Paper CT18

# Automated Comparison of EDC Transfers: Identifying Data Changes Across Versions

Malavika B, Catalyst Flex, Kochi, India
Vignesh V Gopal, Catalyst Flex, Kochi, India
Rahulraj Raju, Catalyst Flex, Kochi, India

## ABSTRACT

Electronic Data Capture (EDC) systems play a vital role in clinical trials by streamlining the collection, storage, and management of data. Ensuring the accuracy and integrity of EDC data is crucial for producing reliable results that meet regulatory standards. While data management team validate data during EDC migrations, programmers encounter unique challenges when handling downstream datasets used for statistical analysis and reporting. Changes in the metadata of revised EDC datasets will result in failing the SDTM programs and taking time to debug the issues. This paper presents a SAS macro developed specifically to support programmers in identifying EDC changes across versions. The macro facilitates the automated detection of changes, validation of data consistency, and early identification of anomalies across data refreshes. By comparing datasets from different time points and clearly highlighting discrepancies, the macro becomes an essential component of efficient and reliable programming workflows.

## INTRODUCTION

In modern clinical trials, frequent EDC data refreshes are common due to ongoing data entry, query resolution, protocol amendments, and database updates. Each new transfer is expected to reflect incremental changes, however, unanticipated modifications to metadata such as variable attributes, lengths, formats, labels, controlled terminology, or dataset structures can significantly impact downstream programming activities.

Although data management teams perform validation during EDC migrations, programmers are often the first to encounter issues when programs fail or generate unexpected results following a data refreshment. Identifying the root cause of these issues manually is time-consuming and error-prone, particularly in large oncology studies or integrated analyses with multiple domains and frequent transfers.

To address these challenges, an automated, standardized, and programmer-friendly solution is required to compare EDC transfers efficiently and transparently. This paper describes the design and implementation of a SAS macro developed to automate EDC comparison and support early detection of changes before they impact SDTM, ADaM, and TFL deliverables.

## CHALLENGES WITH EDC DATA REFRESHES

Programmers commonly encounter several challenges when working with revised EDC datasets:

- Metadata changes such as updates to variable type, length, format, or label
- Addition or removal of CRF variables or entire datasets
- Changes in controlled terminology or value representations requiring updates to decode logic
- Structural changes in repeating forms affecting record counts
- Unexpected shifts in missingness or value distributions
- Lack of clear documentation describing what has changed between transfers

These issues often lead to SDTM mapping failures, data truncation, or inconsistencies in downstream outputs, increasing the risk of delays and rework.

## SOLUTION OVERVIEW

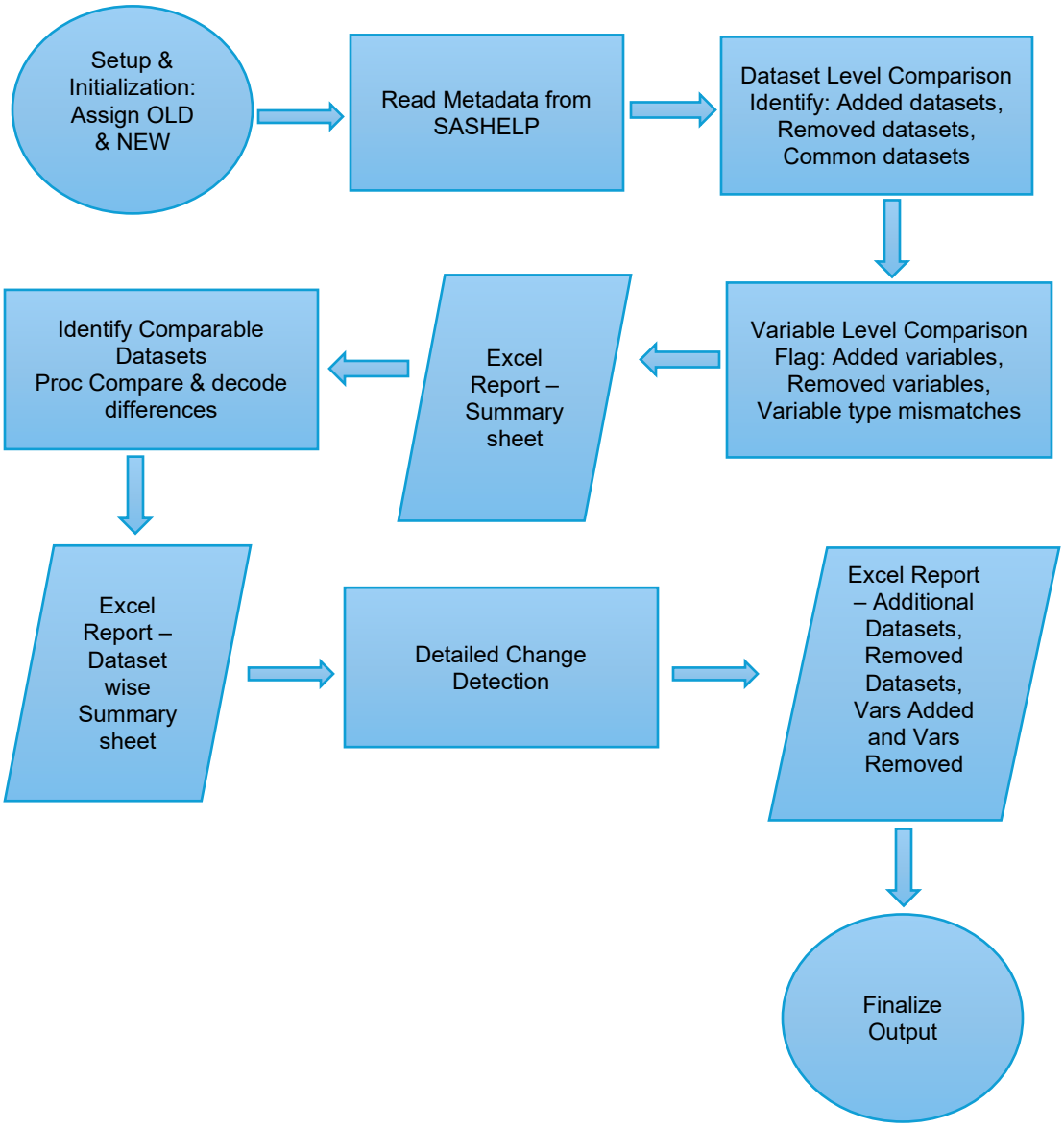
To address these challenges, a SAS macro was developed to automate the comparison of EDC datasets across different transfer versions. The macro is designed to be flexible, reusable, and easy to implement across studies and therapeutic areas.

The solution focuses on identifying both metadata-level and data-level changes and presenting them in a structured and review-friendly format. This allows programmers to quickly assess the impact of EDC changes and take appropriate action before executing the programs.

## MACRO DESIGN AND FUNCTIONALITY

The SAS macro is designed to compare two EDC transfers, referred to as the OLD and NEW versions, using SAS libraries. The macro performs a systematic comparison at both the dataset and variable levels and produces a comprehensive Excel-based comparison report.

The macro leverages SAS metadata tables (SASHELP.VTABLE and SASHELP.VCOLUMN), PROC COMPARE, and dynamic macro programming techniques to ensure scalability across multiple datasets and transfers.




### DYNAMIC MACRO LOGIC AND PROCESSING APPROACH

The macro is implemented using dynamic SAS macro programming to enable automated comparison of EDC transfers without requiring manual updates when datasets or variables change between refreshes. At runtime, the macro assigns library references for the OLD and NEW EDC transfers and defines reusable macro variables to exclude non-informative tracking variables and to retain key identifier variables required for accurate record alignment.

Rather than hardcoding dataset and variable lists, the macro dynamically interrogates SAS metadata tables (SASHELP.VTABLE and SASHELP.VCOLUMN) to identify available datasets, observation counts, variable counts, and variable attributes in both transfers. Dataset-level comparisons are driven by merging metadata from the OLD and NEW libraries, allowing the macro to automatically detect newly added datasets, removed datasets, and mismatches in observation or variable counts. This metadata-driven approach ensures robustness when EDC structures evolve across refreshes.

For datasets present in both transfers, the macro performs variable-level metadata comparisons by merging variable attributes across versions. The logic dynamically flags added variables, removed variables, and variable type changes that may impact downstream SDTM mapping. Only datasets common to both transfers are processed further, improving performance and preventing unnecessary errors.

Data-level comparisons are executed iteratively using macro loops that dynamically process each dataset identified in the metadata step. The macro conditionally skips datasets with zero observations and applies dataset-specific key structures. Non-essential EDC tracking variables are excluded dynamically to focus comparisons on clinically meaningful data.

The macro captures the SYSINFO return code from PROC COMPARE for each dataset and decodes it using binary logic to classify the type of differences detected, including observation-level, variable-level, and value-level discrepancies. These decoded results are stored as dataset-wise summary indicators, enabling automated and interpretable reporting without manual inspection of PROC COMPARE output.

To support detailed change tracking, the macro dynamically constructs variable rename statements and conditional comparison code based on shared variable metadata. This allows identification of record-level value changes and generation of dataset-specific change listings that clearly distinguish new, removed, and modified records. All results are consolidated and written to a structured, multi-sheet Excel report using ODS EXCEL, providing a comprehensive and auditable summary of EDC changes across transfers.

Key considerations during development included scalability for large complex studies, flexibility to accommodate evolving EDC structures, performance optimization through metadata filtering, and clarity of outputs for programmer review. By leveraging dynamic macro variables, metadata-driven logic, and conditional processing, the macro provides a robust and reusable solution for identifying EDC changes before they impact downstream SDTM, ADaM, and TFL deliverables.

**MACRO SETUP AND INPUT LIBRARIES**

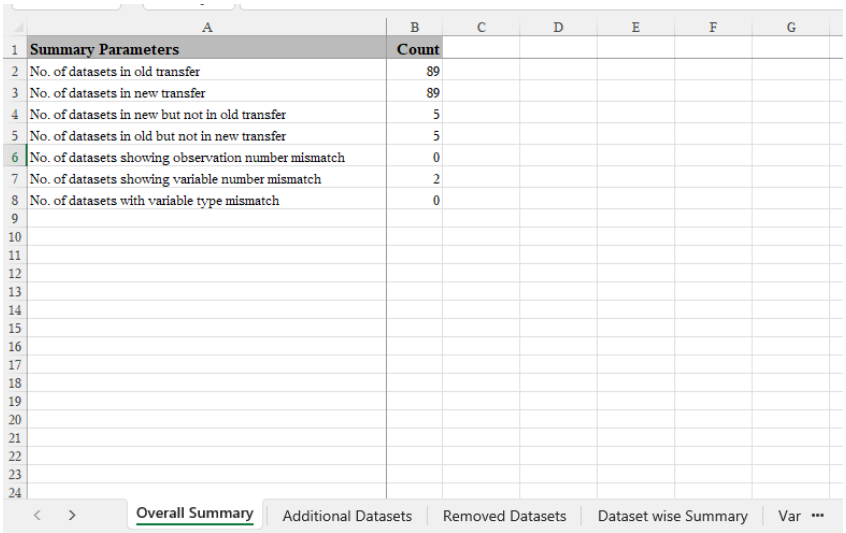
The macro initializes library references for the OLD and NEW EDC transfers representing different data refreshes. Standard EDC tracking and audit variables that do not contribute to meaningful comparison are excluded using predefined macro variable lists, ensuring consistent and focused dataset alignment. Key identifier variables are retained to align records correctly during data-level comparisons

**DATASET-LEVEL COMPARISON**

At the dataset level, the macro compares the number of datasets present in each transfer and identifies:

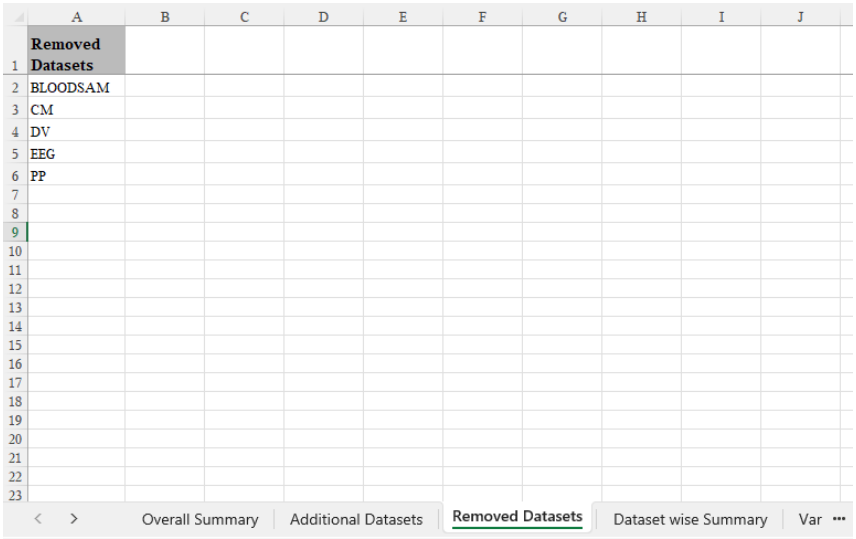
- Datasets newly introduced in the current transfer
- Datasets removed since the previous transfer
- Differences in observation counts
- Differences in variable counts

This comparison is performed using metadata from SASHELP.VTABLE and summarized into a high-level overview report



| | A | B | C | D | E | F | G |
|---|---|---|---|---|---|---|---|
| 1 | **Summary Parameters** | **Count** | | | | | |
| 2 | No. of datasets in old transfer | 89 | | | | | |
| 3 | No. of datasets in new transfer | 89 | | | | | |
| 4 | No. of datasets in new but not in old transfer | 5 | | | | | |
| 5 | No. of datasets in old but not in new transfer | 5 | | | | | |
| 6 | No. of datasets showing observation number mismatch | 0 | | | | | |
| 7 | No. of datasets showing variable number mismatch | 2 | | | | | |
| 8 | No. of datasets with variable type mismatch | 0 | | | | | |
| 9 | | | | | | | |
| 10 | | | | | | | |
| 11 | | | | | | | |
| 12 | | | | | | | |
| 13 | | | | | | | |
| 14 | | | | | | | |
| 15 | | | | | | | |
| 16 | | | | | | | |
| 17 | | | | | | | |
| 18 | | | | | | | |
| 19 | | | | | | | |
| 20 | | | | | | | |
| 21 | | | | | | | |
| 22 | | | | | | | |
| 23 | | | | | | | |
| 24 | | | | | | | |

Overall Summary | Additional Datasets | Removed Datasets | Dataset wise Summary | Var



| Removed Datasets |
|---|
| BLOODSAM |
| CM |
| DV |
| EEG |
| PP |

Overall Summary | Additional Datasets | Removed Datasets | Dataset wise Summary | Var

| New Datasets |
|---|
| AE_DAT |
| EG |
| PK_APS |
| PK_EVENAMIDE_BPAA |
| RAND_BLIND |

Overall Summary | Additional Datasets | Removed Datasets | Dataset wise Summary | Var

**VARIABLE-LEVEL METADATA COMPARISON**

For datasets present in both transfers, the macro performs a detailed variable-level comparison using SASHELP.VCOLUMN. The following attributes are evaluated:

- Variable presence (added or removed variables)
- Variable type differences (character versus numeric)
- Changes that may impact downstream SDTM mapping

These checks enable early identification of metadata changes that frequently cause SDTM program failures.

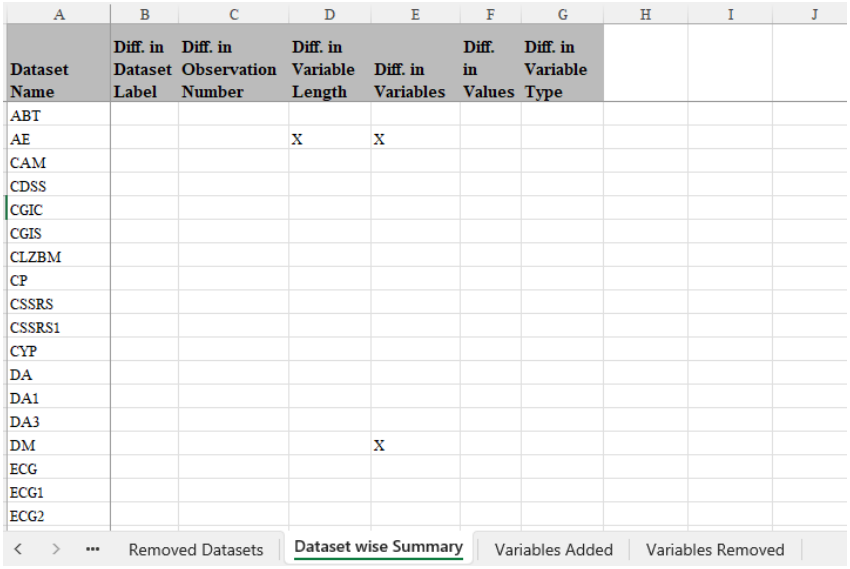
**DATA-LEVEL COMPARISON USING PROC COMPARE**

To assess data-level differences, the macro applies PROC COMPARE for each dataset using key identifier variables such as *subject, recordid*, and *recordposition*. Non-essential tracking variables are excluded to focus on meaningful clinical data changes.

The PROC COMPARE SYSINFO return code is captured and decoded to identify specific types of differences, including:

- Observation count differences
- Variable attribute differences
- Value-level change

This approach allows the macro to generate a concise dataset-wise summary indicating the nature of changes detected in each domain.



| Dataset Name | Diff. in Dataset Label | Diff. in Observation Number | Diff. in Variable Length | Diff. in Variables | Diff. in Values | Diff. in Variable Type |
|---|---|---|---|---|---|---|
| ABT | | | | | | |
| AE | | | X | X | | |
| CAM | | | | | | |
| CDSS | | | | | | |
| CGIC | | | | | | |
| CGIS | | | | | | |
| CLZBM | | | | | | |
| CP | | | | | | |
| CSSRS | | | | | | |
| CSSRS1 | | | | | | |
| CYP | | | | | | |
| DA | | | | | | |
| DA1 | | | | | | |
| DA3 | | | | | | |
| DM | | | | X | | |
| ECG | | | | | | |
| ECG1 | | | | | | |
| ECG2 | | | | | | |



**OUTPUT AND REPORTING**

All comparison results are written to a multi-sheet Excel report using ODS EXCEL. The report includes:

- Overall summary of datasets and discrepancies
- Lists of added and removed datasets
- Dataset-wise comparison summaries
- Variable-level metadata differences
- Detailed listings of new, removed, and changed records

These outputs provide programmers and data management teams with a clear and auditable view of EDC changes across transfers, facilitating rapid issue identification and resolution.

## BENEFITS AND IMPACT

The implementation of this automated EDC comparison macro provides several key benefits:

- Early detection of breaking changes before SDTM programs are executed
- Reduced time spent on manual debugging and investigation
- Improved transparency of EDC changes across transfers
- Standardization of comparison methods across studies and teams
- Enhanced collaboration between programming and data management functions

The macro has proven particularly useful in studies with frequent data refreshes and complex data structures.

## CONCLUSION

Frequent EDC data refreshes are an inherent part of clinical trial conduct, but unmanaged changes can significantly disrupt downstream programming workflows. The SAS macro described in this paper provides a practical and automated solution for identifying metadata and data-level changes across EDC transfers.

By integrating this approach into standard programming workflows, teams can proactively manage EDC changes, reduce rework and improve the overall reliability and efficiency of clinical trial data processing.

## REFERENCES

https://documentation.sas.com/doc/en/pgmsascdc/9.4_3.5/proc/n1jbbrf1tztya8n1tju77t35dej9.htm

## ACKNOWLEDGMENTS

We gratefully acknowledge the guidance and constructive feedback provided by our colleagues, mentors, and reviewers throughout the development of this paper. We also recognize the contribution of AI tools, such as ChatGPT, which assisted in organizing ideas and refining content, thereby supporting a more efficient writing process.

## CONTACT INFORMATION

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

Author Name: Malavika B
Company: Catalyst Flex
Address: 8B Tower 1, TransAsia, Cyber Park Infopark, Phase 2, Kochi, Kerala 682303
Work Phone:
Email: malavika.biju@catalystcr.com
Website: https://catalystcr.com/

Co-Author Name: Vignesh V Gopal
Company: Catalyst Flex
Address: 8B Tower 1, TransAsia, Cyber Park Infopark, Phase 2, Kochi, Kerala 682303
Work Phone:
Email: vignesh.gopal@catalystcr.com
Website: https://catalystcr.com/

Co-Author Name: Rahulraj Raju
Company: Catalyst Flex
Address: 8B Tower 1, TransAsia, Cyber Park Infopark, Phase 2, Kochi, Kerala 682303
Work Phone:
Email: rahulraj.raju@catalystcr.com
Website: https://catalystcr.com/