

PharmaSUG China 2025 – Paper AA-158

Efficient Program Status Management in Clinical Trials: A PowerShell-Based Automation Tool

Hongmei Deng, Daiichi Sankyo (China) Holdings Co, LTD

ABSTRACT

Independent double programming is widely recognized as the gold standard for ensuring the accuracy of analytical results in the pharmaceutical industries.

To improve efficiency in managing bilateral processes during clinical trials, this paper introduces a tool developed by PowerShell that automates the integration of owner of each program, program logs, and comparison status information into a single, comprehensive summary page. With this streamlined solution, project programmers can efficiently monitor and manage their program status without the inconvenience of navigating multiple files.

INTRODUCTION

This tool is designed to optimize user experience by seamlessly integrating multiple information sources into a single page. Additionally, it provides fundamental features such as categorical counting and result filtering for improved data management. As demonstrated in the image below, the table header presents a summary of the dataset comparison, categorizing and calculating the total counts of “Pass” and “Fail” dataset comparisons, alongside the number of datasets that remain “Not Developed”, enabling users to easily track progress and outcomes.

Compare Status Summary

Executed by: denghmik

Execution Time: 2025-07-20 10:44:05

Target datasets: Total: 20 | Pass: 8 | Fail: 3 | Not Developed: 9

OBS	Dataset	Primary Programmer	Validation Programmer	Compare Status	Compare Report	Primary Log Status	Validation Log Status
Search...	Search...	Search...	Search...	Search...	Search...	Search...	Search...
1	ADAE	Hongmei	Chi	Pass	compare_adae.html	unclean, issue: logcheck_primary.html	clean: logcheck_validation.html
2	ADCM	Shiyu	Chi	Pass	compare_adcm.html	unclean, issue: logcheck_primary.html	unclean, issue: logcheck_validation.html
3	ADCV	Shiyu	Hongmei	Pass	compare_adcv.html	clean: logcheck_primary.html	unclean, issue: logcheck_validation.html
4	ADDV	Shiyu	Hongmei	Not Developed			
5	ADEG	Hongmei	Chi	Pass	compare_adeg.html	unclean, issue: logcheck_primary.html	unclean, issue: logcheck_validation.html
6	ADEX	Chi	Hongmei	Fail	compare_adex.html	unclean, issue: logcheck_primary.html	unclean, issue: logcheck_validation.html
7	ADGF	Chi	Shiyu	Not Developed			

Figure 1.Result Interface

The most user-friendly feature is that users can filter the information of interest using the search box under each column. For example, if users want to know about the datasets related to the programmer of primary side “Hongmei”, then can simply search for "Hongmei" in the search box under the "Primary Programmer" column to get the filtered results. Meanwhile, users can obtain detailed information on the comparison results and log checks information for each dataset by clicking the hyperlinks on the page.

Compare Status Summary

Executed by: denghmk

Execution Time: 2025-07-20 10:44:05

Target datasets: Total: 20 | Pass: 8 | Fail: 3 | Not Developed: 9

OBS	Dataset	Primary Programmer	Validation Programmer	Compare Status	Compare Report	Primary Log Status	Validation Log Status
Search...	Search...	hongmei	Search...	Search...	Search...	Search...	Search...
1	ADAE	Hongmei	Chi	Pass	compare_adae.html	unclean, issue: logcheck_primary.html	clean: logcheck_validation.html
5	ADEG	Hongmei	Chi	Pass	compare_adeg.html	unclean, issue: logcheck_primary.html	unclean, issue: logcheck_validation.html
9	ADLB	Hongmei	Chi -> Shiyu	Pass	compare_adlb.html	unclean, issue: logcheck_primary.html	unclean, issue: logcheck_validation.html
20	ADVS	Hongmei	Chi -> Shiyu	Fail	compare_adv.html	unclean, issue: logcheck_primary.html	unclean, issue: logcheck_validation.html

Figure 2. Filtered Result

A detailed explanation of the structure of this summary table, along with the origins of the information it presents, will be provided in the sections below. The first four columns represent the target datasets in the project, along with the respective bilateral programmers. This information is pulled from the project's tracker file.

Following these, two columns of “Compare Status” and “Compare Report” show the comparison status and corresponding result files for the bilateral datasets. The data in these columns comes from the comparison result files of each dataset, generated by SAS under the validation path.

The last two columns of the summary table display the program’s log status and the corresponding log report file. The information in these columns is sourced from the SAS log files located under the bilateral program path.

DEVELOPMENT AND OPERATION OF THE TOOL

PowerShell is a tool that comes pre-installed with Microsoft systems. It provides capabilities for managing and automating the operating system, applications, and other services. Users can open the PowerShell development tool by searching for PowerShell in the Windows operating interface.

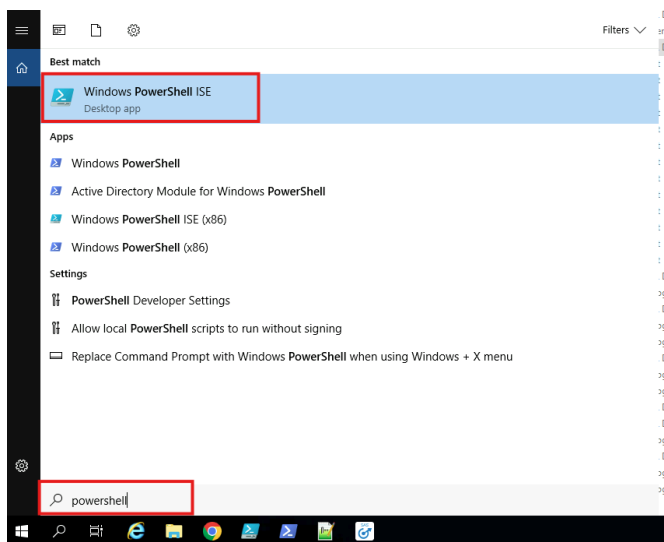


Figure 3. Development Tool: Powershell ISE

The image below presents the code development environment. The white window on top is the interface for writing code, and the window at the bottom displays the log of the code execution.

```

1
2 Add-Type -AssemblyName "System.Windows.Forms"
3
4 # create the window
5 $form = New-Object System.Windows.Forms.Form
6 $form.Text = 'Study Compare Status-File selection and Operation'
7 $form.Size = New-Object System.Drawing.Size(900, 400)
8
9 # create the first selection path
10 $labelPath1 = New-Object System.Windows.Forms.Label
11 $labelPath1.Text = "Primary path 1:"
12 $labelPath1.Location = New-Object System.Drawing.Point(10, 20)
13 $form.Controls.Add($labelPath1)
14
15 $path1Button = New-Object System.Windows.Forms.Button
16 $path1Button.Text = "Browse"
17 $path1Button.Location = New-Object System.Drawing.Point(150, 20)
18 $form.Controls.Add($path1Button)
19
20 $path1TextBox = New-Object System.Windows.Forms.TextBox
21 $path1TextBox.Location = New-Object System.Drawing.Point(300, 20)
22 $path1TextBox.Width = 500
23 $form.Controls.Add($path1TextBox)
24
25 # create the second selection path
26 $labelPath2 = New-Object System.Windows.Forms.Label
27 $labelPath2.Text = "Validation path 2:"

```

```

PS C:\Users\denghmtk> C:\Users\denghmtk\Desktop\Compare_Status.ps1
log Report generated for Primary side program at: S:\DS3201-330_wks_bdmprog\dev\programs\sdtm\routine\primary
log Report generated for Validation program at: S:\DS3201-330_wks_bdmprog\dev\programs\sdtm\routine\validation
the compare status summary report has been output to path : S:\DS3201-330_wks_bdmprog\dev\programs\sdtm\routine\validation\Summary_Report_20250723_140652.html
Cancel
PS C:\Users\denghmtk>

```

Figure 4. Code Development Environment

Upon completion of the code, it should be saved with a .PS1 file extension, signifying that it is a PowerShell script.

Name	Date modified	Type	Size
 Compare_Status	7/20/2025 8:52 AM	Windows PowerShell Script	32 KB

Figure 5. PowerShell Script Display

The operation of this tool is simple. Users just need to right-click the PowerShell script and click "Run with PowerShell" to bring up the interface, as shown in the image below.

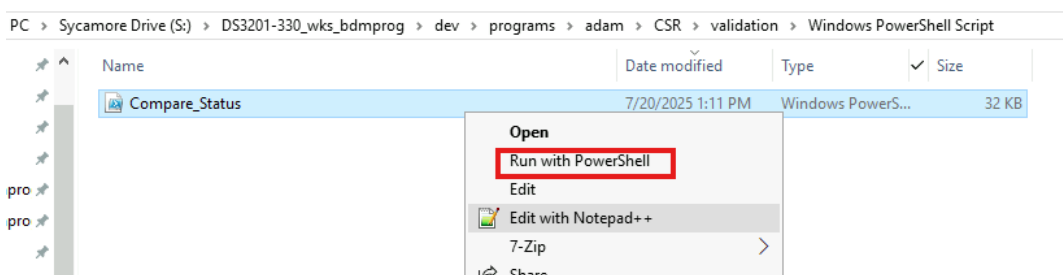


Figure 6. Operational : Right-click and Click "Run with PowerShell"

Users need to select the path of the bilateral program, choose the tracker file, and select the task type, such as ADAM. After clicking the "Run" button, the script will begin executing.

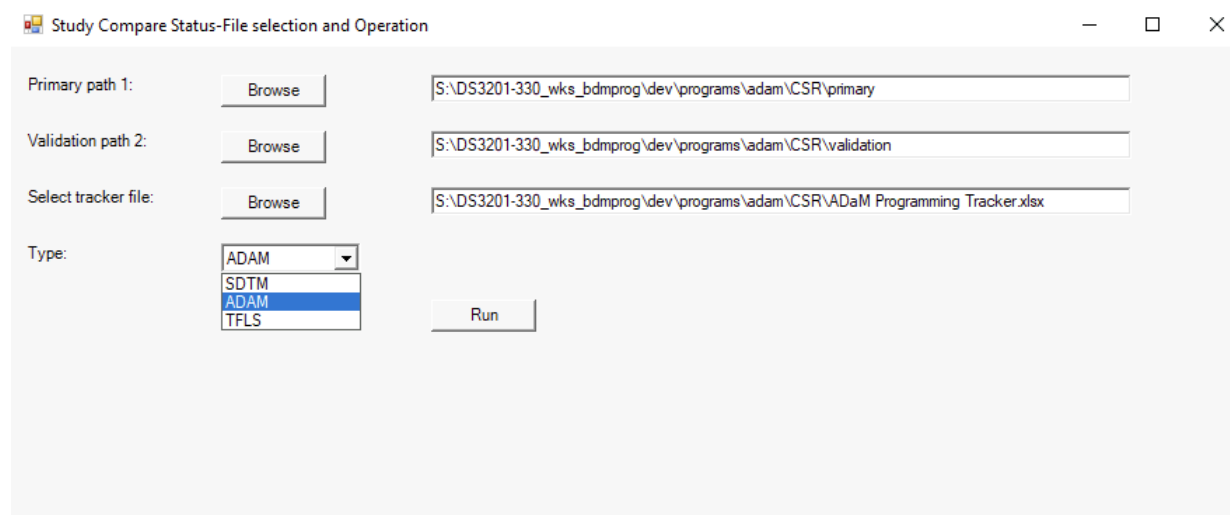


Figure 7.Operational Interface

The execution process of this script tool can be divided into three steps, and the specific processing rules for each part will be explained in detail in the following sections.

INFORMATION FROM TRACKER FILE

The first step of this tool is to obtain the path of the tracker file based on the tracker file and type information selected by the user in the interface. The selected type value is used to match the corresponding sheet in the tracker file.

Once the tracker file and sheet name are determined, we can begin searching and locating the required data within the sheet. The first step is to locate the header by Fuzzy search. When a row in the sheet contains the words "dataset," "primary programmer," and "validation programmer" in three separate cells, the tool treats this row as the header and extracts the data below these three cells. These three columns of data form the main table, which will later serve as the foundation for merging supplementary information.

A significant advantage of this flexible search and extraction method is its ability to adapt to tracker files in various formats. For instance, the tool is capable of handling tracker files organized in different layouts. Two examples of tracker files in different formats are provided below.

A	B	C	D	E	F	G	H
Dataset	Primary Program name	Primary programmer	Primary Program date complete	validation Programmer	validation Program name	validation Program date complete	Status
ADSL	adsl.sas	Chi	3/21/2025	Chaithanya	v_adsl.sas		Passed QC
ADAE	adae.sas	Hongmei	4/17/2025	Chi	v_adae.sas	5/14/2025	Passed QC
ADCM	adcm.sas	Shiyu	4/16/2025	Chi	v_adcm.sas	5/6/2025	Passed QC
ADCV	adcv.sas	Shiyu	4/16/2025	Hongmei	v_adcv.sas		pending
ADEG	adeg.sas	Hongmei	4/28/2025	Chi	v_adeg.sas	6/10/2025	Passed QC
ADEX	adex.sas	Chi	3/28/2025	Hongmei	v_adex.sas		Passed QC
ADXSUM	adexsum.sas	Chi	3/31/2025	Hongmei	v_adexsum.sas		Passed QC
ADLB	adlb.sas	Hongmei	5/16/2025	Chi --> Shiyu			Passed QC
ADMH	admh.sas	Shiyu	4/16/2025	Chi		5/6/2025	Passed QC
ADPC	adpc.sas	Shiyu		Hongmei			
ADPP	adpp.sas						
ADRS	adrs.sas	Chaithanya		Hongmei			
ADTR	adtr.sas	Shiyu	7/1/2025	Hongmei			Ready for QC
ADVS	adv.sas	Hongmei	4/28/2025	Chi --> Shiyu			Passed QC
ADTTE	adtte.sas	Chaithanya		Hongmei			
ADTM	adtm.sas	Chi	4/24/2025	Chi			Passed QC

Figure 8.Tracker File layout 1

[illegible]

Figure 9.Tracker File layout 2

Compare Status Summary

Executed by: denghmtk

Execution Time: 2025-07-27 05:56:28

Target datasets: Total: 3 | Pass: 3 | Fail: 0 | Not Developed: 0

OBS	Dataset	Primary Programmer	Validation Programmer	Compare Status	Compare Report	Primary Log Status		Validation Log Status	
Search...	Search...	Search...	Search...	Search...	Search...	Search...		Search...	
1	ADAE	test-a	test-b	Pass	compare_adae.html	exist issue:	logcheck_primary.html	clean:	logcheck_validation.html
2	ADCM	test-a	test-b	Pass	compare_adcm.html	exist issue:	logcheck_primary.html	exist issue:	logcheck_validation.html
3	ADSL	test-a	test-b	Pass	compare_adsl.html	exist issue:	logcheck_primary.html	exist issue:	logcheck_validation.html

INFORMATION OF COMPARE STATUS

The comparison information is derived from the HTML format comparison files located under the validation paths selected by the user in the interface, with each comparison file corresponding to a dataset. This means that before using the tool, we need to ensure that the comparison results of the datasets are output in HTML format to the validation path after running the each SAS program on validation side. Figure 11 below displays the macro code uniformly used by all SAS programs on the validation side, while Figure 12 shows the HTML-formatted compare result file located in the validation path.

```
%macro mcomp(lib=, dom=);  
  
ods html file="%pgloc/compare_&dom..html" ;  
proc compare base=&lib..&dom.(encoding=ANY) compare=v&lib..v&dom.(encoding=ANY) listall criterion=0.00000000001 method=absolute  
out=diff_&dom. outbase outcomp outnoequal outdif error ;  
run;  
ods html close ;  
  
%mend;
```

Figure 11.Output the compare result to validation path

Name	Date modified	Type	Size
 compare_lb	7/9/2025 1:56 PM	Chrome HTML Document	286 KB
 compare_supplb	7/9/2025 1:56 PM	Chrome HTML Document	97 KB
 compare_rs	7/9/2025 1:56 PM	Chrome HTML Document	60 KB
 compare_supprs	7/9/2025 1:56 PM	Chrome HTML Document	42 KB
 compare_pe	7/9/2025 1:56 PM	Chrome HTML Document	106 KB

Figure 12.Html format of compare files under validation path

The tool automatically scans all HTML files in the validation path and extracts the characters following the underscore in the filenames. For instance, for the file 'compare_ae.html,' it will extract 'ae' as the matching keyword. It then matches this keyword with the values listed under the "dataset" column in the previous tracking file. If a dataset exists in the project tracker file but a corresponding compare HTML file cannot be found, the dataset is considered 'Not Developed.' However, if a match is found, the HTML file link will be inserted into the main table.

It is worth noting that when the task type selected by the user in the interface is SDTM, supplementary datasets such as SUPPAE will not appear in the tracking file. However, we still want to monitor the quality control status of these supplementary datasets. Therefore, when matching datasets in the tracking file with HTML files, the tool performs a full join, allowing relevant information about supplementary datasets like SUPPAE to be retained in the results page.

Compare Status Summary

Executed by: denghmtk

Execution Time: 2025-07-20 09:21:10

Target datasets: Total: 39 | Pass: 16 | Fail: 16 | Not Developed: 7

SUPP datasets: Total: 15 | Pass: 6 | Fail: 9

OBS	Dataset	Primary Programmer	Validation Programmer	Compare Status	Compare Report	Primary Log Status	Validation Log Status
Search...	Search...	Search...	Search...	Search...	Search...	Search...	Search...
1	AE	Kumaran/Chi	Hongmei	Fail	compare_ae.html	unclean, issue: logcheck_primary.html	unclean, issue: logcheck_validation.html
1	SUPPAE	Kumaran/Chi	Hongmei	Fail	compare_suppa.html		
2	CM	Hongmei	Kumaran/Chi	Fail	compare_cm.html	unclean, issue: logcheck_primary.html	unclean, issue: logcheck_validation.html
2	SUPPCM	Hongmei	Kumaran/Chi	Fail	compare_suppcm.html		
3	CO			Not Developed			unclean, issue: logcheck_validation.html
4	CV	Hongmei	Kumaran/Chi	Pass	compare_cv.html	unclean, issue: logcheck_primary.html	unclean, issue: logcheck_validation.html
5	DD	Hongmei	Lu Chi	Pass	compare_dd.html	unclean, issue: logcheck_primary.html	unclean, issue: logcheck_validation.html
6	DM	Kumaran/Chi	Hongmei	Fail	compare_dm.html	unclean, issue: logcheck_primary.html	unclean, issue: logcheck_validation.html
6	SUPPDM	Kumaran/Chi	Hongmei	Fail	compare_suppdm.html		

Figure 13.Result for the SDTM Supplementary Datasets

After matching these two pieces of information, the tool will begin reading and scanning the content of the HTML comparison file corresponding to each dataset. If "Number of Observations with Some Compared Variables Unequal: 0" is found within the file, at the same time, there were no occurrences of unequal numbers of variables and attributes, such as the phrases "Listings of variables in" or "but not in", it indicates that the dataset's QC status is "Pass." Otherwise, the quality control status will be marked as "Fail."

INFORMATION OF PROGRAM LOG STATUS

The log information is sourced from the SAS log files located in the two paths selected by the user in the interface. The tool automatically scans the log files in these two paths and generates summary reports for the log files, which are then saved in the respective paths. Below is a screenshot displaying the summary report of the log files under the primary path, where users can observe the overall log information in the header, as well as statistics for the clean program logs. Additionally, a search function has been integrated for each column, allowing users to search for the corresponding dataset logs and retrieve the relevant log information.

File
S:\DS3201-330_wks_bdmprog\dev\programs\adam\CSR\primary\logcheck_primary.html

Logcheck Report

Executed by: SYCAMORE\denghmtk

Execution Time: 2025-07-23 14:23:45

Summary:

Total Log Files: 15

Log Files Marked as "Clean": 2

Search Results:

Obs	Filename	Target Word	Matched Line
Search	adex.log	Search Target Word...	Search Matched Line...
5	adex.log	W.D FORMAT	NOTE: At least one W.D format was too small for the number to be printed. The decimal may be shifted by the "BEST" format.
5	adex.log	QUERY REQUIRES REMERGING	NOTE: The query requires remerging summary statistics back with the original data.
5	adex.log	MISSING VALUES WERE GENERATED	NOTE: Missing values were generated as a result of performing an operation on missing values.
5	adex.log	UNINITIAL	NOTE: Variable TRT01P is uninitialized.
5	adex.log	UNINITIAL	NOTE: Variable TRT01A is uninitialized.
5	adex.log	UNINITIAL	NOTE: Variable TRT01PN is uninitialized. The same target word has appeared more than three times, the rest are not displayed.

Figure 14.Logcheck Report

The tool searches through each log file line by line for our target words, such as “uninitialized”, which is one of our target words. If the target word is found in the log file, the log report will list the target word along with the corresponding line where it appears in the log.

If multiple target words are found in the log file, each target word will be summarized in the log report. However, if a target word appears more than three times in the same log file, to avoid making the log file too lengthy, the matching lines after the third occurrence of that target word will not be displayed in the log report. Instead, a reminder will be added: "The same target word has appeared more than three times, the rest are not displayed." If no target words are found in a log file, the log report will display that the log file is "log is clean."

Once the log reports are generated for both paths, the second step of the tool is to match the filenames in the log reports with the corresponding datasets from the processed tracker file. If a match is found, a hyperlink will be inserted into the log report. The tool will then read the status of each log file in the log report. If "log is clean" is found, the corresponding dataset's log file status will be considered clean; otherwise, it will be marked as unclear.

Finally, upon completion of all steps, the tool will display a notification in the menu bar to alert the user. As shown in the image below, at this point, the user can close the window and navigate to the selected validation path to open the HTML result file for reviewing the project status.

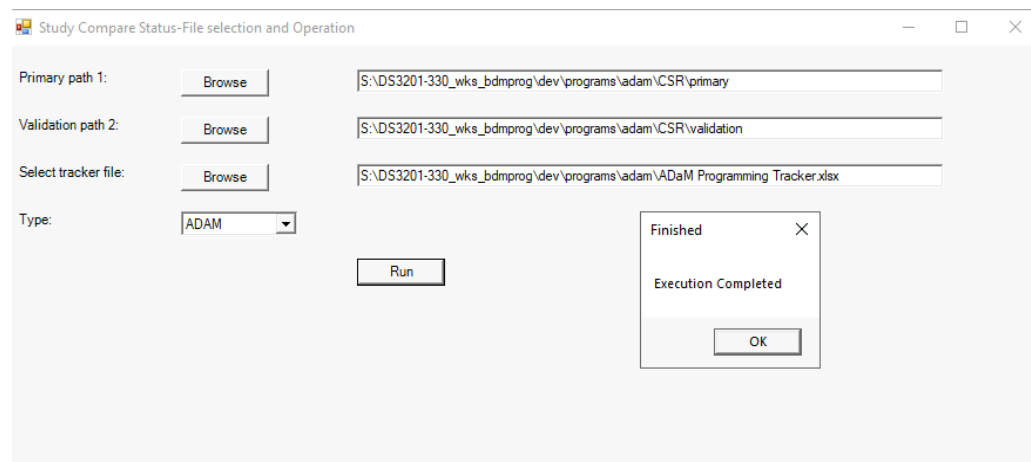


Figure 15. Execution Completed

CONCLUSION

PowerShell is a powerful scripting language and automation framework for system management and configuration, excelling in task automation, remote management, and object manipulation in Windows systems. This paper explains how to use PowerShell to create a tool that manages the bilateral program status of clinical trial projects, consolidating target datasets, bilateral programmers, program comparison statuses, and log information into an interactive interface to effectively display the project's current status.

ACKNOWLEDGE

I would like to express my gratitude to Xuan Yang for her comments and support.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Name: Hongmei Deng

E-mail: 467036370@qq.com