VIP Platform – Userguide, conventions - DRAFT

[1. Folder Structure Summary (technical – Reporting Effort level only) 3](#_Toc104280144)

[2. Standard configuration / Setup macros. 6](#_Toc104280145)

[2.1. %INIT 6](#_Toc104280146)

[2.2. VERASETUP 6](#_Toc104280147)

[2.3. AUTOEXEC.SAS 7](#_Toc104280148)

[3. Naming conventions 8](#_Toc104280149)

[3.1. SAS Programs / datasets / output 8](#_Toc104280150)

[3.2. TFL programs / datasets / output 8](#_Toc104280151)

[3.3. Macros 8](#_Toc104280152)

[4. FORMAT.SAS 9](#_Toc104280153)

[5. Dataset Specifications. 9](#_Toc104280154)

[6. Titles / Footnotes. 9](#_Toc104280155)

[7. ODS Template 10](#_Toc104280156)

[8. General program Style / Layout. 11](#_Toc104280157)

[8.1. Program Templates 11](#_Toc104280158)

[8.1. PROD programs 11](#_Toc104280159)

[8.2. QC Programs 12](#_Toc104280160)

[9. FINAL programs 13](#_Toc104280161)

[9.1. PROD program FINAL.SAS 13](#_Toc104280162)

[9.2. QC PROD program Q\_FINAL.SAS 13](#_Toc104280163)

[10. Batch Execution 14](#_Toc104280164)

[11. Macros 15](#_Toc104280165)

[11.1. %S\_SCANLOG 15](#_Toc104280166)

[11.1.1. Interactive SAS 15](#_Toc104280167)

[11.1.1.1. Pre-requisite 15](#_Toc104280168)

[11.1.1.2. Example call 15](#_Toc104280169)

[11.1.1.3. Result 15](#_Toc104280170)

[11.1.2. Library scanning mode. 16](#_Toc104280171)

[11.1.2.1. Pre-requisite 16](#_Toc104280172)

[11.1.2.2. Example call 16](#_Toc104280173)

[11.1.2.3. Result 16](#_Toc104280174)

[11.1.3. Text in log(s) being scanned 16](#_Toc104280175)

[11.2. %S\_COMPARE 17](#_Toc104280176)

[11.2.1. Single datasets compare 17](#_Toc104280177)

[11.2.1.1. Prerequisite. 17](#_Toc104280178)

[11.2.1.2. Example calls 17](#_Toc104280179)

[11.2.1.1. Result 17](#_Toc104280180)

[11.2.2. Library compares 17](#_Toc104280181)

[11.2.2.1. Prerequisite. 17](#_Toc104280182)

[11.2.2.1. Example calls. 18](#_Toc104280183)

[11.2.2.1. Result 18](#_Toc104280184)

# Folder Structure Summary (technical – Reporting Effort level only)

| **Reporting Effort** | **Category** | **Type** |  | **Description / detail** |
| --- | --- | --- | --- | --- |
| *&\_\_RE* | *&\_\_PROGCAT* | *&\_\_PROGTYPE* |  | Global Macro variable name created by %INIT holding the folder name based on the current program path |
| RE\_xxx |  |  |  | Folder containing everything needed for any single reporting effort |
|  | Config |  |  | Contains VERASETUP  This is the study level configuration containing the logic to assign LIBNAMES / SASAUTOS / FMTSEARCH and other standard configuration items for the study. Ideally this macro will need no change from the default. If update is needed – handle with care, and seek advice. |
|  | Data |  |  | Folder containing all data used through the study reporting |
|  |  | adam |  | ADaM datasets (production)   * If delivered by third party vendor – these will be the Veramed source datasets for TFL programming * If Veramed creating ADaM, this is where ADaM datasets will be created |
|  |  |  | QC | QC ADaM dataset for validation |
|  |  |  | xpt | Veramed created XPT version of production datasets |
|  |  | meta |  | Metadata from specifications / titles etc. Full details pending |
|  |  | Raw |  | Raw data (if not CDISC, or Veramed are creating SDTM as part of the process)   * Source / Raw data directly from data management (no pre-processing to have taken place. Data exactly as received from vendor) * Will contain **all** source data (EDC, third party lab data, protocol deviation data, rand lists (dummy then live). * Source data must only reside in this folder (not in subfolders) * Must only contain current data for the current deliverable. |
|  |  | Refdata |  | * Reference data. E.g. – formats catalogs/dataset for each program type (do not store subject data in here) |
|  |  | sdtm |  | SDTM datasets (production)   * If delivered by third party vendor – these will be the Veramed source datasets for ADaM * If Veramed creating SDTM, this is where SDTM datasets will be created |
|  |  |  | QC | QC SDTM datasets for validation |
|  |  |  | xpt | Veramed created XPT version of production datasets |
|  |  | tfl |  | Datasets saved prior to PROC report, used for TFL validation process |
|  |  |  | qc | QC TFL dataset for validation |
|  |  | util |  | Any data required for a utility program. Note: This must not contain any data or information used directly in the reporting of the study |
|  |  |  | qc | QC util datasets for validation |
|  | Documents |  |  | And many subfolders. Detail pending |
|  |  | Templates |  | And subfolder per program type. Contains example program template to drive adherence to standards. These can be adapted per study as necessary (especially headers) |
|  | Prod |  |  | Product programs, logs and output |
|  |  | Adam |  |  |
|  |  |  | Code | ADaM programs |
|  |  |  | Macro | <Optional>  Whilst it is advised the PROD / QC macros are retained in their respective PROD\share\macro folder to keep central, if there is a need to further delineate (e.g. – due to macro volume – especially with TFL macros) this optional folder can be created for storing macros. SASAUTOS will pick these up automatically (via verasetup) but team must ensure content is managed appropriately |
|  |  |  | Output | Any output from ADaM programs as necessary |
|  |  |  | Saslogs | SAS log file from batch ADaM execution |
|  |  | SDTM | … | Same convention and subfolders as ADaM |
|  |  | Share |  |  |
|  |  |  | Macro | Macros – only to be used by Production SDTM / ADaM / TFL. Single location for all macros |
|  |  | Tfl | … | Same convention and subfolders as ADaM |
|  |  | Util | … | Same convention and subfolders as ADaM. Utility is for storing programs not directly used in the delivery of results |
|  |  | <new> |  | If a new program “type” is required, project team can build following the same logic as above (in both PROC and QC). Consideration must have be made for LIBNAME management which is advised to be dealt with in VERASETUP to ensure centralization |
|  | Qc | … | … | Suite of folders for all QC activities. Sub-structure matches that for PROD.  Ensures all files in QC folders adhere to the Q\_ naming convention to ensure no cross over of code between PROD and QC. |
|  | Share | Macros |  | Shared macros that are “approved” to be used across both the PROD and QC program suites.  If any macros are directly changing any data, there must be a full suite of validation material available to support this. |
|  |  | Tests |  | Suites of test programs for any share macro updates |
|  |  |  | Code |  |
|  |  |  | Output |  |
|  |  |  | saslogs |  |

# Standard configuration / Setup macros.

## %INIT

Global SASAUTOS now includes folder “z:\project\_tools\macros”. Contained within this folder is macro %INIT which is now immediately available in all programs executed in Interactive SAS / EG or batch SAS. This macro is key to the new folder structure and MUST be called in every dataset / TFL program in the new folder structure (see program templates for placement)

In summary this macro

* Automatically Identifies the current SAS execution method
* Automatically Identifies the current program name and location
* Builds multiple global macro parameters / variables for use downstream. E.g.
  + &\_\_PROGCAT (QC or PROD)
  + &\_\_PROGTYPE (SDTM or ADAM or TFL)
  + &\_\_RUNMODE (INTERACTIVE / DMS / EG)
  + &\_\_FULL\_PATH (Full path of current program.

To view full list, execute macro in a saved program and check parameters via %put \_user\_;

* Runs study specific macro %VERASETUP

## VERASETUP

This study level program controls a number of standard configurations at the study level, negating the need to have any folder paths / libnames declared in any programs.  
This program is automatically called by the %INIT macro, and therefore is never directly referenced in any programs.  
Utilising the information passed in via %INIT, the macro will setup libnames, additional SASAUTOS, FMTSEARCH and other useful items that are specific directly to the program you are running, in the study you are running it.

E.g. Libname Matrix.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Program Type (PROD) | | |
| Libname | *SDTM* | *ADaM* | *TFL* |
| *RAW* | Read Only | Read Only | Read Only |
| *SDTM* | Read Only | Read Only | Read Only |
| *SDTMW* | Read/Write | n/a | n/a |
| *ADaM* | n/a | Read Only | Read Only |
| *ADaMW* | n/a | Read/Write | n/a |
| *TFL* | n/a | n/a | Read Only |
| *TFLW* | n/a | n/a | Read/Write |

|  |  |  |  |
| --- | --- | --- | --- |
|  | Program Type (QC) | | |
| Libname | *SDTM* | *ADaM* | *TFL* |
| *RAW* | Read Only | Read Only | Read Only |
| *SDTM* | Read Only | Read Only | Read Only |
| *SDTMQCW* | Read/Write | n/a | n/a |
| *ADaM* | n/a | Read Only | Read Only |
| *ADaMQCW* | n/a | Read/Write | n/a |
| *TFL* | n/a | n/a | Read Only |
| *TFLQCW* | n/a | n/a | Read/Write |

Selected additional global variables are available, directly driven by the location of the current program, to be used in place of any fixed pathname

* &\_\_LOGPATH – Location of log folder for current program
* &\_\_OUTPATH – Location of output folder for current program (e.g. for ODS file statements)
* &\_\_DATAPATH – Location of source data for current program (e.g. for proc import XLS or CSV)
* &\_\_METAPATH – Location of metadata that is created by Dataset Spec and titles / footnotes process for use by multiple macros

The final execution step for VERASETUP is to run the study level AUTOEXEC.SAS program

## AUTOEXEC.SAS

Final piece of the study configuration to control programming environment. This contains the simpler study specific configuration, where there is an expectation of potential change for a studies configuration, whilst adhering to standards.  
This program is automatically called by the %INIT macro, and therefore is never directly referenced in any programs.  
Items within the AUTOEXEC.SAS may include (but are not limited to)

* SAS system options
* Global variables for
  + Study ID (for use in outputs)
  + Status flag (for outputs) if data running on live or dummy rand list
  + Data cut off label (for outputs)
  + Split character (for outputs, and fed into compare)
  + Label for use in SCANLOG and COMPARE doutput docs
* More….

All of the above default configuration will be available when a study is first setup.

Once setup, Programming can commence.

# Naming conventions

Following are the naming conventions for all Programs / Macros / Data / Outputs etc.

## SAS Programs / datasets / output

|  |  |  |  |
| --- | --- | --- | --- |
|  | Program Name | Output Dataset Name | Output name |
| Production | <datasetname>.sas | <datasetname>.sas7bdat | n/a |
| QC | q\_<datasetname>.sas | <q\_datasetname>.sas7bdat | n/a |

**Summary**: All dataset programs to be named per the dataset being created. All QC programs to match, with addition of q\_ prefix only.

## TFL programs / datasets / output

|  |  |  |  |
| --- | --- | --- | --- |
|  | Program Name | Output Dataset Name | Output name |
| Production | T<progid>.sas | T<outputid> | T<outputid>.rtf |
| Production | F<progid>.sas | F<outputid> | F<outputid>.rtf |
| Production | L<prodid>.sas | L<outputid> | L<outputid>.rtf |
| QC | Q\_T<progid>.sas | Q\_T<outputid> | n/a |
| QC | Q\_F<progid>.sas | Q\_F<outputid> | n/a |
| QC | Q\_L<prodid>.sas | Q\_L<outputid> | n/a |

**Summary:**

* TFL programs to be named per the defined *progid*. Whether one program per output or one program per multiple outputs. This is a decision point to be made by the study team as best determined by the similarity of outputs. T F or L prefix to be included
* TFL output dataset to be named with the output number being generated to ensure clear identification. One dataset per output must be adhered too to ensure batch validation can take place. Name zero filled to ensure correct ordering of outputs in folder system
* TFL output to be named with at least the output number being generated to ensure clear identification. One output file per distinctly numbered output. Name zero filled to ensure correct ordering of outputs in folder system.

## Macros

* Macro name is always to be a logical, concise and distinct name, ensuring a clear understanding of the macro action.
* All stored macros to be simple and focus on one task.
* Macros should avoid dependencies to other macros where possible.
* Stored macros must be one macro per macro file for simple identification.
* Macro file name must match macro name to ensure correct SASAUTO access
* Conventions below ensure simple identification of location of macros. Also ensures macros are not crossed over between PROD and QC locations

|  |  |  |  |
| --- | --- | --- | --- |
|  | Macro name  (filename=<macroname>.sas) | Any global macro variables / parameters generated by macro for use elsewhere | Storage location |
| Global | <name> | \_\_<paramname> | Z:\project\_tools\macro |
| Shared | s\_<name> | s\_<paramname> | ..\re\_\*\share\macro |
| Production | p\_<name> | p\_<paramname> | ..\re\_\*\prod\share\macro |
| QC | q\_<name> | q\_<paramname> | ..\re\_\*\qc\share\macro |

The default is to create study specific macros in the prod\share\macro or qc\share\macro folder. This allows clear visibility of study level macros across both PROD and QC folders.

If the is a need to store macros the lowest possible level (e.g. prod\tfl\macro if there are a significant number of TFL specific macros and it is more efficient to separate) such folders can be created on demand. The SASAUTO setting will automatically be updated to point to these folders first

# FORMAT.SAS

SAS formats are defined by program type (SDTM / ADaM or TFL) for use in either PROD or QC. format.sas (or q\_format.sas) programs are to exist in each code folder and are to hold formats to be used across multiple programs (within program type).

If a format is only to be used in a single program, it can be defined as a work format directly within your data/tfl program

Specification formats (CDISC controlled terminologies) coming from dataset specifications may also exist. These can be added via specific macro calls within the respective formats.sas (or q\_formats.sas) program to interact with the specification metadata. These will be saved to the respective catalog.

The FMTSEARCH path is controlled in VERASETUP to ensure there is no PROGTYPE or PROD v QC crossover. WORK is searched first, then your respective specific catalog (only)

# Dataset Specifications.

These are to be completed following the standard Veramed Template and shall be stored here.

Every time the specification is saved, the metadata is automatically exported and immediately available for use by macros to allow the metadata to be used directly in programs.

MORE TO ADD..

# Titles / Footnotes.

All titles and footnotes source files are to be stored in folder “re\_XXXX\documents\SAP and Shells”. The data files generated automatically when saving the source files (that are accessed directly by SAS macros) are to be stored in folder re\_XXXX\data\meta

Within all TFL programs, Macro XXXXX it utilised to “pull” the titles and footnotes from the metadata and apply to the output in question.

# ODS Template

The ODS template to be used will be delivered as a standard template file. By default the standard Veramed template should be used, but if study specific requirements drive a change, please update accordingly and adjust the name of the ODS style.  
The ODS template will be stored as macro in folder \SHARE\MACRO and compiled by the VERASETUP macro.

# General program Style / Layout.

## Program Templates

Program shells can be created directly by the VIPER process. However, to further aid the create of specific programs for any given task, some default template are available in the following location.

…\re\_XXXX\documents\templates

It is the responsibility of the project lead to determine is these need adapting for and study specific process.  
It is advised that team member create new programs by copying the respective template into the require CODE subfolder

## PROD programs

All Dataset and TFL programs must contain at least the following, which will be available via XXXXX templates (or Viper build scripts)

PROD Programs

DM commands to clear interactive LOG and OUTPUT windows for each run. Ignored when running in EG and batch

dm 'out;clear;';

dm 'log;clear;';

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\

\* O

\* /

\* O---O \_ \_ \_ \_ \_ \_ \_|

\* \ \/(/\_| (\_|| | |(/\_(\_|

\* O

\* \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\* Sponsor : <SPONSOR>

\* Study : <STUDY>

\* Program : XXXX.SAS

\* Purpose :

\* \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\* DESCRIPTION

\*

\* Input files: None

Standard Veramed header. Must be kept up to date

\*

\* Output files: None

\*

\* Macros: None

\*

\* Assumptions:

\*

\* \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\* PROGRAM HISTORY

\* <DATE> | <NAME> | Original version

\\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

Initialisation macro setting all required configuration based on current program location

\*\*\*\*\*\*\*\*\*;

%***init***;

\*\*\*\*\*\*\*\*\*;

\*\*\*\* USER CODE FOR ALL DATA PROCESSING \*\*;

\*\*\*\* END OF USER DEFINED CODE \*\*;

\*\*\*\*\*\*\*\*;

Standard macro giving you instant feedback on SAS log content. Works in both SAS DMS and EG

%***s\_scanlog***;

\*\*\*\*\*\*\*\*;

## QC Programs

In addition to items above

dm 'out;clear;';

dm 'log;clear;';

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\

\* O

\* /

\* O---O \_ \_ \_ \_ \_ \_ \_|

\* \ \/(/\_| (\_|| | |(/\_(\_|

\* O

\* \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\* Sponsor : <SPONSOR>

\* Study : <STUDY>

\* Program : XXXX.SAS

\* Purpose :

\* \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\* DESCRIPTION

\*

\* Input files: None

\*

\* Output files: None

\*

\* Macros: None

\*

\* Assumptions:

\*

\* \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\* PROGRAM HISTORY

\* <DATE> | <NAME> | Original version

\\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

\*\*\*\*\*\*\*\*\*;

%***init***;

\*\*\*\*\*\*\*\*\*;

\*\*\*\* USER CODE FOR ALL DATA PROCESSING \*\*;

Standard compare macro giving immediate results from dataset compare (for either Dataset or TFL programming). Note – ALL datasets must be saved to z drive. Never compare work datasets.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

%***s\_compare***(base=<LIB>.<DATASET>,comp=<LIB>QC.Q\_<DATASET>);

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*;

\*\*\*\* END OF USER DEFINED CODE \*\*;

\*\*\*\*\*\*\*\*;

%***s\_scanlog***;

\*\*\*\*\*\*\*\*;

# FINAL programs

Every program folder will have its own FINAL.SAS (for Q\_FINAL.SAS) program, being the last executed program in any batch run (controlled by VIPER – see below). This is an important part of the quality process, to allow quick and easy access to key information. These programs will create PDF summary reports that form part of the quality package / TMF for all deliveries.

Depending on whether the program category is PROD or QC – the following calls will be mandated

## PROD program FINAL.SAS

In addition to ALL standard program template code, the following will be present

\*\* Full folder level log scan and report \*\*;

%***s\_scanlog***(logfldr=%str(&\_\_logpath),debug=n,

logrpt =%str(&\_\_logpath\%str(&\_studyid)\_11.03**.02**\_%sysfunc(date(),yymmddn.)\_  
&\_relabel.-&\_\_PROGTYPE.-SASLog-Summary.pdf));

This %S\_SCANLOG call scans all the SAS logs stored in the current programming folder. This uses the exact same logic as when running in interactive programs. With this execution a PDF file is generated (named per SMF/TMF requirements) in the SASLOG folder, detailing the summary status of all SAS logs

## QC PROD program Q\_FINAL.SAS

\*\* Full library level compare and report \*\*;

%***s\_compare***(base=<LIBRARY>.\_ALL\_,

comp=<LIBRARYQCW>W.\_ALL\_,

comprpt =%str(&\_\_outpath\%str(&\_studyid)\_11.03**.02**\_%sysfunc(date(),yymmddn.)  
 \_&\_relabel.-&\_\_PROGTYPE.-Compare-Summary.pdf));

The %S\_COMPARE call runs a library level PROC COMPARE against all stored datasets. This uses the exact same logic as when running in interactive programs. With this execution a PDF file is generated (named per SMF/TMF requirements) in the OUTPUT folder, detailing the summary status of all compares.

\*\* Full folder level log scan and report \*\*;

%***s\_scanlog***(logfldr=%str(&\_\_logpath),

logrpt =%str(&\_\_logpath\%str(&\_studyid)\_11.03**.02**\_%sysfunc(date(),yymmddn.)  
\_&\_relabel.-&\_\_PROGTYPE.-QC-SASLog-Summary.pdf));

as above for details

# Batch Execution

Standard batch execution of all programs is taken care of by the VIPER tool.

This allows user to control order or execution, saving of SAS log files for QC. This MUST be used for all execution contributing to a client deliverable. Additional details to follow

# Macros

## %S\_SCANLOG

Shared macro, supported for use on both PROD and QC side of programming as program does not change / impact any of the clinical data directly.

Scans either interactive SAS log or library of stored \*.log files for selected messages and returns results to user.

## Interactive SAS

## Pre-requisite

The SAS program is saved to the network in the correct location.

Interactive SAS log only contains SAS log for current program. This should be handled via dm “log clear”; as first code is all SAS programs.

## Example call

%**s\_scanlog;**

## Result

The following is sent directly back to the same interactive SAS log that has just been scanned.

Issues to report:

NOTE:

NOTE:

NOTE: \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

NOTE: \*\* SUMMARY OF LOG EXECUTION \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

NOTE: \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

NOTE: \*\*

ERROR: \*\* 3 x ERROR

WARNING:\*\* 1 x WARNING

NOTE: \*\* 3 x uninitialized

NOTE: \*\* 2 x USER or MACRO DEFINED Debug messages for further assessment

NOTE: \*\*

NOTE: \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

NOTE: \*\* END \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

NOTE: \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

NOTE:

NOTE:

Clear Log:

NOTE:

NOTE:

NOTE: \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

NOTE: \*\* SUMMARY OF LOG EXECUTION \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

NOTE: \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

NOTE: \*\*

NOTE: \*\* NO MESSAGES OF IMMEDIATE CONCERN

NOTE: \*\* ENSURE LOG IS STILL THOROUGHLY REVIEWED

NOTE: \*\*

NOTE: \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

NOTE: \*\* END \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

NOTE: \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

NOTE:

NOTE:

**Note**: When SAS programs are executed in batch mode, the %S\_SCANLOG calls are automatically disabled to prevent double counting.

## Library scanning mode.

## Pre-requisite

SAS batch execution has taken place and all SAS log files are correctly stored on network

## Example call

%***s\_scanlog***(logfldr=%str(&\_\_logpath),debug=n,

logrpt =%str(&\_\_logpath\%str(&\_studyid)\_11.03**.02**\_%sysfunc(date(),yymmddn.)\_  
&\_relabel.-&\_\_PROGTYPE.-SASLog-Summary.pdf));

## Result

PDF report of the result of the scan of all log files to be stored in the same location as the scanned logs.

Graphical user interface, text, application, email

Description automatically generated

## Text in log(s) being scanned

* ERROR
* WARNING
* UNINITIALIZED
* MERGE STATEMENT HAS MORE THAN
* VALUES HAVE BEEN CONVERTED
* NOTE: MISSING
* NOTE: INVALID ARGUMENT
* W.D FORMAT WAS TOO SMALL
* HAS 0 OBSERVATIONS
* VARIABLES NOT IN
* VARIABLES HAVE CONFLICTING
* UNEQUAL
* DIVISION BY ZERO DETECTED
* OPERATIONS COULD NOT BE PERFORMED
* DUPLICATE KEY VALUES WERE DELETED
* OUTSIDE THE AXIS RANGE
* HAS BEEN TRUNCATED
* WAS NOT FOUND OR COULD NOT BE LOADED
* DEBUG: [ or DEBUG:[

## %S\_COMPARE

Shared macro, although expectation is this will only be called on the QC side of programming.  
Compares either a single selected SAS dataset, or two libraries of data. Macro may also be called in a utility program if needing to compare datasets from different deliverables, or checking data received from vendors at different times (utilise parameter “IgnoreMask” to control flagged issues). In addition to all standard PROC COMPARE checks that take place, the following additional items are also checked:

* Date/time of PROD datasets to be older than QC dataset.
* When checking a library – indication of datasets only in one library

## Single datasets compare

## Prerequisite.

All SAS datasets (PROD and QC) saved to the network, with standard names.  
If TFL dataset for compare, page variables dropped from the dataset before save.

## Example calls

**Dataset call**

%***s\_compare***(base=ADAM.ADSL,

comp=ADAMQC.Q\_ADSL);

**TFL Call**

%***s\_compare***(base=TFL.T14010101,

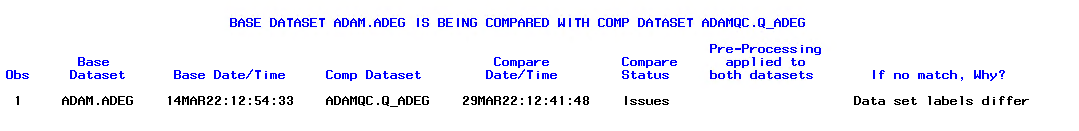
comp=TFLQC,Q\_T14010101,

TFL=Y);

**Note**: see below for detail on macro parameter TFL

## Result

A standard PROC COMPARE report is presented to the users OUTPUT window. Additionally, a summary PROC PRINT of the overall result is also generated to ensure there is no misinterpretation of the result. E.g.



## Library compares

## Prerequisite.

All SAS datasets (PROD and QC) saved to the network with standard names. QC datasets must have same name as their production “partner” with standard prefix “Q\_”.  
If TFL datasets being compared, page variables dropped from the dataset before save to the library and any split characters must be consistent across the whole library

## Example calls.

**Dataset call (via QC program FINAL.SAS)**

%***s\_compare***(base=ADAM.\_ALL\_,

comp=ADAMQC.\_ALL\_,

prefix=Q\_,

comprpt =%str(&\_\_outpath\%str(&\_studyid)\_11.03**.02**\_%sysfunc(date(),yymmddn.)  
 \_&\_relabel.-&\_\_PROGTYPE.-Compare-Summary.pdf)

**TFL compare (via QC program FINAL.SAS)**

%***s\_compare***(base=TFL.\_ALL\_,

comp=TFLQC.\_ALL\_,

tfl=Y,

prefix=Q\_,

splitchar=\*,

comprpt =%str(&\_\_outpath\%str(&\_studyid)\_11.03**.02**\_%sysfunc(date(),yymmddn.)  
 \_&\_relabel.-&\_\_PROGTYPE.-Compare-Summary.pdf)

**Note on key parameters:**

*TFL=Y*: Instructs compare macro to supress messages in PDF report when the follow differences are identified.

* Data set labels differ
* Data set types differ
* Variable has different informat
* Variable has different format
* Variable has different length

Additionally, SAS function COMPBL is applied to all character variables to negate the need for validator to take time aligning data to match (all values must however still be matched)

*Splitchar=\**: Removes the declared split character from any variable labels or character variables, then preventing the need for the QC programmer to purely replicate this character location. Remaining text in any string must still match.

For Additional details on all macro parameters. See macro header for details

## Result

A standard PROC COMPARE report for every single output will be created in the FINAL.LST output file. Additionally, a PDF report for all datasets compared is created and stored in the OUTPUT folder relating to the FINAL.SAS program being executed.

Table

Description automatically generated