

CHAPTER 1

AN OVERVIEW OF STATISTICAL QUALITY CONTROL AND SAS® STATISTICAL QUALITY CONTROL SOFTWARE

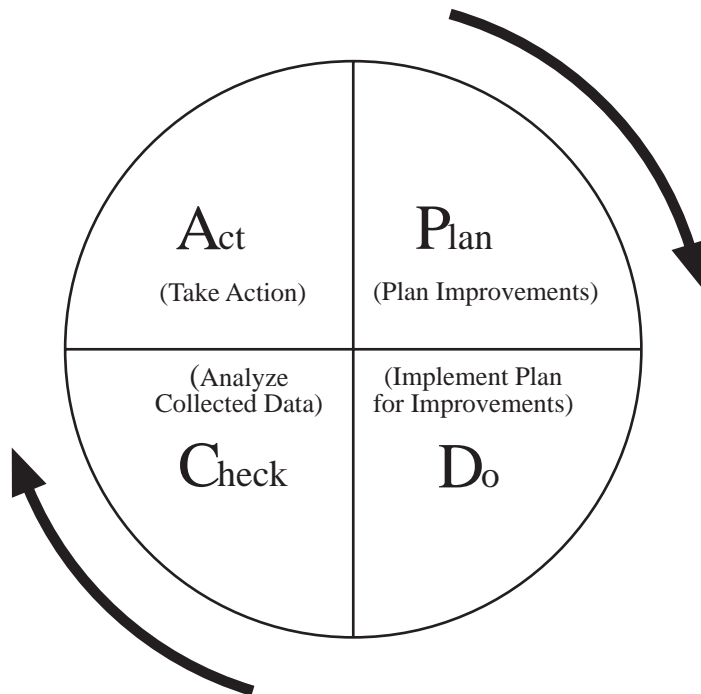
INTRODUCTION TO STATISTICAL QUALITY CONTROL

Quality! As the United States economy becomes part of a global economy, we hear this word spoken by management in all types of industries. Product manufacturers and service industries have realized that competition in a global market will require a continual and committed effort towards the improvement of product and service quality.

No organization can succeed in providing a quality product without an overall effort from its members. Throughout this book, there are references to *process improvement teams*. The process improvement (PI) team is a group of individuals contributing their knowledge to the improvement of the process. The PI team consists of individuals at all levels of management whose jobs affect or are affected by the process. Their knowledge of the process and how it interacts with the overall goals of the company is essential if the process is to be improved.

The Plan-Do-Check-Act (PDCA) Cycle, developed by quality expert W. Edwards Deming, is the blueprint through which teams of employees improve processes. Figure 1.1 illustrates the PDCA cycle.

Figure 1.1 *The Plan-Do-Check-Act Cycle*



The PDCA cycle consists of four phases. Each phase consists of a specific task in the quality improvement cycle. The phases are:

- Plan The PI team is brought together to define what improvements are to be made in the process and how those improvements will be implemented.
- Do The plan for improvement is executed.
- Check The collected data are analyzed using quantitative tools.
- Act The PI team takes action on variation identified by the data analysis.

The execution of these four phases by the PI team is critical to obtaining improvement in a process. Sustained, decisive improvement may involve many repetitions of the PDCA cycle.

QUALITY IN A VARIETY OF ORGANIZATIONS

The quality movement originated in manufacturing environments where its focus was on improving the quality of durable goods produced in factories. In recent years, quality improvement efforts have spread to other types of companies, two of which are medical/analytical laboratories and service industries.

Due to the increasing demand for reliable laboratory results in tests for diseases and food/water contamination, the area of laboratory quality assurance/quality control (QA/QC) has received much attention in recent years. The features of the SAS System that are useful for laboratory QA/QC are discussed in Chapters 4,6 and 8 of this book. Examples of control charting, calibration of lab instruments and repeatability/reproducibility of laboratory methods are presented.

Service industries have begun to define as a process the act of providing a service to the customer. Like manufacturing processes, these processes can be continually improved. Although these improvements may be more simplistic in nature, the need to use quantitative methods to explore the process is just as great. Through charts and graphs created by SAS software, these industries can find ways to increase customer satisfaction and worker productivity.

Although it is impossible to present examples from all types of industries, this book gives you some idea of the range of industries using quality improvement methods. Each type of industry presents varying challenges to PI teams who are using statistical software to display process data.

QUALITY IMPROVEMENT METHODS FOR ALL LEVELS OF EMPLOYEES

As well as spreading to many types of industries, the quality movement has spread to all levels of management within an organization. Statistical methods useful for quality improvement range from simple charts and graphs that are used by upper management to highly theoretical hypothesis-testing situations that are used by the company's statistical staff. This means that any employee within an organization could potentially be a user of quantitative methods for quality improvement.

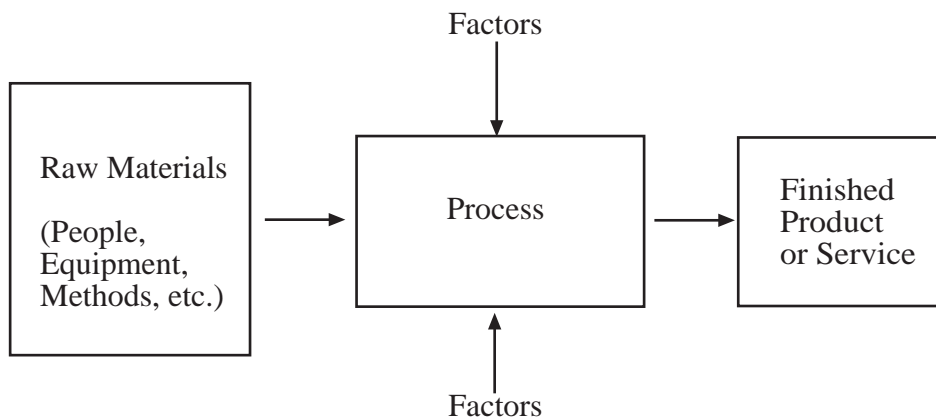
The successful use of statistical software to initiate and continue quality improvement depends on

- A having available flexible and easy-to-use statistical software
- B employing a staff that is dedicated to using data analysis methods to support quality initiatives
- C giving these employees reference materials on how to use the software to perform data analyses relating to quality improvement.

PROCESSES, VARIABILITY AND THE MEASUREMENT OF QUALITY

A *process* is defined as “the combination of people, materials, procedures, methods, policies and environment which when combined form a series of actions leading to a particular result.” Some examples of processes in industry are the production of building lumber by a sawmill, the robot welding of car frame components and the act of checking into a hotel. In all cases, raw materials (large trees, car frame components, guests) are input into the process and a finished product (lumber, car frame, registered guests) is produced. As the process converts the raw materials to the finished products, certain other factors (such as environment, operators, machine settings, etc.) influence the finished goods. The sequence is depicted in Figure 1.2.

Figure 1.2 *Diagram of a Process*



To evaluate the usefulness of the product to the consumer, a quality characteristic is measured on the finished product from a process. In the lumber manufacturing process described in the previous paragraph, the length and the width of the finished lumber are possible quality characteristics of interest. In the car frame process, the number of weld defects could be measured. In the hotel check-in example, the satisfaction/dissatisfaction of the customer with the length of the check-in procedure could be recorded. One or more quality characteristics may be of interest for any particular process or product.

Just as no two snowflakes are the same, no two products output from a process are identical. There is inherent variability in every process and, therefore, variability in the quality characteristics measured on the process. The statistical methods described in this book were developed as the result of a need to identify and quantify sources of variability in processes.

GOALS AND STRATEGIES OF THIS BOOK

As a member of a PI team, you may be responsible for summarizing and analyzing data collected on your process during the check phase of the PDCA cycle. SAS software provides many quantitative tools to accomplish this task. The focus of this book is to show you how the SAS System can be used to provide quantitative information that is essential to improving your process. The examples and code sequences contained in this book show you the framework for building a quality improvement system based on observed data from your process.

One of the goals of this book is to show the broad applicability of the SAS software as a tool for quality improvement. The book contains over 20 examples from many types of quality control applications. Although many examples focus on manufacturing situations, examples from service industries and medical/laboratory applications are also included. In each example, the four phases of the PDCA cycle are illustrated in the discussion, code sequences and outputs shown.

Another goal of this book is to show how the SAS System can be used by employees at all levels of management and statistical expertise to analyze and present data collected on a process. In order to accomplish this goal, the book is divided into two sections.

The first section of the book, consisting of Chapters 1, 2 and 3, requires only a basic knowledge of simple statistics used in the interpretation of charts and graphs. This section is intended for use by personnel at all levels of the organization. The remainder of this chapter gives an overview of the quantitative methods for process improvement discussed in the book and describes SAS software components available to perform these analyses. Chapter 2 gives a brief overview of data collection procedures and the use of the SAS DATA step for entering observed data from your process into the SAS System. Chapter 3 demonstrates how the SAS System can be used to construct charts and graphs to display information relating to quality improvement.

The second section, consisting of Chapters 4–8, details the statistical methods used for quality control. For those interested in the theoretical basis of the methods described, these chapters can be used in conjunction with statistics texts on quality control such as those by Montgomery (1991b), Duncan (1986), or Grant and Leavenworth (1988).

SAS software has developed into one of the most versatile packages on the market for quantitative display and analysis of data. The quality control (QC) component of the SAS system, SAS/QC, makes available to the user a large number of statistical methods for the identification of process variability. A summary of these methods is given in the next section.

USING THE SAS SYSTEM FOR STATISTICAL QUALITY CONTROL

Most of the statistical methods used in industry for analyzing process data fall into five main categories: graphical methods (requiring no statistical theory), control charts, capability analyses, design and analysis of experiments, and sampling plans. SAS/QC software, the QC component of the SAS System, contains many procedures and functions that directly implement most of these statistical methods. The remainder can be obtained through programming using the SAS DATA step, base SAS procedures, SAS/STAT procedures and SAS/GRAPH software. Table 1.1 summarizes the statistical methods described in this book and the procedures and components of the SAS System used to compute each of these methods. The location of these procedures in the SAS System is

shown in parentheses (B=base SAS, QC=SAS/QC, SG=SAS/GRAPH, SS=SAS/STAT, SE=SAS/ETS).

Table 1.1 SAS System Components Used for Quality Control

Statistical Methods	SAS System Procedures and Components Needed
graphical and tabulation methods	PROC PARETO (QC) PROC ISHIKAWA (QC) PROC GPLOT (SG) PROC TABULATE (B)
control charts and adaptive/automated process control	PROC SHEWHART (QC) PROC MACONTROL (QC) PROC CUSUM (QC) PROC FORECAST (SE) PROC ARIMA (SE) SQC Menu System (QC)
capability analysis	PROC CAPABILITY (QC) SQC Menu System (QC)
sampling plan design and evaluation	DATA step (B) SAS functions (B and QC)
variability reduction and process optimization using designed experiments (DOE)	PROC FACTEX (QC) PROC OPTEX (QC) PROC GLM (SS) PROC PLAN (B) PROC MIXED (B) PROC NESTED (B) PROC RSREG (SS) PROC NLP (SS) ADX Menu System (QC)
repeatability and reproducibility studies	GAGE application (QC) DATA step (B) PROC MIXED (SS)
statistical tolerancing	DATA step (B) SAS macros (B)
calibration	DATA step (B) PROC REG (SS)
reliability	PROC CAPABILITY ¹ (QC) PROC LIFEREG (SS)

In general, users of SAS software components shown in Table 1.1 fall into two groups: users who are trained programmers (either in SAS or another language) and users whose computer skills are limited. For the latter group, the use of SAS for statistical quality control is made easy by the SQC (Statistical Quality Control) and ADX

¹PROC CAPABILITY can be used to analyze noncensored reliability data. PROC RELIABILITY, due to be released in Release 6.11 of SAS, will have the ability to analyze both noncensored and censored reliability data.

(Automated Design of Experiments) Menu Systems. Those with programming experience may be more interested in examples involving SAS code.

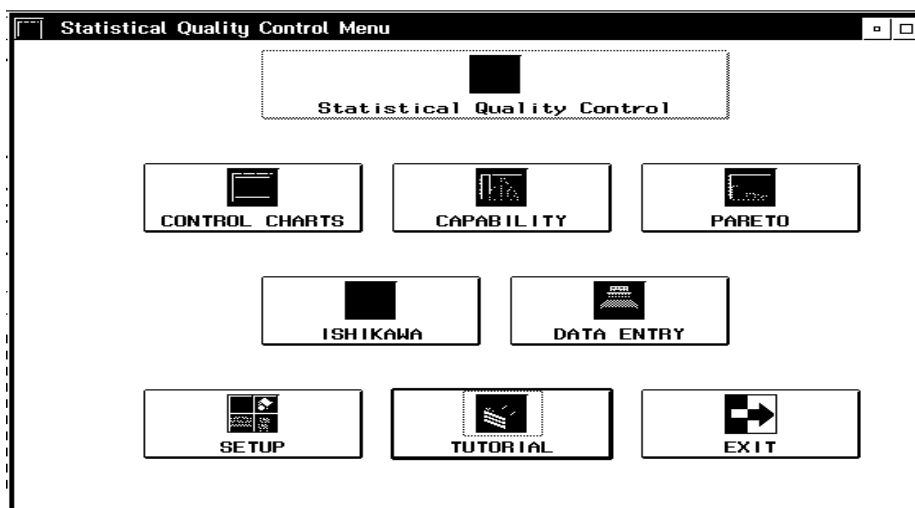
In order to satisfy both groups, the examples in this book are a mix of menu system and code usage. Many of the simpler examples can be performed using only the menu systems, although some of these have been done with code to save space in the book. The more complicated statistical tasks require code usage. An introduction to the menu systems is given in the next section. Some preliminary steps for the code examples are shown in the later sections.

USING THE SQC MENU SYSTEM

The SQC Menu System, available with SAS/QC, provides a set of point-and-click windows that give you the ability to summarize and analyze process data using a number of statistical methods. Many of the features available in SAS/QC procedures for control charts, process capability, Ishikawa and Pareto diagrams are available in the SQC Menu System through a set of menu selections. The SQC Menu System translates your menu selections into SAS code which is then submitted for execution. The submitted code produces SAS outputs that can be viewed or saved. In addition, you have the option to view the code used to generate these outputs and save them for later use. Although the SQC Menu System does not provide all the analyses available in SAS/QC software, you will find that the features of this system are sufficient to do many of the common QC analyses without a SAS programming background.

In order to initiate the SQC Menu System, type `sqc` on the command line in the Program Editor window of the SAS Display Manager System and press ENTER². The initial screen is shown in Figure 1.3. From this screen, you can enter data collected on the process or do a variety of analyses on the data, all of which are described in the following chapters of this book.

Figure 1.3 *Initial Window of SQC Menu System*



²If you use pull-down menus in the display manager, the SQC Menu System can be entered by choosing the GLOBALS option from the menu bar at the top of the screen and selecting ANALYZE→QUALITY IMPROVEMENT from the resulting pull-down menus.

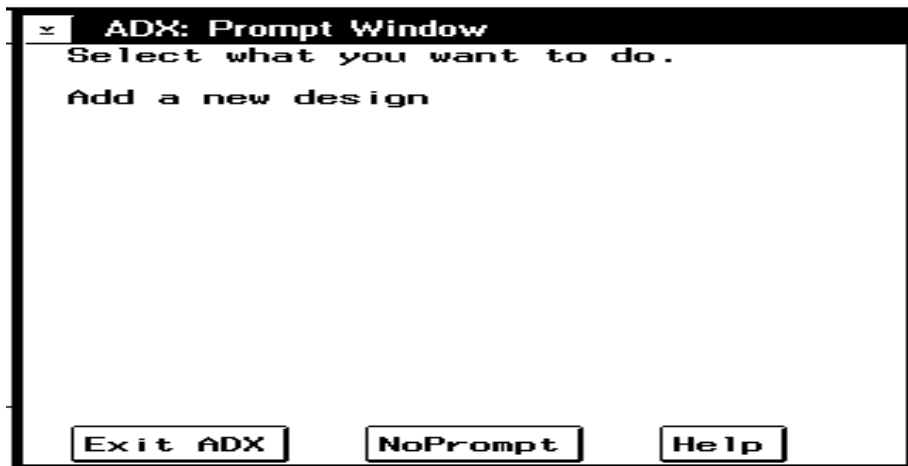
One of the only things that needs to be done prior to using the SQC Menu System is to set the graphics output device. All of the analyses that can be performed with the SQC Menu System produce graphics output. The first time you enter the menu system, you will want to set the graphics device you are using for hardcopy output. In order to do this, click on the SETUP icon, then choose the GRAPHICS radio button and select the GRAPHICS DEVICE icon on the SQC Setup Selections window. The graphics device for screen output will already be set to the proper screen device. To specify a hardcopy output device, choose the HARDCOPY DEVICE icon to trigger the Device Types window. From this window, select the device type and then page down the list of devices until you find the output device you wish to use. If you need help, a detailed explanation of how to use this menu system is given in *SAS/QC Software: SQC Menu System, Version 6, First Edition*.

USING THE ADX MENU SYSTEM

The ADX Menu System gives you the ability to design and analyze experiments conducted on your process. Several types of statistical designs are available through a series of easy to use menus. Many of the features available in the SAS/QC experimental-design procedures FACTEX and OPTEX are available in the ADX Menu System through a set of menu selections. The ADX Menu System translates your menu selections into SAS code which is then submitted for execution. The submitted code produces SAS output that can be viewed or saved. In addition, you have the option to view the code used to generate the output and to save the code for later use.

In order to initiate the ADX Menu System, type `adx` on the command line of the Program Editor window of the SAS Display Manager System and press ENTER. An initial screen is displayed which shows the designs that currently exist in your SAS user library and a list of possible actions (see Figure 1.4). In general, there are no options that need to be set before entering this menu system. If you need help, a detailed explanation of how to use this menu system is given in *SAS/QC Software: ADX Menu System for Design of Experiments*, Chapter 2.

Figure 1.4 *Initial Window of ADX Menu System*



THE INTERACTION OF SAS/QC AND SAS/GRAPH SOFTWARE

If you use SAS code to perform your analyses, the following paragraphs describe the initial steps to take in order to analyze your process data. These steps will allow you to present your data graphically in a more effective manner.

By nature, statistical methods for quality control are graphical. This allows the results of the analyses to be conveyed to a wide audience within an organization. Due to the relationship of graphics to quality-related statistical methods, much of the syntax used in the graphics procedures of SAS/GRAPH was carried over to SAS/QC. The following paragraphs explain these concepts.

All of the SAS/QC procedures that involve graphical displays (PARETO, SHEWHART, CAPABILITY, CUSUM, MACONTROL) can produce output in what is known as character or “line printer” mode. This type of output is known as low-resolution output. However, if your installation has SAS/GRAPH installed, high-resolution output can be obtained using a graphics output device. In order to obtain graphics output, you need to make two changes to your code. The first is to add the GRAPHICS option to the PROC statement of your QC procedure as in the following example:

```
proc shewhart graphics data=....;
```

The second is to add the GOPTIONS statement to your program ahead of your use of SAS/QC procedures. For most of the examples shown in this book, the GOPTIONS statement has been submitted just prior to invoking a SAS/QC procedure:

```
goptions hsize=6 vsize=4.5 border device=hplj300 ftext=swissx;
```

This statement indicates that the size of the graph to be drawn will be 6 inches wide and 4.5 inches high. The graph will have a border around it and will be output to an HP LaserJet Printer at 300 dots-per-inch resolution. All text on the graph will be drawn in SWISSX font. A few of the examples shown in the book require special settings of these graphics options. These are indicated with a GOPTIONS statement shown with the specific example.

In addition, many graphics options available in SAS/GRAPH procedures are also available in SAS/QC. Two of the most useful are the ability to save graphics output using the GOUT= option in the PROC statement and the ability to annotate graphics output using the ANNOTATE= option. For a complete description of all graphics statements and options, see *SAS/GRAPH Software: Reference, Version 6, First Edition, Volumes 1 and 2*.

THE USE OF MACRO LANGUAGE IN THIS BOOK

Some of the statistical analyses in this book are written in SAS macro language. This language is provided with base SAS software and gives you the ability to write programs that generate SAS code sequences. The result is that SAS code sequences used for statistical calculations can be written such that the parameters of the analysis correspond to macro variables. This allows the code sequence for the analysis to be used repeatedly with different initial values for the parameters.

It is not a prerequisite that you know or understand macro language to use this book. You need only know how to call the macros provided in certain sections. As an example, consider these statements:

```
%macro red(dsn=,var=);  
  proc print data=&dsn;  
    var &var;  
  run;  
%mend red;  
  
%red(dsn=one,var=x y z)
```

All macro language statements begin with the % sign. The %MACRO statement is the first statement in the macro and %MEND is the last. This macro, named RED, generates code to print a SAS data set using PROC PRINT and has two arguments, DSN= and VAR=. These arguments contain information the macro uses to generate the appropriate SAS statements.

The final statement shown is known as the macro call statement. This statement causes the macro to begin execution and initializes the macro variables, DSN and VAR. In order to use prewritten macros, an understanding of the macro call statement is all that is needed. For the macro RED, the name of the data set to be printed is given in the DSN= argument and the variables on the data set to be printed are given in the VAR= argument.

When the %RED statement executes, the values one and x y z are assigned to their respective macro variables (DSN and VAR). The macro processor then executes the macro instructions and produces a set of SAS programming statements. For this simple macro, the only macro processing operation is the substitution of values of the macro variables in the three SAS statements shown. These substitutions occur where a macro variable name is preceded by the & sign. The macro generates the following code:

```
proc print data=one;  
  var x y z;  
run;
```

This code is then executed by SAS to print the specified variables in the specified data set.

Many of the examples of macro language used in this book involve the passing of arguments to a macro as illustrated in the simple example above. For help with the more complicated macro applications discussed in this book, see *SAS Language: Reference, Version 6, First Edition*, Chapter 20 and the *SAS Guide to Macro Processing, Version 6, First Edition*.

EXAMPLE DATA SETS

There are many example data sets used in this book. As each example is introduced, at least a portion of the SAS data set containing the example data is displayed through the use of PROC PRINT. In order to make the chapters less cluttered, Appendix A contains the SAS DATA step code that creates the SAS data set for each example. Refer to this appendix to examine the source data.

