Paper 173-2008

Creating Complex Reports

Cynthia L. Zender, SAS Institute, Inc., Cary, NC

ABSTRACT

Are you confused about whether you need a detail report or a summary report? Do you wonder whether you are using the right reporting procedure for your report? Have you ever spent a lot of time going down the road with one procedure, only to discover that you need to switch to a different procedure to get what you want or closer to what you want?

This paper is for intermediate SAS programmers who want to see a gallery of complex reports and review the code that produced them. It covers the pros and cons of PROC REPORT, PROC TABULATE, and DATA _NULL_ for each report. All report output is produced using ODS (rather than LISTING) output. Get out your wrenches and tool belts and fasten your seat belt for the ride, this is going to be a real nuts and bolts paper presentation.

INTRODUCTION

SAS users have a variety of reporting procedures available to them. PROC PRINT produces detail reports or listings, where one report row represents one observation in the data set or subset being processed. PROC REPORT produces both detail reports and summary reports. Like PROC PRINT, a detail report row represents one observation. However, in a summary report, each report row represents the summary of a group of observations. PROC REPORT offers more control over subtotals and grand totals than PROC PRINT. In addition, PROC REPORT allows the use of selected DATA step language statements to calculate new report items or change existing report items (such as the custom text on the summary line). PROC TABULATE, on the other hand, produces only summary reports, based on the class variables and the analysis variables used to calculate the statistics for the report cells. Although many users consider PROC SQL to be a query and reporting language, a discussion of using PROC SQL for reporting falls outside the scope of this paper.

BASIC REPORT TYPES

Reports serve many purposes. There is the “just get me a number” report, the “I’m checking the data” report, the “who/what/where/when” report, the production report, the “get the information for another product” report, the “send this to the VP’s BlackBerry” report, the “put this information on a Web site” report, and so on. What all reports have in common is that there is some kind of input (usually the data), some kind of process, selection criteria to limit the data, statistics to be produced, groups to be set, or recoding to be done. And, there is some kind of output (the report). The report has to go someplace. It has to be used, distributed, or routed to a printer, a Web server, an e-mail box, or a snail mail box. Frequently, what happens to a report after it is produced controls how the report is written or what the report looks like. If you are generating the “just get me a number” report, then the cosmetics of the report do not need much consideration. But, if you are generating a production report, it is possible that the report will be e-mailed to company directors, and cosmetics need more consideration. It is possible that the report will be submitted to a governing body where cosmetics are controlled by the governing body.

Data is not the only input for the report process. All of the considerations mentioned above are input. These considerations are related to the data, and are questions you should answer before you start coding.

- Do you need a detail report or a summary report?
- Is the data ready to use or are there transformations to be performed?
- How much weight should be given to cosmetic issues? What are the cosmetic issues or requirements?
- What is the ultimate distribution of the report?

The answers to all of these questions control the ODS destination that you use and the procedure and syntax you use to create and enhance the program output. The answers will help you make other decisions before you start coding. For example, your boss wants some summary numbers, but wants them in Microsoft Excel to use them as input to a budget process driven by Microsoft Excel macros. The data might need to be lightly summarized, perhaps with PROC MEANS or PROC TABULATE. To get the output to Microsoft Excel in the quickest way, you might use ODS CSV or PROC EXPORT. There are no cosmetic issues if all your boss wants are the data points.

But, what if your boss wants a report for distribution? What if the report needs to be composed of two parts—a summary report and a detailed list of all data rows. The report is going to be delivered via the Web, but cannot be in HTML format. It has to be in a format that is not easily edited. These requirements point to a PDF file. In one ODS "sandwich," you can create a single PDF file that holds the summary report and the detailed list of all data rows (the detail report). When creating a PDF with ODS, you can automatically create PDF bookmarks and a Table of Contents for the two different parts of the report. The first, most important question to answer is whether your report needs to be a detail report or a summary report.

DETAIL OR LISTING REPORTS

Consider the data set SASHELP.CLASS. It has 19 observations, 9 female students, and 10 male students. The following reports are detail reports, produced with PROC PRINT and PROC REPORT for the subset of female students.

PROC PRINT Detail Report

Sex	Name	Age	Height
F	Alice	13	56.5
	Barbara	13	65.3
	Carol	14	62.8
	Jane	12	59.8
	Janet	15	62.5
	Joyce	11	51.3
	Judy	14	64.3
	Louise	12	56.3
	Mary	15	66.5
N = 9			

PROC REPORT Detail Report

Sex	Name	Age	Height
F	Alice	13	56.5
	Barbara	13	65.3
	Carol	14	62.8
	Jane	12	59.8
	Janet	15	62.5
	Joyce	11	51.3
	Judy	14	64.3
	Louise	12	56.3
	Mary	15	66.5
Female Students N = 9			

Figure 1: PROC PRINT Detail Report Compared to PROC REPORT Detail Report

Many users would call this a listing report because it lists a subset of observations, and each observation has its own row. I prefer not to mix two different meanings of the term "listing," one being a listing report and the other being the ODS LISTING destination. My preference is to describe the above reports as detail reports. Except for minor cosmetic differences, you get the same kind of detail report from PROC PRINT and PROC REPORT. In fact, you could generate the report with a DATA _NULL_ program as well, but that seems to me like using a sledgehammer to pound in a thumbtack. DATA _NULL_ reporting capabilities have changed significantly since the days of FILE PRINT and PUT statements sending monospace text to the LISTING window. DATA _NULL_ can produce detail or summary reports. When used with ODS, these reports can be only tabular in structure (in SAS 9.1.3). Future developments for DATA _NULL_ reporting capabilities include a new object dot syntax for report creation, as described in the SUGI paper "Next Generation Data _NULL_ Report Writing Using ODS OO Features." This new object dot syntax allows free-format reports to be created with DATA _NULL_ and ODS.