Chapter 1

Pharmaceutical Industry Overview

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), 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), 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.*[1] 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."[2] 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.

[1] www.labcompliance.com/info/links/fda/regulations.aspx

[2] 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.”[3] 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.*[4] 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*.[5] This document references a separate guidance that is very relevant for programmers, titled “Study

[3] Federal Register, 21 CFR Part 11 – Subpart B §11.10 (e)

[4] www.fda.gov/cber/gdlns/elecgen.htm

[5] www.fda.gov/cder/guidance/7087rev.htm

Data Specifications."[6] 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*[7] 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*[8] is a more narrow guidance that lays forth

[6] www.fda.gov/cder/regulatory/ersr/Studydata.pdf

[7] www.fda.gov/cder/guidance/959fnl.pdf

[8] 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), 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."[9] 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."[10]

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.

[9] www.m-w.com/dictionary/quality%20control (Merriam-Webster's Online Dictionary)

[10] www.m-w.com/dictionary/quality%20assurance (Merriam-Webster's Online Dictionary)