PharmaSUG China 2025 - Paper DV-181

Real-time SDTM Data Quality Check

Nana Yang, Chunpeng Zhao, BeOne Medicines

ABSTRACT

As the source data for Analysis Data Model (ADaM) and Tabular Listing (TFL), the data quality of SDTM is of utmost importance. Traditionally, most validations are centrally performed via the Pinnacle 21 or internally developed SAS macros before database lock, leading to the dilemma: SDTM programming issues cannot be identified and resolved immediately after raw data extraction. This delay causes issues to persist for extended periods, often only being resolved near the database locking deadline, which undermines the goal of making data available for immediate use.

To address such challenge, the paper introduces the concept and importance of real-time SDTM quality check, along with a specific, executable, and replicable technical implementation framework. Key components include Developing SAS macros that account for data access constraints, incorporating common validation rules that can be interactively enhanced, collaborating with the company's auto-run team to integrate the macro into the auto-run workflow. After raw data extraction, SDTM datasets are automatically refreshed, followed by macro execution, and check results—structured similarly to the Pinnacle 21 reports—are then auto-generated and delivered via email to relevant SDTM programmers and study lead. The email sending function leverages the python package including 'smtplib', 'ssl', and 'email' to implement a robust SMTP-based mailing module, supporting HTML formatting, attachment handling, CC configuration, priority flagging, and issue handling. Through such real-time auto-run checking approach, SDTM data quality issues can be identified and addressed within hours of raw data extraction on the same day, significantly enhancing data quality and enabling confidence in real-time refreshing and using SDTM datasets.

INTRODUCTION

In the realm of clinical data management, SDTM serves as the foundational source data for both the ADaM and TFL. Given its critical role in subsequent data analysis and reporting, the quality of SDTM data is of paramount importance. Any inaccuracies or inconsistencies within SDTM datasets can cascade through the entire data processing pipeline, leading to flawed analyses, incorrect conclusions, and potentially detrimental impacts on clinical trials and regulatory submissions.

THE LIMITATIONS OF TRADITIONAL SDTM DATA QUALITY VALIDATION

Historically, the majority of SDTM data validations have been carried out in a centralized manner. Typically, this involves using tools like the Pinnacle 21 or internally developed SAS macros, and these validations are predominantly executed just before the database lock or before a significant official submission, such as those to the FDA, EMA, PMDA, or NMPA. While this traditional approach has been the norm for many years, it presents significant drawbacks that undermine the efficiency and effectiveness of data management processes.

One of the most pressing issues with the traditional validation model is the substantial time lag between raw data extraction and the identification of SDTM programming issues. Once raw data is extracted, SDTM programmers begin the process of transforming and formatting the data according to the SDTM standards. However, issues in this programming process often go undetected until the batch validation is performed much later in the timeline. This delay means that issues can fester for an extended period, sometimes only being identified and addressed as the database locking deadline looms large or just before the deadline of a crucial official submission.

This late discovery of issues not only causes unnecessary stress and pressure on the scientific programming team but also has practical implications. For instance, late SDTM update may impact existing ADaM programming code or rushed fixes near the deadline increase the risk of introducing new issues during the correction process. Moreover, it disrupts the smooth flow of data-driven decision-making, as stakeholders cannot rely on the more accurate SDTM data for immediate analysis. This situation directly contradicts the overarching goal of having data readily available for use, which is crucial for timely clinical trial progress and informed decision-making.

THE CONCEPT AND SIGNIFICANCE OF REAL-TIME SDTM DATA QUALITY CHECK

To overcome the limitations of traditional validation methods, the concept of real-time SDTM data quality check emerges as a game-changer. Real-time SDTM data quality check refers to the immediate simplify validation of SDTM datasets as soon as raw data is extracted, enabling the rapid identification and resolution of data quality issues. This approach shifts the focus from reactive issue correction to proactive quality assurance, ensuring that SDTM data remains accurate and reliable throughout the data lifecycle.

The significance of real-time SDTM data quality check cannot be overstated. Firstly, it greatly enhances the overall data quality. By catching issues early, programmers have more time to correct them thoroughly, reducing the likelihood of residual issues. This leads to cleaner, more consistent SDTM datasets that serve as a solid foundation for downstream analyses. Secondly, it improves the

efficiency of the data management process. With real-time feedback, SDTM programmers can make immediate adjustments to their programming scripts, eliminating the need for time-consuming rework later. This streamlines the workflow and helps keep the project on schedule. Additionally, for downstream processes, real-time quality checks instill greater confidence in the data among stakeholders. They can trust that the SDTM data being used is up-to-date and accurate, enabling more informed decision-making at every stage of the clinical trial. For upstream processes, more accurate and standardized formatted SDTM data can serve as an important reference for the upstream department or team to support data monitoring and data cleaning, so as to inspect the quality of raw data in the future.

TECHNICAL IMPLEMENTATION FRAMEWORK OF REAL-TIME SDTM DATA QUALITY CHECK

The successful implementation of real-time SDTM data quality check relies on a well-structured technical framework. This framework encompasses several key components, each playing a vital role in ensuring the effectiveness of the real-time validation process.

Figure 1 illustrates the real-time SDTM data quality check process, encompassing the development of a global check macro, its application during the auto-run process or via the SDTM Digital Employee platform, automated delivery of check results to the study's SDTM programmer and leader with storage in the study folder, review of check messages and subsequent SDTM updates as needed, and feedback provision to enhance the global macro.

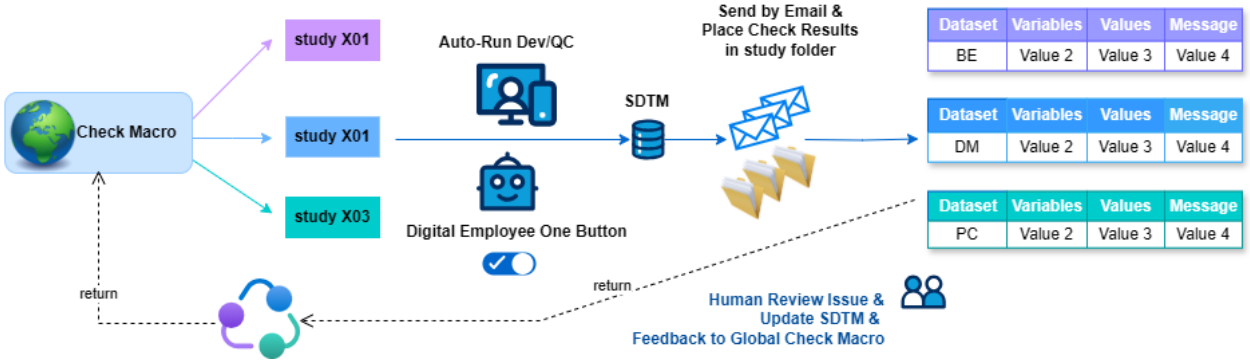

Figure 1. Process of Real-Time SDTM Data Quality Check

DEVELOPING SIMIPLFY VALIDATION SAS MACROS

Data Access Implementation:

The development of specialized SAS macros forms the core of the technical implementation. The macro is designed to perform a wide range of validation tasks on SDTM datasets. When creating the validation macro, it is essential to take into account various data access constraints. Different projects may have specific security requirements, data storage locations, and access permissions that need to be accommodated. By carefully considering these constraints, the macro can be tailored to work seamlessly within the existing data infrastructure.

In this regard, the operation process is as follows: place the macro at the global level in the same folder as other SDTM macros. No additional macro parameters are required. Only the macro variables set in the centralized environment of each study, along with a super account that has passed data access verification, are needed to read the location of the study's SDTM dataset. The advantage of this approach is that it enables convenient access to SDTM data without the need for separate settings in each study. When the macro execution is complete, the check results will be output to each study folder in the excel format and also show result in the real-time SDTM datasets fresh email message, and those without study permissions cannot access the check results.

Validation Rule Design:

In addition to handling data access, the SAS macro incorporates common validation rules. These rules cover aspects such as data type consistency, the presence of required variables and so on. What makes these macros particularly powerful is their ability to be interactively enhanced. As new validation requirements emerge or as existing rules need to be refined, the macros can be updated and extended without significant disruption. This flexibility ensures that the validation process remains relevant and effective over time.

Considering usage habits, the output structure is similar to the Pinnacle 21 structure. However, the data quality check focus on SDTM mapping issues and SDTM programming issues. Source data issues are excluded as many as possible. The benefit of this approach is that real-time checks will not impose additional burdens on SDTM programmers, while enabling timely detection and resolution of programming issues such as annotations during the real-time data process. In addition, since the macro serves all real-time fresh studies, data quality issues identified in one study can immediately be set as a check rule, which provides a good reference for other studies. If similar issues occur in other studies, they can be detected promptly at the first opportunity.

INTEGRATION WITH THE AUTO-RUN WORKFLOW AND DIGITAL EMPLOYEE PLATFORM

Collaboration with the company's auto-run team is another crucial aspect of the implementation framework. The goal is to integrate the developed SAS macros into the auto-run workflow. Once raw data is extracted, the auto-run system is configured to automatically refresh the SDTM datasets. This ensures that the datasets are based on the most current raw data.

Following the dev side datasets are refreshed and QC is completed for specific study, the check macro is executed automatically as part of the workflow. The macro analyzes the SDTM datasets and identifies any data quality issues. After the validation process is completed, the check results are generated in the table format in email and .xlsx format in the same subfolder location of study folder. These results are structured in a format similar to Pinnacle 21 report. This standardization makes it easy for stakeholders to understand and interpret the results. We also collaborate with the company's SDTM Digital Employee platform, enabling the macro to be executed not only in the auto-run process but also on-demand for each study delivery or whenever SDTM data rerun is required. The operation is highly user-friendly—by default, the button is activated, and the check output remains consistent with that of the auto-run process.

Figure 2 is a sample check result table in the email.

SDTM Issues:

Note: excel report in sdtm_rt/tools

Dataset	Variables	Values	Message
BE	BETERM		Null value in BETERM variable marked as Required
DS	DSDECOD		Null value in DSDECOD variable marked as Required
	DSTERM		Null value in DSTERM variable marked as Required

Figure 2. Check Result Figure in E-mail

AUTOMATED RESULT DELIVERY

To ensure that the data quality check results reach the relevant parties in a timely manner, an automated result delivery mechanism is implemented. The email sending function leverages the 'smtplib', 'ssl', and 'email' packages in Python to create a robust SMTP-based mailing module. This module is capable of sending emails with a high degree of functionality. It supports HTML formatting, allowing for more visually appealing and informative reports. Attachments can be easily included, such as detailed validation reports or issue logs. The module also enables the configuration of carbon copy (CC) recipients, ensuring that all relevant stakeholders, including SDTM programmers and study leads, receive the necessary information. Priority flagging can be set to indicate the severity of the data quality issues, helping recipients prioritize their response. Additionally, the module is equipped with comprehensive issue handling capabilities, ensuring that emails are sent successfully even in the face of network or server issues.

In Python, the 'smtplib', 'ssl', and 'email' packages are core libraries for handling email sending and receiving: 'smtplib' is a standard library module that implements an SMTP client for sending emails, enabling connection to SMTP servers via `SMTP()` or 'SMTP_SSL()', user authentication through 'login()', and email transmission with 'sendmail()'. The 'ssl' module provides TLS/SSL encryption to secure email data during transmission, creating secure contexts via 'SSLContext()' and verifying server legitimacy to prevent man-in-the-middle attacks. The 'email' package, a collection of modules for constructing, parsing, and processing email content, includes 'email.message' for creating objects supporting text, HTML, and attachments, 'email.mime' for building MIME-type multipart emails, and 'email.parser' for parsing received messages. Collaboratively, they enable a workflow where the 'email' package constructs content with subjects, senders, recipients, bodies, and attachments, 'ssl' creates encrypted contexts for secure connections to SMTP servers via 'smtplib', and 'smtplib' dispatches emails using 'sendmail()' or 'send_message()', thus implementing capabilities from simple text to complex HTML emails with attachments.

Here is a sanitized dummy code example:

```
def load_email_config():
    # Load email configuration from secure file
    config = configparser.ConfigParser()
    config.read('email_config.ini')
    return {
        'smtp_server': config.get('Email', 'smtp_server'),
        'port': config.getint('Email', 'port'),
```

```
        'sender_email': config.get('Email', 'sender_email'),
        'password': config.get('Email', 'password')
    }
def email(emailto, cc=None, subject="", content="", suc=True, files=None):
    # Set default values
    cc = cc or []
    files = files or []
    # Load config and process recipients
    config = load_email_config()
    recipients = emailto + [f"{c}@example.com" for c in cc]
    # Construct message
    msg = MIMEMultipart("alternative")
    msg["From"] = config['sender_email']
    msg["To"] = ', '.join(emailto)
    msg["Cc"] = ', '.join(cc) if cc else ''
    msg["Subject"] = subject
    msg["X-Priority"] = '0' if suc in [False, 0] else '1'
    # Build HTML content
    html = f"""
    <html><body><font>
        <p>{content}</p>
    </font></body></html>
    """
    msg.attach(MIMEText(html, "html"))
    # Attach files
    for file in files:
        with open(file, "rb") as f:
            part = MIMEBase("application", "octet-stream")
            part.set_payload(f.read())
            encoders.encode_base64(part)
            part.add_header(
                "Content-Disposition",
                f"attachment; filename={os.path.basename(file)}"
            )
            msg.attach(part)
    # Send email with error handling
     with smtplib.SMTP(config['smtp_server'], config['port']) as server:
        server.starttls(context=ssl.create_default_context())
        server.login(config['sender_email'], config['password'])
        server.sendmail(config['sender_email'], recipients, msg.as_string())
```

CONCLUSION

In conclusion, real-time SDTM data quality check represents a significant advancement in clinical data management. By addressing the limitations of traditional validation methods, it provides a more efficient, effective, and reliable approach to ensuring the quality of SDTM data. The technical implementation framework, consisting of customized SAS macros, integration with the auto-run workflow, and automated result delivery, offers a practical and replicable solution for organizations looking to enhance their data quality processes. As the importance of accurate and timely data continues to grow in the clinical research landscape, real-time SDTM data quality check is poised to become an essential practice for data management teams worldwide.

REFERENCES

Danny Hsu etc., 2025. "Automating Preliminary CDISC SDTM Compliance Checks With a Customized SAS Macro." https://phuse.s3.eu-central-1.amazonaws.com/Archive/2025/Connect/US/Orlando/PAP_PP19.pdf

Marisa Wyckmans etc., 2025. "Our Road to Adopting CORE: an Implementation Journey in SDTM Dataset Validation." https://phuse.s3.eu-central-1.amazonaws.com/Archive/2024/Connect/EU/Strasbourg/PAP_SI03.pdf

CONTACT INFORMATION

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

Nana Yang

BeOne Medicines

nana.yang@beigene.com

Jie Liu