PharmaSUG 2026 - Paper DV-298

Dynamic Patient Profile Plot Development with SAS® Graph Template Language

Raghava Pamulapati, Dan Deng Fu, and Guowei Wu, Merck & Co., Inc., Rahway, NJ, USA

ABSTRACT

Patient profile reporting typically involves collecting all relevant information for a given patient and presenting it as a listing, which is often informative and straightforward. However, patient profile plots take this a step further by visually displaying a patient's data across multiple domains. These plots are not only more visually appealing but also provide a more powerful representation compared to listings.

We propose here a solution that involves developing a SAS® macro to generate patient profile plots that aligns data from multiple domains on a common study-duration axis, enabling efficient review and identification of potential safety signals for a given patient. The profile consolidates treatment exposure, key laboratory parameters with reference ranges, AEs with timing and severity, and optional concomitant medications, providing a clear visual context for evaluating potential safety signals such as drug-induced liver injury (DILI).

The solution addresses core challenges in automating patient profile generation: harmonizing heterogeneous clinical domains with differing structures and timing convention while balancing readability with variable data density across subjects. The output patient profile also aligns with FDA Safety Tables and Figures (SSTF) guidelines.

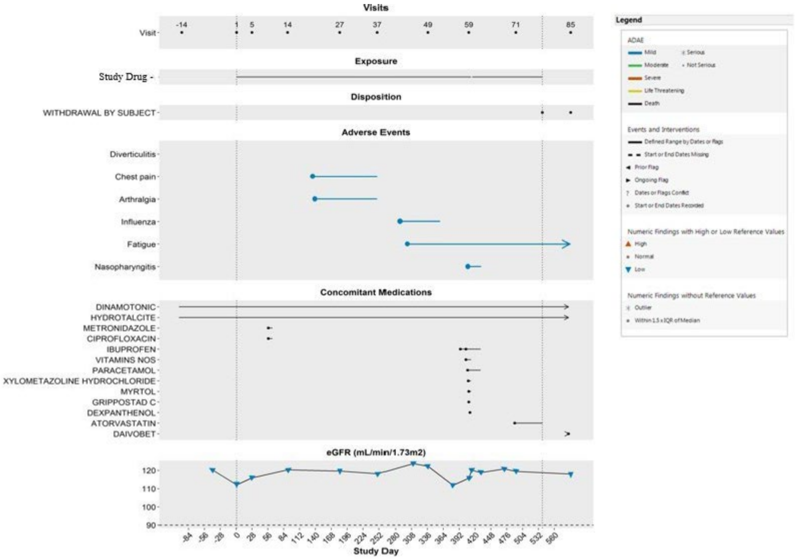
Our SAS® Graphics Template Language (GTL) based macro incorporates data from relevant ADaM datasets such as ADSL, ADEX, ADLB, ADAE, ADCM and uses automated algorithms to allocate panel spacing to adapt layout to individual subjects. Key innovations include weight-based space allocation, conditional template generation, and intelligent pagination producing consistent visualizations that streamline patient-level review and improve detection of potential safety signals.

INTRODUCTION

The U.S. Food and Drug Administration (FDA) published the Standard Safety Tables and Figures Integrated Guide, initially released in August 2022 and later updated in April 2025 to provide a standardized framework for presenting safety data in regulatory submissions. Developed in collaboration with the Targeted Analysis Guide (TAG) initiative, this guidance establishes consistent formats for safety visualizations across the pharmaceutical industry. The guide includes standardized templates for common safety displays, including Kaplan-Meier plots, forest plots, and individual patient profiles, enabling more efficient safety reviews and cross-study comparisons. The FDA TAG mockup shown in Figure 1 provides the standardized reference layout that guided the development of our graphical patient profile.

A patient profile is a comprehensive, patient level summary that integrates diverse clinical data into a single view. Traditionally, patient profiles include demographics, medical history, concomitant medications, adverse events, and laboratory results. They serve as a valuable tool for clinicians, researchers, and regulators to understand the trajectory of an individual's health during a trial. By consolidating information across multiple domains, patient profiles make it easier to identify potential safety signals, assess causality, and support decision making.

When patient profiles are combined with graphical representation of key laboratory parameters, they become even more powerful. Graphical patient profiles plot key lab parameters such as the ones related to DILI over time, annotated with baseline, onset, peak, and recovery. These visual cues allow reviewers to quickly grasp the progression of liver injury, identify critical thresholds (e.g., ALT > 3× ULN), and correlate changes in lab results with clinical events such as Adverse Events and concomitant medication use.



Source: [include Applicant source, datasets and/or software tools used].
Abbreviations: ADAE, adverse events analysis dataset; eGFR, estimated glomerular filtration rate.

Figure 1. TAG Mockup for Patient Profile

Although visually appealing and highly useful, such graphical patient profiles are programmatically challenging to produce. The complexity arises from incorporating data from multiple domains and generating individualized plots for subjects who meet the subset criteria. Developing a SAS® macro that can adapt to different scenarios for each patient is not an easy task. This paper demonstrates how we achieved this using SAS® Graph Template Language (GTL), the challenges we encountered, and our future plans to extend this to additional scenarios.

METHODOLOGY

Developing multi-panel plots in SAS can be achieved using PROC SGPANEL or the Graph Template Language (GTL). PROC SGPANEL is well suited and more cost-effective for standard multi-panel figures based on categorical groupings with uniform plot structures. However, it is fundamentally optimized for classification-based paneling and lacks the flexibility required for heterogeneous, cross-domain patient profile integration.

In contrast, GTL offers the architectural flexibility, independent panel control, and dynamic adaptability needed to build FDA-compliant patient profiles that integrate diverse clinical data types with varying structures and visualization requirements.

The comparison from Table 1 highlights the key differences between PROC SGPANEL and GTL in the context of building multi-panel patient profile plots.

Capability	**PROC SGPANEL**	**GTL**
Multi-panel support	Yes (automatic)	Yes (fully customizable)
Different plot types in each panel	Yes, but requires workarounds (all statements apply to all panels)	Yes, cleanly and intentionally
Layout flexibility	Lattice, uniform panels	Any layout (nested, asymmetric, custom grids)
Best use cases	Standard multi-panel comparison	Publication-quality custom layouts

Table 1. Comparison of PROC SGPANEL and GTL

Because GTL provides independent control over each graph cell, it is far better suited for advanced visualization needs such as combining different plot types, creating customized panel layouts, and independently configuring each component graph. For these reasons, GTL was selected as the framework for developing this plot.

IMPLEMENTATION

We developed a comprehensive SAS® macro built using the Graph Template Language (GTL) to produce integrated graphical patient profiles by consolidating data from multiple clinical domains into a single, time-aligned display. The macro includes extensive parameterization, enabling flexibility across different study designs while preserving a consistent and standardized output format. Macro consists of several key components:

1. Data Integration Layer: Extracts and harmonizes data from multiple ADaM datasets (ADSL, ADEX, ADLB, ADAE, ADCM)
2. Dynamic Space Allocation Engine: Calculates optimal panel heights based on data volume
3. Intelligent Pagination System: Distributes panels across pages when space constraints are exceeded
4. GTL Template Generation: Creates customized graph templates for each subject
5. Output Management: Generates publication-quality RTF or PDF files with appropriate resolution

Each of these components are discussed in detail below.

1. DATA INTEGRATION LAYER

The Data Integration Layer serves as the foundation of the macro, responsible for extracting, transforming, and harmonizing data from multiple ADaM datasets into a unified structure suitable for visualization.

1.1 Population Extraction

The macro begins by extracting the target population from the subject-level dataset (ADSL) based on user-specified criteria. This population determines which subjects proceed to subsequent data processing and plot generation.

Input: ADSL Dataset – Process Flow

- Apply user-defined WHERE clause
- Calculate treatment duration (end date – start date + 1)
- Extract demographic variables for title generation
- Generate list of subjects for iterative processing to produce subject-level plots

1.2 Panel-Specific Data Processing

Each visualization panel requires specialized data extraction and transformation: Across all panels, the macro performs four common operations:

- Extract relevant records based on user-specified criteria
- Calculate study-day values to align all data on a unified X-axis.
- Standardize common variables to ensure harmonization across domains
- Prepare panel-specific variables used in plotting

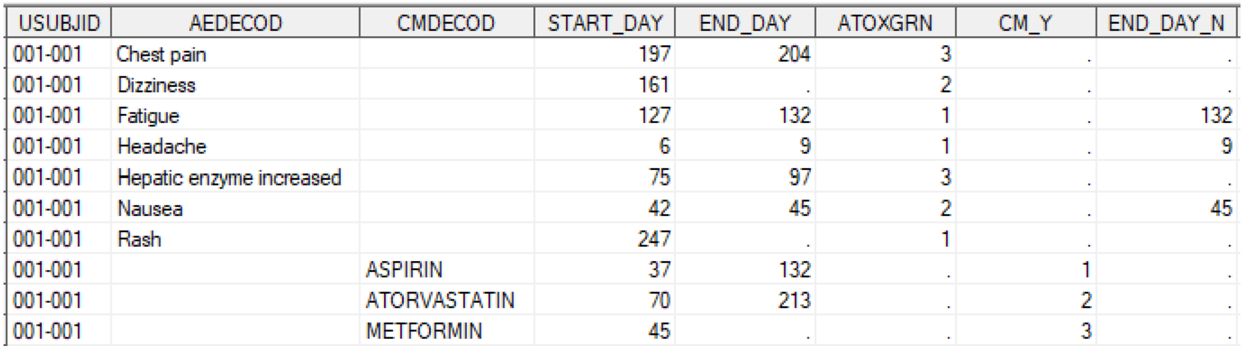
Beyond these shared steps, each panel applies additional logic tailored to its data domain:

- Panel 01: Exposure (ADEX) → Supports single-dose markers to display treatment intervals
- Panel 02: Laboratory (ADLB) → Captures reference ranges for contextual interpretation
- Panel 03: Adverse Events (ADAE) → Represents seriousness, severity, and ongoing events
- Panel 04: Concomitant Medications (ADCM) → Uses drug abbreviations and indicates ongoing medication flags

1.3 Data Harmonization

Table 2 presents the unified dataset created by harmonizing records across all contributing ADaM domains. Each source dataset ADEX, ADLB, ADAE, and ADCM contains domain-specific structures and timing conventions, which must be standardized to create a single patient-level timeline.

The consolidated dataset shown here illustrates how key variables such as study day, start and end dates, domain identifiers, and panel-specific attributes are aligned into a consistent structure.



USUBJID	AEDECOD	CMDECOD	START_DAY	END_DAY	ATOXGRN	CM_Y	END_DAY_N
001-001	Chest pain		197	204	3	.	.
001-001	Dizziness		161	.	2	.	.
001-001	Fatigue		127	132	1	.	132
001-001	Headache		6	9	1	.	9
001-001	Hepatic enzyme increased		75	97	3	.	.
001-001	Nausea		42	45	2	.	45
001-001	Rash		247	.	1	.	.
001-001		ASPIRIN	37	132	.	1	.
001-001		ATORVASTATIN	70	213	.	2	.
001-001		METFORMIN	45	.	.	3	.

Table 2. Consolidated Output Dataset from Multi-Domain Sources

2. DYNAMIC SPACE ALLOCATION ENGINE

As shown in Table 3, the dynamic space allocation engine calculates optimal panel heights based on actual data content, ensuring efficient use of available page space while maintaining readability.

2.1 Weight Calculation Algorithm

Each panel is assigned a "weight" representing its required vertical space. The weight is calculated dynamically based on data volume:

Panel Weight Formula: Weight = Base Weight + (# of Data Rows × Row Weight Factor)

Where:

Base Weight = 0.10 (minimum panel allocation)

Row Weight Factor = 0.05 (additional space per data row)

Panel	#of Data Rows	Calculated Weight
Exposure	1	0.10 + (1 × 0.05) = 0.15
Laboratory	5	0.10 + (5 × 0.05) = 0.35
Adverse Events	7	0.10 + (7 × 0.05) = 0.45
Concomitant Meds	4	0.10 + (4 × 0.05) = 0.30
Legend	-	0.09 (fixed)
TOTAL		1.34

Table 3. Example Calculations

2.2 Y-Axis Optimization

For panels with variable content (Adverse Events, Concomitant Medications), the engine optimizes Y-axis parameters to minimize wasted space:

Y-Axis Optimization Parameters

- viewmin: Dynamically calculated Y-axis minimum
- viewmax: Maximum value based on row count
- offsetmin: Reduced top padding for dense panels
- offsetmax: Reduced bottom padding for dense panels

Optimization Logic

- If data_rows ≤ 5 → *offset = 0.05* (standard padding)
- If data_rows ≤ 10 → *offset = 0.02* (reduced padding)
- If data_rows > 10 → *offset = 0.00* (no padding)

2.3 Row Weight Normalization

Before rendering, panel weights are normalized to ensure they sum to 1.0 (100% of available space):

Normalization Process

- Step 1: Compute total_weight as the sum of all panel weights
- Step 2: For each panel, calculate as follows:
  normalized_weight = panel_weight / total_weight
- Step 3: Pass the normalized weights to the GTL rowweights parameter.

3. INTELLIGENT PAGINATION SYSTEM

The Intelligent Pagination System automatically distributes panels across multiple pages when content exceeds single-page capacity, ensuring all data is displayed without truncation or overcrowding.

3.1 Page Threshold Determination

The system uses a threshold-based approach to determine when multi-page layout is required:

Pagination Decision Logic

- Page Threshold: 1.10 (110% of normalized space)
- Rule:
    - If total_weight > threshold, use a *multi-page* layout
    - Otherwise, a *single-page* layout is sufficient
- **Rationale:**
  The 110% buffer allows for slight variation in content density while preventing excessive visual compression.

3.2 Cumulative Weight Distribution Algorithm

When multi-page layout is triggered, panels are distributed using a cumulative weight algorithm:

Distribution Algorithm

- **Step 1:** Initialize cumulative_weight = 0.09 (legend weight).
- **Step 2:** Loop through panels in order (1 → 4):
    - Compute test_weight = cumulative_weight + panel_weight
    - If test_weight ≤ page_threshold → assign to **Page 1** and update weight
    - Otherwise → assign to **Page 2**
- **Step 3:** Add the legend to any page that includes Lab or AE or ConMed panels.
- **Step 4:** Assign the X-axis label to the final panel on each page.

3.3 Page-Specific Adjustments

The system automatically adjusts page elements based on panel distribution:

- Title continuity: Page 1 includes the suffix “(Continued)” to indicate a multi-page profile.
- X-axis efficiency: Only the bottom panel on each page displays the X-axis.
- Legend placement: Legends are shown only on pages containing panels that require them.
- Footnote logic: Panel-specific footnotes (AE, CM) appear only on pages where those panels are present.
- End notes and data source citations appear only on the final page to avoid redundancy.

4. GTL TEMPLATE GENERATION

The GTL (Graph Template Language) Template Generation component creates customized PROC TEMPLATE definitions for each subject, enabling precise control over visualization elements.

4.1 Template Structure

Figure 2 provides a simplified view of the GTL template structure, showing how the patient profile is built from a top-level STATGRAPH definition down through the lattice layout and individual panel overlays. This hierarchy highlights how GTL organizes global settings and panel-specific elements to create a coordinated multi-panel display.

```
STATGRAPH participant_profile
├── BEGINGRAPH
│   ├── Graph-level attributes (border, dimensions, padding)
│   │
│   └── LAYOUT LATTICE (rows=n, rowweights=(...))
│       │
│       ├── Panel 1: LAYOUT OVERLAY (Exposure)
│       │   ├── Y-axis options
│       │   ├── X-axis options (conditional)
│       │   ├── SCATTERPLOT
│       │   └── Reference lines
│       │
│       ├── Panel 2: LAYOUT OVERLAY (Laboratory)
│       │   ├── Y-axis with grid
│       │   ├── SERIESPLOT with line patterns
│       │   ├── Reference lines (thresholds)
│       │   └── Legend reference (name=)
│       │
│       ├── Panel 3: LAYOUT OVERLAY (Adverse Events)
│       │   ├── Reversed Y-axis with tick labels
│       │   ├── HIGHLOWPLOT (duration bars)
│       │   ├── SCATTERPLOT (start/end markers)
│       │   └── Serious/Non-serious differentiation
│       │
│       ├── Panel 4: LAYOUT OVERLAY (Concomitant Meds)
│       │   ├── Reversed Y-axis with drug labels
│       │   ├── HIGHLOWPLOT (duration bars)
│       │   └── SCATTERPLOT (start/end markers)
│       │
│       ├── SIDEBAR (Legend)
│       │   └── DISCRETELEGEND
│       │
│       └── SIDEBAR (Footnotes)
│           └── ENTRY statements
│
└── ENDGRAPH
```

Figure 2. GTL Template Hierarchy

4.2 Conditional Panel Inclusion

The Program 01 code block demonstrates how the macro conditionally includes each panel based on whether data are available for that subject. By checking the panel-specific record counts (e.g., &n_pn1, &n_pn2), the macro ensures that only panels with meaningful content are rendered.

```
 %if &n_pn1 GE 1 %then %do;
    /* Panel 1 template code */
%end;
%if &n_pn2 GE 1 %then %do;
    /* Panel 2 template code */
%end;
/* ... and so on for each panel */
```

Program 01. Conditional inclusion of the panel

4.3 Visual Encoding Standards

The template implements consistent visual encoding to represent different data elements, helping reviewers easily correlate multiple data points. Table 4 summarizes the symbols used for each data type.

Element	Description	Symbol
Treatment period	Dotted vertical reference lines	⠿
Lab parameters	Different line patterns per parameter	— ━ ···
Lab thresholds	Horizontal dotted reference lines	···
AE duration	Horizontal bars (HIGHLOWPLOT)	—
AE start (non-serious)	Open right-pointing triangle	◁
AE end (non-serious)	Open left-pointing triangle	▷
AE start (serious)	Filled right-pointing triangle	◀
AE end (serious)	Filled left-pointing triangle	▶
CM duration	Horizontal bars (HIGHLOWPLOT)	—
CM start	Open right-pointing triangle	◁
CM end	Open left-pointing triangle	▷

Table 4. Visual Encoding Standards

4.4 Adaptive X-Axis Display Strategy

Challenge: Maintain synchronized timeline across panels while showing axis label only once to maximize vertical space.

Solution: Conditional X-axis display based on panel position

Key Innovation: All panels share identical X-axis range for temporal alignment; however, as implemented in Program 02, only the bottom panel displays labels, creating a clean, uncluttered appearance while maximizing data display area.

```
%if &last_panel_page1 = 3 %then %do;
  xaxisopts=(label="Study Duration (Days)"
       linearopts=(viewmin=&x_axis_start   /* Day 1 or baseline */
            viewmax=&endmaxdy)       /* Treatment + follow-up */
       display=(line tickvalues label))    /* Full display */
%end;
%else %do;
  xaxisopts=(display=none              /* Hidden on intermediate panels */
       linearopts=(viewmin=&x_axis_start
            viewmax=&endmaxdy))      /* Same range for alignment */
%end;
```

Program 02. Conditional X-axis

4.5 Reusable Reference Line Macro

Challenge: Apply consistent temporal markers (treatment start/end) across all panels without code duplication.

Solution: Macro-based reference line generation.

Key Innovation: Program 3 shows the reference line macro ensuring consistent reference line styling and positioning across all panels, with subject-specific values calculated once and reused.

```
%macro add_reference_lines;
  /* Treatment start (subject-specific) */
  referenceline x=&trt_start /
    lineattrs=(color=black pattern=dot thickness=2);
  /* Treatment end (subject-specific) */
  referenceline x=&endtrtdy /
    lineattrs=(color=black pattern=dot thickness=2);
%mend;
/* Applied consistently to all panels */
layout overlay / ...;
  /* Plot elements */
  %add_reference_lines;
endlayout;
```

Program 3. Reference line helper macro

5. OUTPUT MANAGEMENT

The Output Management component handles the generation of publication-quality output files with appropriate formatting, resolution, and metadata.

5.1 Output Format Support

The macro supports multiple output formats. Table 5 summarizes the output formats supported by the macro and highlights their key characteristics. It provides a quick comparison between RTF and PDF, outlining differences in editability, resolution, and rendering style to help users select the most appropriate format for their reporting needs.

Format	**Characteristics**
RTF	**• Editable in Microsoft Word** **• Default resolution: 200 DPI** **• Embedded PNG images**
PDF	**• Non-editable, fixed layout** **• Default resolution: 300 DPI** **• Vector/raster hybrid**

Table 5. Supported Output Formats

5.2 Graphics Configuration

The Program 02 sets the global graphics options used for producing the patient profile, including image format, size, and resolution. These settings ensure that all plots are generated with consistent quality and meet the required output specifications.

```
ods graphics on /
    reset=index
    border=off
    outputfmt=&graphic_fmt
    width=&plot_width.in
    height=&plot_height.in
    imagefmt=png
    dpi=&resolution_dpi;
```

Program 04. ODS GRAPHICS statement

5.3 Multi-Subject Output Handling

The macro generates a single output file containing all subjects. Depending on the their data volume, each subject may have a one- or two-page profile. Figure 04 illustrates the ODS process flow used to produce the output for each subject

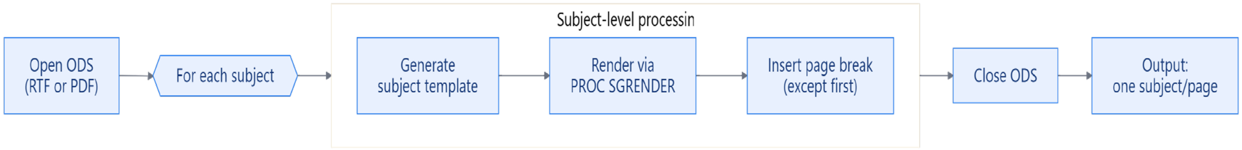


Figure 4. Multi-Subject Output Flow

5.4 Title and Footnote Management

Dynamic title and footnote generation ensures consistent formatting:

Title Structure:

Line 1: Main title (e.g., "Graphical Participant Profile for DILI")

Line 2: Subject demographics (PID, Treatment, Age, Sex)

Line 3: Subtitle (optional, user-defined)

Footnote Structure:

- AE severity scale legend (within graph)
- Concomitant medication mapping (within graph)
- End notes (below graph, user-defined)
- Data source citation (below graph, optional)

Summary

The macro architecture integrates these six components into a cohesive system that

- Extracts and harmonizes data from multiple clinical domains
- Calculates optimal space allocation based on actual data content
- Intelligently distributes panels across pages when needed
- Validates data constraints before rendering to prevent errors
- Generates customized templates for each subject's unique data profile
- Produces publication-quality output suitable for regulatory submissions

This modular design enables flexibility in accommodating different study designs while maintaining consistent, high-quality visualizations for safety assessment and individual case narratives. The figure 5 provides a clear visual summary of how data moves through each processing layer before being rendered into the final multi-panel plot.

Figure 5. Component Integration Flow

FINAL OUTPUT:

The example plot shown in Figure 6 represents a subject with 7 adverse events and 3 concomitant medications. Based on the data, the macro dynamically places the AE and ConMed panels on the first page, as these three panels fit together. The Lab panel, which has a wider Y-axis range, is large enough to warrant its own page and is therefore allocated separately.

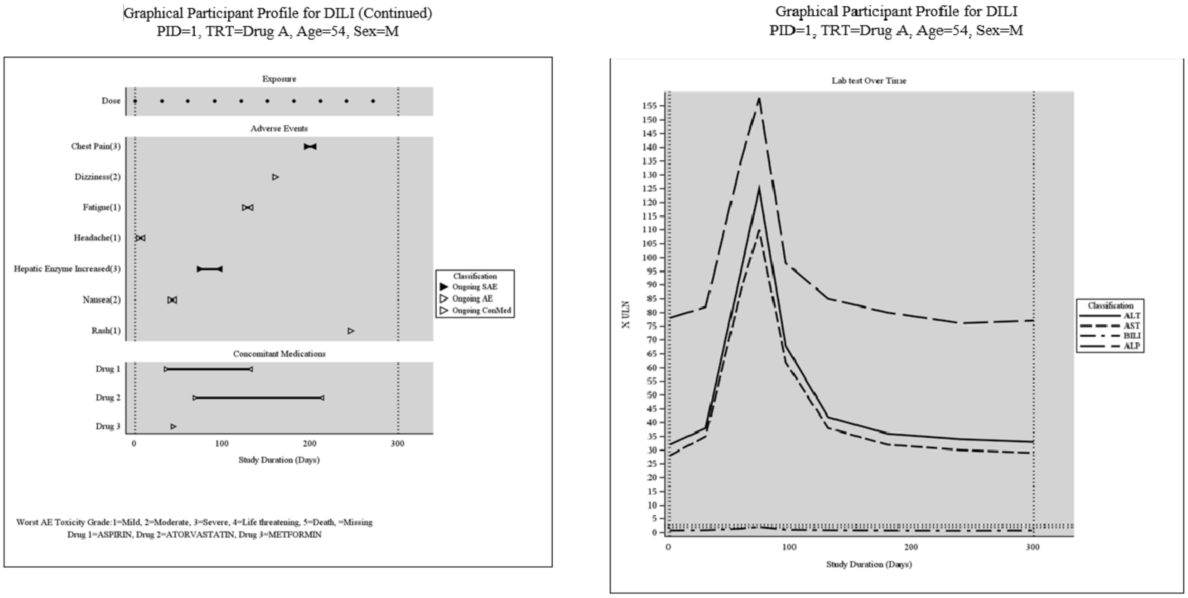


Figure 6. Final output example

LIMITATION

This approach does not yet support automatic pagination capable of handling an unlimited number of records—an important requirement for multi-panel Adverse Events (AE) and Concomitant Medications (ConMed) displays. In its current form, the workflow relies on manual control of panel rendering, legend placement, and page breaking, all of which depend on the number of records associated with each panel. As datasets increase in size, this limitation becomes more pronounced, as panels may exceed available page space without dynamic pagination.

In most clinical scenarios, not all adverse events and concomitant medications are equally relevant to the primary safety assessment, particularly when evaluating laboratory abnormalities. It is strongly recommended to work closely with the clinical team to identify and filter for the specific adverse events and concomitant medications that are clinically relevant.

Although automated pagination has not yet been implemented, we fully recognize its importance for production-quality outputs. This capability is planned for a future enhancement. Once introduced, it will eliminate the current constraints and enable better scalability for studies with large or highly variable record counts.

Additional notes:

- ConMed medication labels are annotated in the section below the graph.
- Panel order is predefined and cannot be modified.
- A maximum of 10 combined records from AE and ConMed may be displayed, or up to 10 AE records if no ConMed panel is present
- The total number of panels is limited to four.

FUTURE WORK

Looking ahead, we plan to implement advanced pagination controls to ensure optimal page layout, prevent unintended breaks, enabling the plot to dynamically accommodate subjects with larger numbers of AE and ConMed records without imposing record limits. In addition, we will explore ways to extend this solution beyond SAS® by leveraging R for patient profile plot development. R offers advanced visualization capabilities through packages such as ggplot2 and plotly, which could enable interactive and highly customizable patient profiles.

CONCLUSION

This paper presents a comprehensive solution for generating integrated graphical patient profiles that address a critical need in clinical drug development, particularly for Drug-Induced Liver Injury (DILI) assessment. By consolidating multi-domain patient data including treatment exposure, laboratory parameters, adverse events, and concomitant medications into a single, time-aligned visualization, we have created a powerful tool that enhances clinical insight and regulatory review efficiency.

ACKNOWLEDGMENT

We would like to acknowledge the support and guidance of our management team, who reviewed this article and provided valuable contributions:

- Amy Gillespie
- Suhas Sanjee Ramesh
- Susan Kramlik

REFERENCE

Patient Profile Mockups

Standard Safety Tables and Figures: Integrated Guide V2.0 April 2025
https://www.fda.gov/media/187065/download

SAS GTL Guide
https://support.sas.com/documentation/cdl/en/grstatgraph/63878/HTML/default/viewer.htm#p0891gx3y0z8xqn1k9ijhv5xughi.htm