

# APAC 2026

**Cross-Platform Programming for Clinical Development : Practical Use of SAS, R, and Python**

Sandeep Lakkol and Vinay S  
Atorus Research



# Using SAS, R, and Python in Cross-Platform Programming for Clinical Development

Integrating Analytical Strengths for Regulatory and Research Excellence



## **Purpose of Cross-Platform Programming**

Combining SAS, R, and Python allows clinical programmers to optimize for regulatory rigor, analytical flexibility, and computational power across different stages of drug development.



## **Regulatory and Research Alignment**

Clinical data analysis must satisfy stringent FDA/EMA requirements while enabling exploratory analytics an integration achieved by leveraging multiple Programming languages.



## **Strategic Objective**

Enable reproducible, transparent, and efficient workflows that merge traditional SAS-based processes with innovative open-source methods.



# Clinical Development Programming Overview

## Data, Compliance, and Analytics Foundation



### Clinical Development Lifecycle

Encompasses phases from preclinical research through post-marketing surveillance, where data collection, cleaning, and analysis drive regulatory and scientific decisions.



### Programming's Central Role

Statistical programmers build validated datasets, generate TLFs (Tables, Listings, Figures), and ensure traceability across study artifacts for submission readiness.



### Compliance and Quality Control

Regulatory compliance (FDA, EMA, PMDA) mandates adherence to standards such as CDISC SDTM and ADaM, with audit trails ensuring data integrity and reproducibility.



# Role of SAS in Regulatory Submissions

Gold Standard for Compliance and Data Integrity



## **Regulatory Acceptance**

SAS is trusted by regulatory agencies such as the FDA and EMA for producing validated, auditable outputs aligned with CDISC SDTM and ADaM standards.



## **Validated Workflow and Traceability**

Its structured data step and PROC framework allow consistent dataset transformations, quality checks, and reproducible TLF generation for submissions.



## **Integration with Standards and Tools**

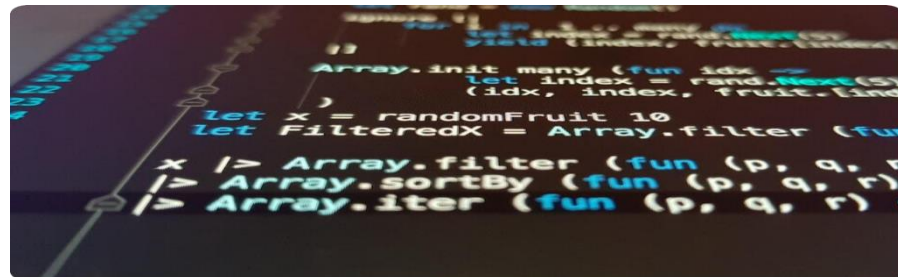
SAS integrates seamlessly with Pinnacle 21, Define.xml, and validation tools to ensure outputs meet compliance and submission-ready criteria.



# Practical Use of R in Clinical Research

## Driving Statistical Innovation and Visualization

- **Advanced Statistical Modeling:** R supports modern statistical techniques such as mixed models, survival analysis, and Bayesian inference, enhancing the analytical capabilities of clinical teams.
- **Visualization and Interactive Reporting:** Tools like ggplot2 and Shiny enable reproducible, dynamic data visualization and real-time dashboards for monitoring clinical trial progress.
- **Packages:** **admiral** (ADaM generation), **rtables/Tplyr** (regulatory tables), **haven** (SAS data import), **tidyverse** for data manipulation.



R's validation gap is being addressed through solutions like OpenVal, which provides a repository of 450+ validated packages with IQ/OQ documentation, enabling GxP-compliant R workflows for clinical programming.



## Practical Use of R in Clinical Research Contd.

### Driving Statistical Innovation and Visualization

| Initiative                             | Role                                                                 | Impact                                                            |
|----------------------------------------|----------------------------------------------------------------------|-------------------------------------------------------------------|
| <b>R Validation Hub</b>                | Frameworks & tools to assess R package quality in regulated contexts | Makes R adoption safer and more documented                        |
| <b>FDA Pilot Submissions (2021–23)</b> | Test cases showing that FDA can accept R-based submission packages   | Demonstrates real regulatory pathway validity                     |
| <b>Pharmaverse</b>                     | Shared, curated R tools for clinical reporting and submissions       | Reduces duplication, increases consistency, accelerates workflows |



# Practical Use of Python in Clinical Research

## Automation, Integration, and Statistical Programming



### **Automation and Workflow Efficiency**

Python automates data preparation, transformation, and reporting tasks, reducing manual effort while ensuring reproducibility in large-scale clinical studies.



### **Advanced Statistical Programming**

Through libraries like pandas, SciPy, pyreadstat (SAS/SPSS file reading), polars (high-performance dataframes), great-tables (clinical tables) and statsmodels, Python supports linear and mixed models, simulation-based analyses, and modern statistical methodologies aligned with regulatory expectations.



### **Integration and Reproducibility**

Python connects seamlessly with SAS and R, allowing hybrid pipelines for data validation, statistical analysis, and visualization within unified, auditable environments.



# Cross-Platform Integration and Hybrid Workflows

Unifying SAS, R, and Python for End-to-End Clinical Analytics



## Unified Interoperability

Modern analytics platforms such as SAS Viya, Posit Workbench, Jupyter Lab, Positron and open-source connectors allow R and Python code to execute directly within SAS environments, establishing a governed, multi-language ecosystem.



## Collaborative Efficiency

Statisticians can depend on SAS for validated deliverables, while data scientists employ R and Python for advanced modeling and visualization bridging regulated programming with data science innovation.



## Traceability and Compliance

Containerization, version control, and CI/CD pipelines maintain reproducibility, auditability, and GxP compliance across integrated workflows.



# Comparative Strengths & Limitations of SAS, R, and Python

## Choosing the Right Tool for Each Clinical Programming Task



### **SAS – Regulatory Reliability**

Industry gold standard for validated, auditable outputs. Ideal for SDTM/ADaM datasets, controlled environments, and FDA/EMA submissions.



### **R – Statistical Flexibility**

Excels in modeling, visualization, and exploratory analyses. Supports reproducibility and collaboration through open-source ecosystems (CRAN, Bioconductor, pharmaverse).



### **Python – Automation and Scalability**

Best suited for data pipelines, machine learning, and integration with modern infrastructures such as APIs and cloud systems.



# Case Studies: Cross-Platform Programming in Pharma

## Real-World Applications of SAS, R, and Python Integration



### Hybrid Workflow in Oncology Trials

A global pharma team used SAS for CDISC deliverables and R for survival analysis visualizations, reducing reporting cycle time by 30%.



### Python-Powered Automation

Python scripts automated data validation across multiple studies, integrating directly with SAS datasets through APIs, improving audit readiness.



### Machine Learning for Recruitment

Using Python and R, predictive models identified underrepresented patient groups, improving recruitment diversity and efficiency.



# Career & Skills Development in Clinical Programming

## Building Multi-Language Expertise for the Future



### **Expanding Skill Portfolios**

Modern programmers must master SAS for regulatory deliverables, R for statistical innovation, and Python for automation and integration.



### **Industry Demand for Cross-Platform Proficiency**

Pharma companies increasingly seek professionals who can bridge data science, programming, and regulatory requirements.



### **Continuous Learning and Professional Validation**

Ongoing education through credentials like SAS Certified Clinical Trials Programmer and advanced R/Python certifications enhances credibility and compliance readiness. Active participation in open-source and professional communities strengthens applied expertise and industry alignment.



## Conclusion & Future Outlook

### Converging Compliance, Innovation, and Automation in Clinical Programming



#### **Unified Analytical Ecosystems**

The convergence of SAS, R, and Python is creating cohesive ecosystems where validated analytics and data science coexist seamlessly.



#### **Evolving Regulatory Acceptance**

Regulatory agencies are increasingly open to hybrid workflows as traceability, reproducibility, and validation frameworks mature.



#### **Next-Generation Skillsets**

Future programmers will act as integrators - balancing statistical rigor, coding efficiency, and automated intelligence across platforms.



# Summary of Key Insights

## Leveraging SAS, R, and Python for Modern Clinical Programming



### **Unified Cross-Platform Strategy**

SAS ensures validated, regulatory-grade outputs; R delivers statistical innovation; and Python drives automation and integration, forming a synergistic analytics ecosystem.



### **Efficiency and Collaboration**

Hybrid workflows bridge statisticians, data scientists, and regulatory teams, fostering faster, compliant, and more transparent analytics.



### **Future-Ready Skill Development**

Professionals fluent across SAS, R, and Python are best positioned to lead in data-driven clinical environments emphasizing compliance and innovation.



## The Global Healthcare Data Science Community

### Contact Channels

UK +44 1843 609600  
office@phuse.global  
phuse.global

### Social Media

@phusetwitta  
/phusebook  
/phusetube  
/company/phuse