

## Guidelines for SAS Program Development to Facilitate Migration of Standalone Code to SAS/PH-Clinical Templates

Steve Light & Paul Gilbert  
DataCeutics, Inc.

Many companies are currently developing SAS reporting programs that will eventually be migrated to SAS/PH-Clinical Templates. These guidelines for the development of SAS programs facilitate the migration of stand-alone SAS programs to SAS/PH-Clinical templates.

1. **Follow a standard program organization.** The process of converting programs to Templates is simplified when important program components can always be found in the same place. The following are ideas for standard program organization:

- Fully document all programs. At the minimum, the program should contain a documentation block to describe the program, input data and output files.
- Locate all libname and SAS System options statements at the start of the program.
- Locate all data steps or proc sorts which bring in permanent data at the start of the program. These must be modified during template conversion.
- Pair data manipulation and procedures which display results, rather than doing all data manipulation for several reports or tables and then running all output procedures. In other words, write

<pre>data a; set x;   &lt;your code here&gt;; run;</pre>	instead of	<pre>data a; set x;   &lt;your code here&gt;; run;</pre>
<pre>proc print data=a; run;</pre>		<pre>data b; set y;   &lt;your code here&gt;; run;</pre>
<pre>data b; set y;   &lt;your code here&gt;; run;</pre>		<pre>proc print data=a; run;</pre>
<pre>proc report data=b; data=b; run;</pre>		<pre>proc report run;</pre>

- Develop and follow a convention for title and footnote numbering. SAS/PH-Clinical allows use of global titles and footnotes which are automatically applied to all templates. These globals can be overwritten by statements in programs if the numbering is not worked out.
- Comments should only be written using the \* ... ; syntax. This will allow blocks of code to be commented out using the /\* \*/ comment syntax during template conversion.
- Use keep= statements in initial data steps or sorts of permanent data. During template conversion all the variables used in a program must be mapped to the PH-Clinical dictionary. The keep statement will make it easy to identify what variables are required.

2. **Use only standard data and tools.**

- The data library and format library used in standalone programming and in PH-Clinical should be the same.
- Do not use 'personal' format or macro libraries.
- Do not use %include files. Standardized %include files should be converted to SAS macros and placed in the autocall library.
- Maximize use of standard utility and report macros to minimize the quantity of code requiring modification when migrating to PH-Clinical templates.

3. **Remember that PH-Clinical is an interactive SAS environment:** Some programming statements which are convenient in a non-interactive environment can be disruptive in PH-Clinical. Some things to keep in mind include:

- Do not use Proc Datasets to delete the work library at the end of a program, or to save only selected datasets. This can delete work datasets created by PH-Clinical.
- Do not use ENDSAS statements. This will terminate an interactive session.
- Keep in mind that once global macro variables, temporary formats and system options are defined they will remain in effect for the current PH-Clinical session.



DataCeutics, Inc. is a consulting group specializing in software solutions, system integration, programming and support in the areas of clinical data management and statistical reporting. For information on SAS/PH-Clinical consulting or the Clinical Reporting Toolkit (CR Toolkit<sup>®</sup>) for SAS/PH-Clinical, contact Paul Gilbert at 610-970-2333 or gilbertp@dataceutics.com.