Paper SM06

GraphStudio: Simplifying the Journey from Clinical Data to Submission-Ready Visuals

Poonam Deshmukh, Johnson & Johnson, Mumbai, India

ABSTRACT

Clinical graphs are indispensable to understanding trial data, yet they remain one of the least standardized and most manually validated components of regulatory deliverables. Unlike tables and listings, graphical outputs often rely on visual estimation for quality control, introducing inefficiencies, subjectivity, and traceability risks.

This paper introduces **GraphStudio**, an automation-first visualization framework designed to address not only plot generation but also the long-standing gaps in graph QC, traceability, and stakeholder usability. Built using R Shiny, it supports multiple plot types including Mean (±SE) line plots, Box Plots, Grouped Bar Plots, and Kaplan–Meier plots across Laboratory, Vital Signs, Exposure, and Pharmacokinetic domains, integrating visualization, numeric transparency, and submission-ready outputs into a single interface. Beyond standardized plot creation, GraphStudio outputs a structured CSV containing the exact numeric values underlying each plot, establishing a rigorous and reproducible QC mechanism for graphs, comparable to that of tables and listings.

Beyond submission workflows, the framework functions as an early-phase exploratory and decision-support tool, allowing technical and non-technical stakeholders to collaboratively evaluate visualization needs before finalizing submission outputs. This paper demonstrates how GraphStudio transforms clinical graphs from visually interpreted artifacts into governed, auditable, and decision-ready deliverables.

INTRODUCTION

In clinical reporting, graphs carry significant weight. They are often the first artifacts reviewed by clinicians, statisticians, and other decision makers, shaping interpretation long before detailed tables are examined. Despite this importance, the processes used to generate and validate graphs have not evolved at the same pace as those for tables and listings. Visualization workflows are often script-heavy, manually adjusted, and validated through visual inspection rather than structured numeric verification.

Statistical programmers today operate in an environment defined by increasing data complexity and compressed timelines. Automation has become essential, yet its application remains uneven. While tables and listings benefit from mature frameworks such as standard macros and structured QC, development and validation of graphs often remain fragmented. Custom code, ad hoc revisions, and visual estimation using axis scales - an approach that is time-consuming and inherently imprecise.

This inconsistency is not merely inefficient; it introduces operational risk. As organizations scale across studies, therapeutic areas, and technologies, the absence of a standardized, auditable approach to graph generation and QC becomes increasingly difficult to justify.

GraphStudio was developed to address this gap, not as another plotting tool, but as a framework that treats graphs with the same rigor, governance, and complete with numeric transparency, reproducibility, and cross-functional usability.

THE HISTORICAL GAP IN GRAPH QC

Unlike tables and listings, graphical outputs traditionally do not have a standardized QC dataset. QC programmers often reconstruct derivations, apply assumptions around filtering and grouping, and visually approximate values from axes, error bars, or box boundaries. When discrepancies arise, identifying whether the issue lies in derivation logic, plotting code, or interpretation can be time-consuming and ambiguous.

Minor differences such as in visit ordering, population definitions, or summary statistic calculations can lead to prolonged review cycles, not because the graph is incorrect, but because it is difficult to prove correctness. As studies grow and timelines tighten, the reliance on visual estimation often based on axis interpretation becomes both inefficient and risky. More importantly, it creates a fundamental asymmetry: graphs, despite their importance, are validated with less precision than other deliverables.

From an organizational perspective, this represents a structural inefficiency. Graph QC effort scales linearly with output volume, while confidence in results does not. Moreover, the lack of standardized QC artifacts limits traceability and makes onboarding new team members more challenging.

GraphStudio directly challenges this paradigm by embedding **numeric transparency as a core requirement** into the graph generation process itself, ensuring that every graph is accompanied by verifiable data.

DESIGN PRINCIPLES AND OBJECTIVES

GraphStudio was intentionally designed to close this QC gap while simultaneously improving efficiency and usability. The framework is guided by four principles:

- **Automation over repetition:** Analytical variation should be driven by inputs, not code changes.
- **Numeric transparency by design:** Every plotted value must be explicitly accessible.
- **QC equivalence:** Graphs should be QC'ed with the same rigor as tables and listings.
- **Stakeholder inclusivity:** Visualization exploration should not be limited to technical users.

Rather than focusing solely on visualization aesthetics, the framework emphasizes governance, traceability, and automation.

ARCHITECTURE OVERVIEW

GraphStudio operates as a parameter-driven orchestration layer between ADaM datasets and submission-ready outputs.

Rather than requiring manual dataset selection, GraphStudio loads the appropriate analysis data based on the selected domain tab. Users configure graphs through controlled inputs rather than code changes. Plot type, parameter, treatment group, visit, and analysis value (e.g., AVAL or CHG), ensuring consistency while maintaining flexibility. A dedicated **Graph ID** input acts as the central traceability key, driving output naming and ensuring correct retrieval of titles and footnotes from an internal specification file, enabling alignment with submission standards.

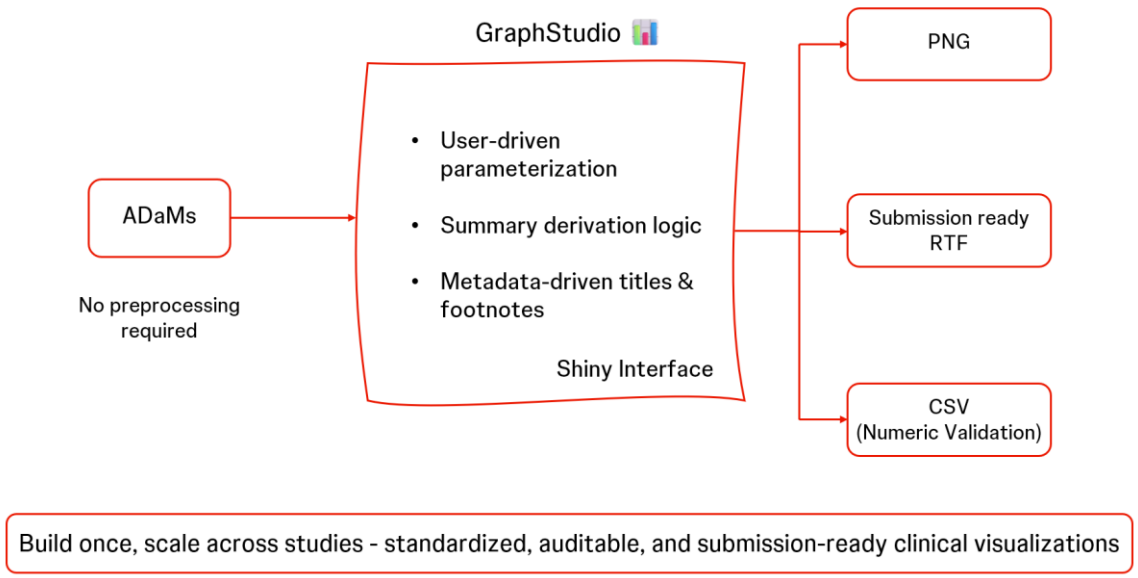


Figure 1: High-level architecture and workflow

At a high level, analysis-ready datasets serve as inputs, while the application layer applies standardized derivation logic based on user-selected parameters. Outputs are generated simultaneously in multiple formats, each serving a distinct purpose within the review and submission process.

This separation of concerns allows organizations to extend incrementally adding new plot types or domains without redesigning the core framework.

MULTI-DOMAIN AND MULTI-PLOT SUPPORT

Currently supports Mean (±SE) line plots, Box Plots, Grouped Bar Plots, and Kaplan–Meier time-to-event plots across Laboratory, Vital Signs, Exposure, and Pharmacokinetic domains. Each plot type follows a standardized processing pipeline, while accommodating domain-specific summary requirements.

For example, Box Plots dynamically compute and display distributional statistics, while Kaplan–Meier plots derive survival probabilities and event counts. Despite these differences, all plots ultimately produce a consistent set of outputs, reinforcing standardization across domains.

FROM VISUAL ESTIMATION TO NUMERIC VALIDATION: REDEFINING GRAPH QC

One of most impactful contributions is the introduction of a graph specific QC dataset in the form of a CSV file. This file contains exactly the values used to generate the visual output, including treatment group, visit, and relevant summary statistics such as mean, standard error, minimum, maximum, quartiles, or event counts, depending on the plot type.

Traditional Approach vs GraphStudio Framework

Dimension	Traditional	GraphStudio
QC Basis	Visual inspection using axes & gridlines	Direct comparison against exported csv
Reproducibility	Logic often reimplemented, increasing variability	Single derivation logic derives plot & QC output
Effort Required	High manual effort & repeated interpretation	Streamlined, programmatic validation
Scalability	Poor	High; supports reuse across studies & TA's

Figure 2: Comparison of traditional graph QC versus GraphStudio enabled QC

This fundamentally changes how graph QC is performed. Rather than estimating values visually from axes, QC programmers can independently derive expected results and compare them directly to the CSV. Validation becomes precise, reproducible, and efficient.

This approach eliminates ambiguity, reduces QC effort, and significantly improves confidence in graphical outputs. Importantly, the CSV output is named using the same Graph ID as the plot, ensuring clear traceability across all artifacts.

BEYOND SUBMISSION: EARLY-PHASE COLLABORATION

This application is not limited to final reporting stages. It can also be used during early study phases to explore potential visualizations, compare parameter behavior across visits, and evaluate which outputs are most meaningful for interpretation.

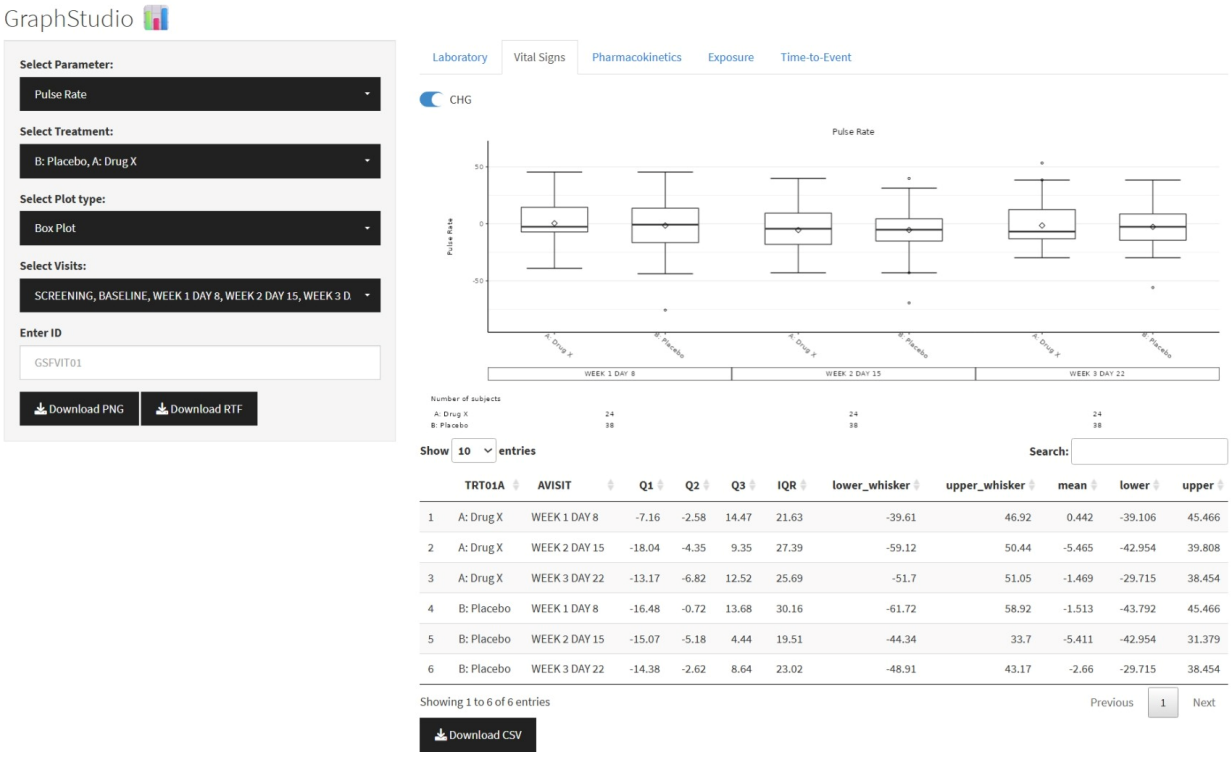
Because the interface is intuitive and parameter-driven, non-technical stakeholders can participate directly in these discussions. This shifts visualization decisions upstream, reducing late-stage rework and aligning expectations earlier in the study lifecycle.

This end-to-end alignment from exploration to QC to final delivery demonstrates how automation can enforce standards without reducing flexibility.

EMBEDDED NUMERIC CONTEXT FOR REVIEW AND DECISION-MAKING

A fully synchronized numeric table is displayed directly beneath each graph within the application reflecting the same values used in plotting and CSV generation. This table is designed explicitly for **statisticians and other decision-making teams like clinicians**, enabling them to interpret results with clarity and confidence during exploratory review and output finalization discussions. By having direct access to means, variability measures, or distributional statistics alongside the graph, teams can immediately contextualize visual trends without relying on approximations or follow-up analyses.

Importantly, this embedded table is not intended for QC purposes. QC programmers operate independently and rely exclusively on the exported CSV files for validation, maintaining appropriate separation between production and QC activities. The table beneath the graph instead serves as a transparency and communication layer, accelerating exploratory review, supporting informed decision-making, and reducing iteration cycles between stakeholders.



FUTURE EXTENSIONS

GraphStudio is designed to evolve. Potential extensions include batch execution, integrated graph library, and built-in visual checks. As regulatory expectations and data complexity continue to increase, such frameworks will become essential rather than optional.

CONCLUSION

GraphStudio represents a shift in how clinical graphs are produced and validated. By embedding automation, numeric transparency, submission readiness, and QC rigor into a single framework, it eliminates long-standing inefficiencies in graph QC. More importantly, it elevates graphs from static deliverables to active decision-support tools, enabling statisticians and stakeholders to engage with data more effectively. Through its scalable design and low barrier to adoption, GraphStudio demonstrates how scripting, automation, and thoughtful architecture can transform everyday statistical programming activities. As organizations continue to modernize clinical reporting, frameworks like GraphStudio offer a practical blueprint for achieving consistency, audit readiness, and collaboration without sacrificing flexibility or speed.

CONTACT INFORMATION

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

Author Name: Poonam Deshmukh
Company: Johnson & Johnson
Email: PDeshm20@ITS.JNJ.com
LinkedIn: https://in.linkedin.com/in/poonam-deshmukh