labcompliance.com/info/links/fda/regulations.aspx)

<sup>2</sup> Code of Federal Regulations, Title 21, Volume 8; cite 21CFR820.70

Part 11 of this code contains several sections. Each section outlines the steps to take to ensure that the electronic records, electronic signatures, and handwritten signatures that are applied to electronic clinical data are truthful, dependable, and equal to paper records and handwritten signatures on paper. Most of these regulations are implemented and completed by IT professionals (those responsible for hardware and software installation, documentation, and maintenance). Most important to SAS programmers is the section that dictates how records can be modified: “Use of secure, computer-generated time-stamped audit trails to independently record the date and time of operator entries and actions that create, modify, or delete electronic records. Record changes shall not obscure previously recorded information.”<sup>3</sup> The key principal of this regulation is to understand that data cannot just be changed; a specific procedure must be followed. This regulation is the reason that programmers are not permitted to hard code data changes and why a key part of the validation process is ensuring that the result of a programming effort accurately represents the original data that it is based on.

While the FDA has narrowed the scope and application of this regulation, this does not mean that you can disregard these procedures while conducting clinical trials. The FDA is incorporating the general guidelines in this regulation into other regulations and guidance documents, specifically in the Guidance For Industry, Part 11, Electronic Records; Electronic Signatures—Scope and Application. In this document, the FDA clarifies that it has moved to a risk-based approach to this regulation. In it, the FDA “... recommend[s] that you base your approach [to validation] on a justified and documented risk assessment and a determination of the potential of the system to affect product quality and safety, and record integrity.” While most SAS programming in the pharmaceutical industry would be considered individual programs rather than systems, the general approach to all programs and the development of relevant standard operating procedures (SOPs) governing validation of those programs should take into account the FDA’s thinking on computerized systems.

---

### 1.2.3 Guidance for Industry

A series of guidance documents published by the FDA details how information from clinical trials should be submitted. One example of an older guidance document specifically pertaining to programming is *Providing Regulatory Submissions in Electronic Format—General Considerations*.<sup>4</sup> This guidance document provides some detail on how data sets should be structured and which file formats are acceptable. More recently, the FDA has encouraged use of electronic common technical documents (eCTDs) for submissions. See *Providing Regulatory Submissions in Electronic Format—Human Pharmaceutical Product Applications and Related Submissions Using the eCTD Specifications*.<sup>5</sup> This document references a separate guidance that is very relevant for programmers, titled “Study

---

<sup>3</sup> Federal Register, 21 CFR Part 11 – Subpart B §11.10 (e)

<sup>4</sup> [www.fda.gov/cber/gdlns/elecgen.htm](http://www.fda.gov/cber/gdlns/elecgen.htm)

<sup>5</sup> [www.fda.gov/cder/guidance/7087rev.htm](http://www.fda.gov/cder/guidance/7087rev.htm)

Data Specifications.”<sup>6</sup> As requirements change, the FDA issues these documents to notify the industry of what those changes are and how to comply with them.

For example, currently the FDA accepts data only as SAS Version 5 compatible transport files. This can be challenging at times because most companies now use SAS Version 8 or later. These versions offer much more flexibility and greater functionality than SAS Version 5; specifically, variable names can be longer than 8 characters, character variables can be larger than 200 bytes, and variable labels can be longer than 40 characters. However, due to SAS Version 5 compatibility restrictions, many of these data set features cannot be used. Until this restriction changes, programmers need to remain aware and work with data set structures prior to SAS Version 8 throughout the programming process so significant restructuring of data is not required later.

Another technical issue is the file size restrictions imposed by the FDA. At one time, the maximum file size allowed in a submission was 5 MB. Currently, the maximum file size is 100 MB, and while this may seem adequate for most types of data, keep this restriction in mind when designing all data sets. Unnecessary variables and duplication of information can push the limits of this restriction and cause future issues. While requirements may change over time, it is important to keep abreast of any such issues that could impact how you structure your programs and the output they create.

---

### 1.2.4 International Conference on Harmonisation of Technical Requirements

While the US FDA is the world’s leading drug approval agency, other countries also develop drugs and have agencies that regulate their approval. In a global setting, it is important for all parties involved in drug development to have a standard set of definitions for similar concepts and a common understanding for how drugs should be developed. This way, companies that develop drugs in one country under one set of rules can apply to have the same drug approved in other countries without having to redevelop it. If all countries have the same understanding of the rules, data developed elsewhere will follow a consistent set of rules. The International Conference on Harmonisation of Technical Requirements for Registration of Pharmaceuticals for Human Use (ICH) is a global organization that provides these common definitions and guidelines and is often a source for standard values for certain data (e.g., *country of origin*). *E6 Good Clinical Practice: Consolidated Guidance*<sup>7</sup> is one of the more general guidance documents published by ICH that defines many common terms (such as adverse drug reaction) and general guidance for how trials should be conducted (such as how safety data should be reported). *E9 Statistical Principles for Clinical Trials*<sup>8</sup> is a more narrow guidance that lays forth

---

<sup>6</sup> [www.fda.gov/cder/regulatory/ersr/Studydata.pdf](http://www.fda.gov/cder/regulatory/ersr/Studydata.pdf)

<sup>7</sup> [www.fda.gov/cder/guidance/959fnl.pdf](http://www.fda.gov/cder/guidance/959fnl.pdf)

<sup>8</sup> [www.fda.gov/Cder/guidance/ICH\\_E9-fnl.pdf](http://www.fda.gov/Cder/guidance/ICH_E9-fnl.pdf)

the general statistical principles that guide the development of complete programs (what types of studies should be conducted to support claims of safety and efficacy) and how individual studies should be designed (sample size, parallel group or crossover or other design, randomization/blinding, for example) and reported. While these guidance documents may not impact your programming responsibilities directly, they are part of the framework that built the studies and the specifications you work with regularly.

---

### **1.2.5 Clinical Data Interchange Standards Consortium**

The Clinical Data Interchange Standards Consortium (CDISC) is a team of industry professionals, including members from the FDA. According to CDISC ([www.cdisc.org](http://www.cdisc.org)), its mission is “to develop and support global, platform-independent data standards that enable information system interoperability to improve medical research and related areas of healthcare.”

In other words, the CDISC end product is a set of data standards that companies in the industry can follow to expedite filing a clinical trials outcome. Each module that CDISC delivers contains the structure, derivation rules, attributes, and components of the data that the FDA will receive. The goal is to achieve a standard set of data that the FDA needs to program only once. Consequent receipt of clinical data can then be analyzed using standard programming, and the review process can be expedited.

It is important for programmers to understand CDISC standards and to realize that CDISC actually has several standards. Two key sets of standards that affect the majority of clinical trial programmers are the Study Data Tabulation Model (SDTM) used for submitting data tabulations and the Analysis Data Set Model (ADaM) used for submitting analysis data sets. While these two sets of standards overlap in many areas, both have many distinct components that can effect how data is stored. Other standards are currently under development, so it is important to keep abreast of the most recent documentation.

While these standards are not yet a requirement, but rather a guideline, the FDA does recommend following them. Ultimately, the use of these standards will depend on your company's policies. These standards are quickly becoming industry standards, so implementing them is highly recommended. Regardless, having a set of standards for data collection and storage such as those provided by CDISC streamlines programming for the pharmaceutical company and expedites the review and approval process for the FDA. Once the CDISC standards have been completed, the FDA will probably adopt them as a requirement for submitting data. Getting to know the CDISC standards now and implementing those standards as much as possible will save time in the future.

---

## 1.3 Documentation

Another way that FDA requirements directly affect a SAS programmer's daily responsibilities is in the area of documentation. The term *documentation* refers to several things—both information that programmers work with and information that programmers provide. It can refer to the documents that are used to form the programming structures and ideologies within a company, including standard forms, guidelines, standard operating procedures, and other written guidance documents. It can also mean keeping hardcopy and electronic records of the process and results of programming. In addition, documentation can refer to keeping detailed flow information within a program itself to instruct other users of the purpose and methods used within the program.

All aspects of programming must be documented in one way or another. Documentation is an integral part of the programming process and provides the evidence that your programming efforts were effective. The documentation that is directly involved in programmers' day-to-day activities is discussed in detail in a later chapter. The documentation that is standard for the industry and forms the framework for how programmers perform their job functions, including the requirements for validation, is discussed below.

---

## 1.4 Standard Operating Procedures

One key set of documents required by the FDA is standard operating procedures (SOPs). SOPs are documents that describe procedures to follow for a specific operation or task. They detail all aspects of working in the pharmaceutical industry from high-level SOPs (such as defining the process for creating and/or modifying SOPs) to lower-level SOPs (such as defining each step to be followed while programming, validating, and delivering SAS programming output). SOPs may be created for several different levels of clinical trial programs.

In general, if a process is listed or mentioned in the CFR, then there will be an SOP that outlines the process. While following these CFR-related SOPs is required, following other procedures outlined in SOPs (as opposed to guidelines or no guidance at all) is up to the individual company. It is important for programmers to know which SOPs directly influence how their jobs are performed. There are several categories of SOPs that can affect programming processes.

---

### 1.4.1 Companywide Standard Operating Procedures

Each pharmaceutical company or clinical research organization (CRO) creates and maintains standard operating procedures for the daily functioning of its business. These high-level SOPs usually contain general company operating guidelines followed by every employee. Typically, they identify:

- company operating structure
- document handling
- employee training
- physical business information

---

### **1.4.2 Department Standard Operating Procedures**

Each pharmaceutical company or CRO also creates and maintains standard operating procedures for the daily functioning of its individual departments. Programmers are trained in these detailed SOPs, which typically identify:

- using SAS programming standards or guidelines
- computer system structure, usage, and permissions
- randomization scheduling and programming
- blinding and unblinding procedures

---

### **1.4.3 Task Standard Operating Procedures**

Sometimes programmers must perform job tasks that need to be described in more detail than company and department standard operating procedures. In most cases, a SAS programming department creates task-level SOPs to outline standard procedures for dealing with these varying tasks. Task-level SOPs normally identify procedures to follow to accomplish programming in the following areas:

- importing data
- validating derived or analysis data
- validating summary tables and figures
- exporting of data and/or reports
- studying drug compliance

Each company's SOPs structure and layout may differ, but they all accomplish the same task: creating a standard, structured, and controlled set of procedures for all employees to follow. These standards ensure that tasks are completed consistently and with a similar level of quality. SOPs often specify checklists that include the individual processes that need to be followed to ensure a consistent level of quality. For example, an SOP that details how the validation of data set programs is performed may also have a checklist to

be completed for every program that creates a data set. That checklist may include items such as:

- ensure all variables detailed in the specification are included in the data set
- ensure that numeric variables are rounded correctly and per specification
- ensure that values in character variables are not truncated
- check a sample of derived variable values against source data to ensure correct derivation

It is important to know whether your company has SOPs governing validation and what these SOPs include. If they are available, following validation SOPs will help to ensure that each programmer produces the same quality of output.

---

## 1.5 SAS Programming Guidelines

Standard operating procedures are normally written as an overview or on a very general level. This generality avoids the need to change the SOPs frequently, when minor details need to change. Because SOPs must be approved by several levels of management and controlled through a document management system, frequent changes become time-consuming and problematic. To avoid making multiple changes to the programming SOPs, SAS programming guidelines are created. These guidelines serve as a more detailed set of instructions for programmers to follow to maintain a consistent program structure and methodology for performing common tasks. The guidelines often outline program structure (headings, comments, white space, and compute blocking, for example), standard calculation formulas, methods for validation, and how to handle deviations from the SOPs. Programming guidelines are often the key to providing consistency between members of a programming team.

Because programming guidelines are not as tightly controlled as SOPs, they allow for more flexibility and change. When a version of SAS changes, operating systems change, or other changes are made, the guidelines can easily be updated, distributed, and taught.

---

## 1.6 Quality Control versus Quality Assurance

Quality control (QC) and quality assurance (QA) are important parts of a clinical trials environment. They act to maintain standards and excellence in completing a successful trial. Quality control is defined as “an aggregate of activities (as design analysis and inspection for defects) designed to ensure adequate quality especially in manufactured

products.”<sup>9</sup> Quality assurance is defined as “a program for the systematic monitoring and evaluation of the various aspects of a project, service, or facility to ensure that standards of quality are being met.”<sup>10</sup>

The main difference between QA and QC is that QC is performed within each department. For programmers, QC is maintained using standards and documentation (for example, standard operating procedures and SAS programming guidelines). QC occurs when a programmer checks his or her own output (for example, printing observations from a data set before and after manipulation and then comparing the results) and when two programmers within the same department independently produce output and then compare the results.

On the other hand, QA is performed by an independent group outside of the programming department. In the pharmaceutical industry, this is typically the Regulatory Department. In some companies, this department also has SAS programmers who independently try to replicate the results produced by the programmers in other departments. The Regulatory Department is well-versed in the requirements of both FDA and federal law and will scrutinize all of the clinical trial’s output that comes from the company to make sure it is in compliance with these requirements.

---

## 1.7 Patient versus Subject

For as long as the industry has been thriving, there has been an ongoing debate about what terminology to use to refer to the participants of clinical trials. In the beginning, the term *patient* was used. As clinical trials became more involved and started going through developmental cycles, the term *subject* was used because many of the trials were being conducted on healthy participants. For consistency, we use the term *subject* to refer to all participants in clinical trials throughout this book.

---

## 1.8 Conclusion

There are many rules, regulations, and guidelines that affect a programmer’s work and govern the validation process. It is helpful to understand the source of these rules so that any changes are easier to follow. Often these rules can be subject to interpretation. When you are making validation policy decisions, it can be important to refer to the original documentation rather than relying on secondary sources. Detailed sources of information are available for many of the topics discussed in this chapter. Refer to the References section for details. Now that the basis for validation has been established, we can discuss more specific topics that directly influence SAS programming.

<sup>9</sup> [www.m-w.com/dictionary/quality%20control](http://www.m-w.com/dictionary/quality%20control) (Merriam-Webster’s Online Dictionary)

<sup>10</sup> [www.m-w.com/dictionary/quality%20assurance](http://www.m-w.com/dictionary/quality%20assurance) (Merriam-Webster’s Online Dictionary)

From *Validating Clinical Trial Data Reporting with SAS®* by Carol I. Matthews and Brian Shilling. Copyright © 2008, SAS Institute Inc., Cary, North Carolina, USA. ALL RIGHTS RESERVED.



From *Validating Clinical Trial Data Reporting with SAS®*. Full book available for purchase [here](#).

# Index

## A

- aCRFs (annotated CRFs) 26–29
- action taken upon adverse events 126
- ADaM (Analysis Data Set Model) 6
- adverse events 26, 124–126
  - cross-checking (example) 148–152
  - dictionaries for 124
  - outcomes 126
  - seriousness of 125
- alert version, documenting 33
- alignment in output presentation 154
- Analysis Data Set Model (ADaM) 6
- analysis data sets 114
- annotated CRFs (aCRFs) 26–29
- archiving incoming data 97
- archiving validation process 157–158
- ASCII files 103
- asking questions 16
- attitude regarding validation 14
- AXIS1 and AXIS2 statements 133, 135

## B

- baseline values for vital signs 121
- body comments 34–35, 43
- body systems 119, 120
- box plots 132
- BY statements, PRINT statement with 61

## C

- capitalization in program code 41–42
- case in program code 41–42
- case report forms (CRFs) 26–29, 96
  - annotated (aCRFs) 26–29
- categorical data, validating 64–67
- CDISC (Clinical Data Interchange Standards Consortium) 6
- cell index 61–63
- CFR (Code of Federal Regulations) 3–4
- checklists for validation 20

- chemistry data 126
- clarity of program code 38–44
- Clinical Data Interchange Standards Consortium (CDISC) 6
- clinical laboratory test data 26, 126–128
- code documentation 32–35
- Code of Federal Regulations (CFR) 3–4
- code review 140–141
  - See also* validation
- coding medication names 123
- column alignment 154
- combining data 82
- comma-delimited files 103
- comments
  - flagging problem data 83–85
  - in code 34–35, 43
  - on studies 26
  - tracking dropped data 89–93
  - variable and data set labels 49–50
- company-wide SOPs 7–8
- COMPARE procedure
  - for validation 108–112
  - ID statement 112
- COMPARE statement 44–49
  - LISTALL option 47
- comparing versions of incoming data 98
- compatible transport files 5
- compliance of subjects 26
- concomitant medications 26, 123
- consent status 118
- consistency in programming 51
- CONTENTS procedure 26–28, 35
  - for validation 104–108
  - ordering data with 104–108
  - POSITION option 104
  - requesting output from 97
  - VARNUM option 104
- continuous variables, validating values of 65–67
- converting between file formats 97
- converting units, validating 121, 128–129

COPY procedure, creating transport files with 101–102  
 correctness of information, importance of 13  
 cosmetics of output, checking 153–157  
 CPORT procedure, creating transport files with 101–102  
 CRFs (case report forms) 26–29, 96  
   annotated (aCRFs) 26–29  
 criteria for inclusion/exclusion 26, 117  
 cross-checking related output 146–152  
 CSV files 103

## D

data, understanding 143  
 data definition tables (DDTs) 35–36  
 data domains 116–137  
   adverse events 26, 124–126, 148–152  
   clinical laboratory test data 26, 126–128  
   concomitant medications 26, 123  
   inclusion and exclusion criteria 26, 117  
   medical history 26, 118–120  
   physical examination 26, 120  
   subject demographics 26, 116–117, 146–148  
   subject disposition 26, 118  
   treatment exposure 122–123  
   vital signs 26, 120–122  
 data export process  
   *See* export process  
 data import process  
   *See* import process  
 data listing programs 42–43  
 data maintenance 44–55  
 data management system variables 26–28  
 data ordering 44–49  
   with CONTENTS procedure 104–108  
 data presentation 153–157  
 data rounding, checking 146  
 data sets 100–101  
   deleting records from 89–90  
   displaying selected records 58–63, 81–82  
   for analysis 114  
   hard-coding data 53–55

  labeling 49–50  
   merging 64–67, 69–71, 87, 116–117  
   naming conventions for 38  
   programs for creating 42–43  
   sorting records in 44–49  
   subject demographics 26, 116–117, 146–148  
   tracking problematic data sets 82–83  
   transposing data for analysis 85–88  
   validating input for selected records 98  
 data sets, exporting  
   *See* export process  
 data sets, importing into  
   *See* import process  
   *See* import programs  
 DATA step  
   combining data with 82  
   dropping duplicate records 90  
   ending 42  
   transposing data with 85–88  
 data subsets  
   displaying 58–63, 81–82  
   validating input 98  
 data summaries  
   *See* reporting  
 data traceability 99  
 data transfers  
   *See* export process  
   *See* import process  
 data transposition 85–88  
 data types 114–137  
   *See also* data domains  
   *See also* variables  
   applying normal ranges 121, 128, 132–137  
   data transfers, issues with 99  
   Excel files, issues with 102–103  
   for study populations 114–116  
   standardizing units 128–132  
 data unit standardization 128–132  
 data validation  
   *See* validation

## dates

- adverse events 124–125
- data transfers 99
- discontinuation dates 26, 118
- imputing dates for medical history 120
- medication start and stop dates 123
- with Excel files 102–103
- with treatment exposure 122

DDTs (data definition tables) 35–36

## deleting data

- when duplicated 90–93
- when unnecessary 89–90

delimited text files 103

demographics data 26, 116–117

- cross-checking (example) 146–148

department SOPs 8

design phase (SDLC) 21

## dictionaries

- for adverse events 124
- for medical history events 119, 120
- for medications 123
- MedDRA 119, 124
- WHODRL 123

directories of programs 36–37

discontinuation dates 26, 118

disposition 26, 118

documentation 7, 24–37

- alert version 33
- assembling key specifications 25–32
- external 35–37
- internal program documentation 32–35
- list of data domains 116–137
- needed for validation 14–15
- of export process 99
- of validation 14–15, 24, 37
- validating 20
- what to keep 157–158

## domains of data

*See* data domains

DROP statements 53

dropped data, tracking 89–93

duplicated records, dropping 90–93

DUPOUT= option, SORT procedure 92

**E***E6 Good Clinical Practice* 5*E9 Statistical Principals for Clinical Trials* 5–6

eCTDs (electronic common technical documents) 4–5

efficacy population flags 116

efficiency through validation 13, 16

electronic common technical documents (eCTDs) 4–5

electronic dictionaries

*See* dictionaries

## errors

*See also* validation

discovering early 13

reviewing 141–142

event dictionaries 119, 120

examination 26, 120

Excel files 102–103

exclusion criteria 26, 117

## export process

*See also* output validation

documentation of 99

reproducibility of data transfers 99

validating 98–100, 103

what to watch for 99–100

with ASCII files 103

Export Wizard 98

exposure to study treatment 122–123

external documentation 35–37

**F**

FDA (Food and Drug Administration) 2, 6

files sent to 101–102

guidance documents 4–5

industry guidelines 4–5

federal laws in pharmaceutical industry 2

figure creation programs 42–43

file content issues 99–100

file formats (file types)  
     converting between 97  
     delimited 103  
     flat files 103  
     incoming data 96–97  
     what to watch for 99–100  
 file size restrictions 5  
 filename conventions 101  
 files, SAS 100–102  
 first-level validation 18  
 flags  
     efficacy population flags 116  
     flagging problem data 83–85  
 flat files 103  
 Food and Drug Administration  
     *See* FDA  
 FOOTNOTE statement 136  
 FORMAT procedure 65–67  
 formats  
     checking with MEANS procedure 146  
     user-defined 50–51, 101  
 formatted values, checking 146  
 formatting of program code 38–44  
 FREQ procedure, for validation 64–67, 129,  
     144–145

## G

good clinical practice 3–4  
 GPLOT procedure 136  
     PLOT statement 136  
 GUESSINGROWS option, IMPORT  
     procedure 99  
 guidance documents, FDA 4–5

## H

hard-coding data 53–55  
 header, program 32–33  
 Health Insurance Portability and  
     Accountability Act (HIPPA) 2–3  
 hematology data 126  
 HIPAA (Health Insurance Portability and  
     Accountability Act) 2–3

histograms 131  
 history, subject 26, 118–120  
 HSIZE graphing option 136  
 human subjects, protection of 3–4

## I

ICH 5–6  
 ID statement  
     COMPARE procedure 112  
     PRINT statement with 61  
     TRANPOSE procedure 85  
 implementation phase (SDLC) 21  
 IMPORT procedure, GUESSINGROWS  
     option 99  
 import process  
     archiving incoming data 97  
     comparing versions of incoming data 98  
     file formats 96–97  
     from Excel or ASCII files 102–103  
     reproducibility of data transfers 99  
     validating 96–98, 102–103  
     what to watch for 99–100  
 import programs 97  
 Import Wizard 97  
 imputing dates  
     for adverse events 124–125  
     for medical history 120  
     medication start and stop dates 123  
 IN= system option 67, 68, 82  
 inclusion criteria 26, 117  
 incoming data  
     *See* import process  
 incorrect information, avoiding  
     *See* validation  
 indentations in program structure 40  
 independent programming (validation)  
     18–19  
 industry guidance, FDA 4–5  
 information, importance of correctness 13  
 informed consent status 118  
 intent-to-treat population 115–116  
 internal program documentation 32–35

International Conference on Harmonisation on  
Technical Requirements for  
Registration of Pharmaceuticals for  
Human Use 5–6

interval for treatment exposures 122

ITT (intent-to-treat) population 115–116

## K

key specifications 25–32

## L

LABEL statement 50

labeling variables 49–50

laboratory test data 26, 126–128

laws in pharmaceutical industry 2

layout of program code 38–44

legibility of program code 38–44

%LET identifier 59

LISTALL option, COMPARE statement 47

listing programs 42–43

logs 20, 69

after merging data sets 68

clean, starting with 80

notes in 80, 141–142

reviewing 141–142

LVREF= option, PLOT statement (GPLOT)  
136

## M

macros

effective use of 72–73

for general use 22

judicious use of 44

MLOGIC system option for validation  
79

MPRINT system option for validation  
73–76

SYMBOLGEN system option for  
validation 76–78

validating 73–79, 81

maintenance

of data 44–55

of programs 38–44

maintenance phase (SDLC) 21

maximum file size 5

MEANS procedure

to check formats 146

to check truncation 145

measurement unit standardization 128–132

MedDRA dictionary 119, 124

medical history 26, 118–120

medications

coding names of 123

dictionaries for 123

start and stop dates 123

meeting minutes 31–32

merging data sets 64–67

demographics data 116–117

log and 68

merging data to itself 69–71, 87

Microsoft Excel files 102–103

minutes of meetings 31–32

misspellings 153

MLOGIC system option 79

modification information for programs 33

MPRINT system option 73–76

MSGLEVEL= system option 67–69

## N

N option, PRINT statement 61

names of medications, coding 123

naming conventions

data sets 38

filenames 101

output files 38

programs 38

transport files 101

variables 38

NDA (New Drug Application) 2

NOBYLINE graphing option 135

NODATE graphing option 135

NODUPKEY option, SORT procedure  
90–93

NODUPREC option, SORT procedure  
90–93

NOGFOOTNOTE option, ODS statement 136  
 NOGTITLE option, ODS statement 136  
 NONUMBER graphing option 135  
 normal probability plots 132  
 normal ranges for data 121, 128, 132–137  
 notes in logs  
     removing 80  
     reviewing 141–142  
 numeric data, truncated 145

## O

OBS= option, PRINT procedure 35  
 ODS statement  
     for applying normal ranges 132, 136  
     NOGFOOTNOTE option 136  
     NOGTITLE option 136  
 one-off programs, validating 22  
 ORDER BY statement 44–46  
 ordering data 44–49  
     with CONTENTS procedure 104–108  
 ORIENTATION= graphing option 136  
 outcomes of adverse events 126  
 outgoing data  
     *See* export process  
 output files, naming 38  
 output from CONTENTS and PRINT procedures 97  
 output titles 35  
 output validation 142–158  
     *See also* validation  
     checking results 143–146  
     cross-checking related output 146–152  
     pre-output validation 140–142  
     presentation cosmetics 153–157  
     understanding data and output 142–143  
     validating export process 98–100, 103  
     what documentation to keep 157–158

## P

page breaks in output 154–157  
 Part 11 rules (CFR) 3–4  
 patients versus subjects (terminology) 10

peer review (validation) 19–20  
 pharmaceutical industry 2–10  
     regulation in 2–6  
 physical examination 26, 120  
 planning the validation process 15  
 PLOT statement, GPLOT procedure 136  
     LVREF= option 136  
     VREF= option 136  
 PLOT statement, UNIVARIATE procedure 131–132  
 plots  
     box plots 132  
     histograms 131  
     normal probability plots 132  
 populations  
     checking population counts 148  
     data types for 114–116  
     efficacy population flags 116  
     intent-to-treat (ITT) 115–116  
 POSITION option, CONTENTS procedure 104  
 positive relationships through validation 14  
 pre-output validation 140–142  
 predicate rules 3  
 presentation cosmetics, checking 153–157  
 PRINT procedure 35  
     appropriate use of 81–82  
     displaying data subsets with 58–63, 81–82  
     OBS= option 35  
     requesting output from 97  
 PRINT statement  
     BY statements with 61  
     ID statement with 61  
     N option 61  
 proactive validation 16–17  
 probability plots, normal 132  
 problematic data sets 82–83  
 PROC steps, ending 42  
 procedural validation 58–67  
 program code  
     case in 41–42  
     clarity of 38–44

- documentation 32–35
- formatting 38–44
- review 140–141
- program comments 34–35, 43
- program directories 36–37
- program header 32–33
- program logs
  - See* logs
- program maintenance 38–44
- program modification information 33
- program naming 38
- program types 42–43
- programming, independent 18–19
- programming consistency 51
- programming guidelines 9
- programming specifications 24
  - assembling 25–32
  - updating 157
  - validation and 14–15
- proof of validation 14, 24
  - validation files 37
- protection of human subjects 3–4
- protocols 25

## Q

- QA (quality assurance) 9–10
- QC (quality control) 9–10, 158
- questions, asking 16
- QUIT statements 42

## R

- randomization status 118
- ranges for data 121, 128, 132–137
- readability of program code 38–44
- records
  - deleting from data sets 89–90
  - dropping duplicates 90–93
  - sorting within data sets 44–49
- records management 4
- regulation in pharmaceutical industry 2–6
- related output, cross-checking 146–152
- removing duplicated records 90–93

- reporting 140–158
  - final QC 158
  - output validation 142–158
  - pre-output validation 140–142
- reproducibility of data transfers 99
- requirements phase (SDLC) 21
- resources for validation, obtaining 16–17
- results checking 143–146
- reviewing SAS code and logs 140–142
  - See also* validation
- ROUND function 146
- rounding accuracy, checking 146
- rules in pharmaceutical industry 2–6
- RUN statements 42

## S

- safety population data types 115
- SAPs (statistical analysis plans) 29–31
- SAS alert version, documenting 33
- SAS code review 140–141
- SAS data sets
  - See* data sets
- SAS Export Wizard 98
- .SAS file, reviewing 140–141
- SAS files 100–102
- SAS/GRAPH, for validation 132–136
- SAS Import Wizard 97
- SAS logs
  - See* logs
- SAS notes, reviewing 141–142
- SAS programming guidelines 9
- SAS statement structure 39–40
- SAS transport files 101–102
- SAS Version 5 compatible transport files 5
- SAS versions 100–101
- .SAS7BDAT files 101
- scheduling time for validation 16–17
- .SD2 files 100
- SDLC (software development life cycle) 21–22
- SDTM (Study Data Tabulation Model) 6
- second-level validation 18–19

- section breaks in program code 43
- separate programming 18–19
- single-use programs, validating 22
- software development life cycle (SDLC) 21–22
- SOPs (standard operating procedures) 7–9
- SORT procedure
  - DUPOUT= option 92
  - NODUPKEY option 90–93
  - NODUPREC option 90–93
  - to remove duplicates 90–93
- sorting data
  - See* ordering data
- specifications
  - See* programming specifications
- spelling mistakes 153
- spreadsheets, Excel 102–103
- SQL procedure for ordering data 44–49
- standard operating procedures (SOPs) 7–9
- standardizing units 128–132
- start and stop dates for medications 123
- statement structure 39–40
- statistical analysis plans (SAPs) 29–31
- statistics
  - summary statistics 114, 143, 146–152
  - understanding 143
- structuring program code 38–44
- Study Data Tabulation Model (SDTM) 6
- study disposition 26, 118
- study populations, data types for 114–116
- study protocols 25
- study treatment, exposure to 122–123
- subject compliance 26
- subject demographics 26, 116–117
  - cross-checking (example) 146–148
- subject disposition 26, 118
- subject examination 26, 120
- subject exclusion criteria 26, 117
- subject inclusion criteria 26, 117
- subject medical history 26, 118–120
- subject populations, data types for 114–116
- subject protection 3–4
- subjects versus patients (terminology) 10

- subsets of data
  - displaying 58–63, 81–82
  - validating input 98
- summarizing data
  - See* reporting
- summary statistics 114, 143
  - cross-checking related output 146–152
- SYMBOL1 and SYMBOL2 statements 133, 135
- SYMBOLGEN system option 76–78
- system options for validation 79
  - IN= 65–67
  - MLOGIC 79
  - MPRINT 73–76
  - MSGLEVEL= 65–67
  - SYMBOLGEN 76–78

## T

- tab-delimited files 103
- table creation programs 42–43
- task-level SOPs 8–9
- temporary variables 51–53
- test data 26, 126–128
- test ranges, normal 121, 128, 132–137
- testing phase (SDLC) 21
- text files 103
- time savings through validation 13, 16
- TITLE statement 136
- titles for output 35
- TLFs
  - annotated CRFs and 28–29
  - statistical analysis plans and 29–31
- traceability of data 99
- tracking dropped data 89–93
- tracking problems (validation) 82–85
- transferring data
  - See* export process
  - See* import process
- transformation of units, validating 121, 128–129
- transport files 5, 101–102
- TRANSPOSE procedure 70–71, 85–88
  - ID statement 85

transposing data for analysis 85–88

treatment exposure 122–123

truncation

checking for 144–145

issues with data transfers 99–100

of numeric data 145

## U

unit conversions, validating 121, 128–129

unit standardization 128–132

U.S. Food and Drug Administration

*See* FDA

UNIVARIATE procedure

for validation 129–132

PLOT statement 131–132

unnecessary data 89–90

updating specifications 157

user-defined formats 50–51, 101

## V

validation 12–22

*See also* output validation

archiving validation process 157–158

attitude regarding 14

categorical data 64–67

checklists 20

COMPARE procedure for 108–112

CONTENTS procedure for 104–108

documenting 14–15, 24, 37

efficiency through 13, 16

export process 98–100, 103

FDA guidelines for 4, 9

first-level 18

FREQ procedure for 64–67, 129, 144–145

how to approach 14–17

import process 96–98, 102–103

independent programming 18–19

methods for 17–20

normal ranges, applying 121, 128, 132–137

obtaining resources for 16–17

of documentation 20

of macros 73–79, 81

of unit conversions 121, 128–129

of values for continuous variables 65–67

one-off programs 22

peer review 19–20

planning 15

positive relationships through 14

pre-output validation 140–142

proactive 16–17

procedural 58–67

programming specifications and 14–15

proof of 14, 24, 37

reasons for 13–14

SAS/GRAPH for 132–136

scheduling time for 16–17

second-level 18–19

single-use programs 22

software development life cycle (SDLC) 21–22

study population data 114–116

subsets of data 98

system options for 65–67, 73–79

techniques for facilitating 80–93

time savings with 13, 16

tools for 58–79

tracking problems 82–85

verification versus 12–13

validation files 37

variables

*See also* data types

confirming inclusion in data transfers 100

continuous, validating values of 65–67

data management system variables 26–28

formatted values, checking 146

in study populations 114–116

labeling 49–50

naming conventions for 38

ordering 44–49

temporary, cleaning up 51–53

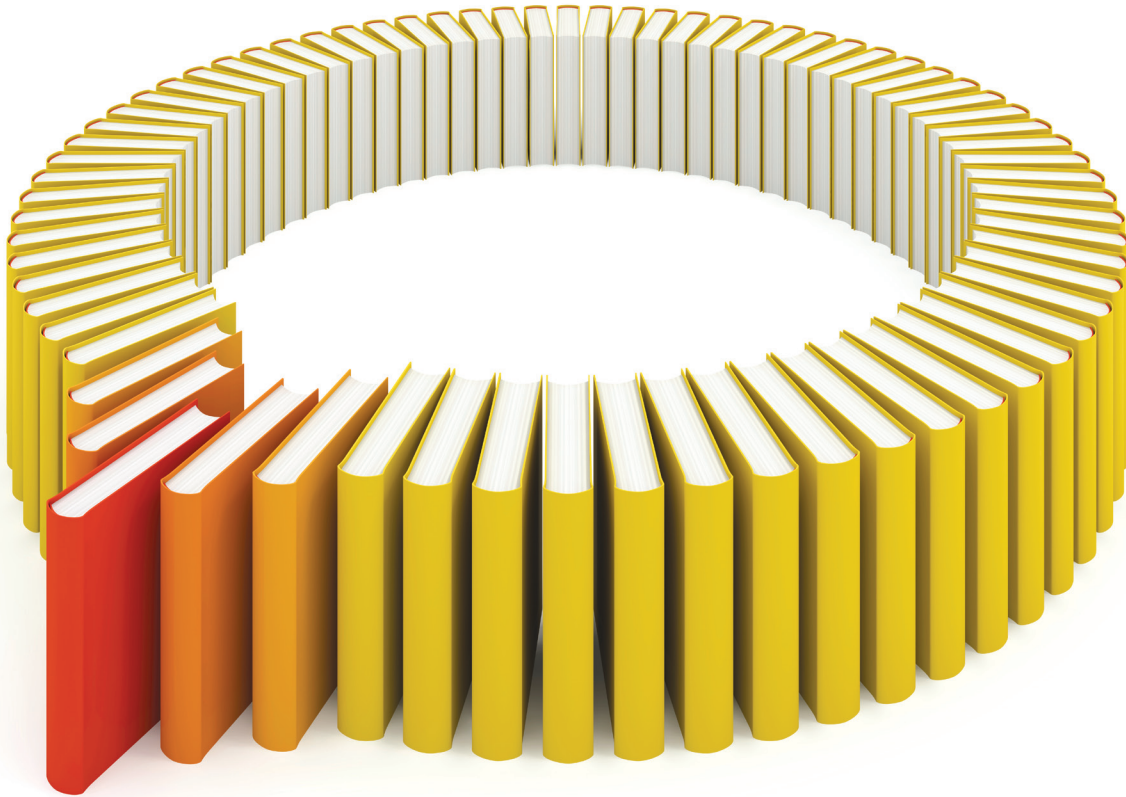
UNIVARIATE procedure for 129–132

VARNUM option, CONTENTS procedure  
104  
verification, versus validation 12–13  
Version 5 compatible transport files 5  
versions of incoming data, comparing 98  
versions of SAS 100–101  
vital signs 26, 120–122  
VREF= option, PLOT statement (GPLOT)  
136  
VSIZE graphing option 136

## **W**

warnings in logs, reviewing 141–142  
white space in program code 41  
WHODRL dictionary 123

From *Validating Clinical Trial Data Reporting with SAS®* by Carol I. Matthews and Brian Shilling. Copyright © 2008, SAS Institute Inc., Cary, North Carolina, USA. ALL RIGHTS RESERVED.



# Gain Greater Insight into Your SAS<sup>®</sup> Software with SAS Books.

Discover all that you need on your journey to knowledge and empowerment.

