Integrating ViralFlow and MacWorP for Automated Viral Genomic Surveillance Workflows

Raphael Pinho[1], **Sintia Almeida**[1,2], **Roberto Simões**[2],
Cassia Kahwage[1,2], **Marcos Amaris**[1]

[1]Universidade Federal do Pará, Belém – PA – Brasil

[2]Instituto Evandro Chagas, Belém – PA – Brasil

`raphael.pinho@icen.ufpa.br`, `{sintiaalmeida, robertosimoes}@iec.gov.br`
`{cassiak, amaris}@ufpa.br`

Abstract. *We present an integrated web platform combining ViralFlow and MacWorP for automated viral genomic surveillance through distributed workflow orchestration and accessible web-based execution. A proof-of-concept deployment demonstrated the feasibility of integrating specialized viral genomics pipelines within a scalable workflow management environment. Preliminary results indicate gains in accessibility, reproducibility and distributed processing, supporting the potential of the platform as a modular framework for genomic surveillance in public health laboratories.*

1. Introduction

Genomic surveillance has become essential for monitoring emerging pathogens and supporting public health responses, but routine deployment of bioinformatics workflows remains limited by command-line complexity, software dependencies and computational infrastructure requirements, particularly in resource-constrained settings [Gardy and Loman 2018]. Existing solutions address this challenge only partially. ViralFlow provides an automated and reproducible viral genomics pipeline [da Silva et al. 2024], but still requires computational expertise, whereas MacWorP offers accessible workflow orchestration through a web-based environment, but lacks specialized viral genomic pipelines.

To address this gap, we present an integrated platform combining ViralFlow and MacWorP for automated viral genomic surveillance through distributed and containerized workflow execution. Unlike standalone pipelines or generic workflow managers, the proposed integration couples specialized genomics analysis with scalable workflow orchestration in a unified architecture oriented to public health applications. The main contributions of this work are:

(i) an integrated architecture coupling ViralFlow and MacWorP for automated viral genomic surveillance;
(ii) a distributed computational environment supporting scalable workflow execution;
(iii) a proof-of-concept implementation demonstrating integration feasibility; and
(iv) an open modular framework aimed at reducing technical barriers for genomic surveillance.

2. Concepts and Theoretical Background

Genomic surveillance complements traditional epidemiological surveillance by enabling lineage tracking, mutation detection and outbreak monitoring, but its routine application depends on computational infrastructures often difficult to deploy in public health settings [Gardy and Loman 2018]. ViralFlow is a Nextflow-based workflow for automated viral genomic surveillance, integrating quality control, genome assembly, variant calling and lineage assignment [da Silva et al. 2024]. Although reproducible and specialized for surveillance analyses, its use still depends on command-line expertise.

MacWorP is a web-based workflow management platform that supports distributed execution and reproducible orchestration through modular services. While it improves accessibility to complex workflows, it lacks native support for specialized viral genomic pipelines. Their integration provides the conceptual basis of the proposed platform by coupling domain-specific genomics analysis with accessible workflow automation.

3. Related Works

Several platforms address accessibility or scalability in bioinformatics workflows, but generally only partially support genomic surveillance automation. Galaxy provides graphical workflow execution but lacks specialization for viral surveillance [Afgan et al. 2018]. IRMA supports viral genome analysis with limited orchestration [Shepard et al. 2016], while Nextstrain focuses primarily on downstream phylogenetic analysis [Hadfield et al. 2018]. Workflow-oriented ecosystems such as nf-core and Nextflow provide scalable workflow management, but typically target technically experienced users. A comparative summary of representative platforms is presented in Table 1.

Table 1. Comparative overview of computational platforms for genomic surveillance and bioinformatics workflow execution.

Platform	Viral-specific	Web GUI	Distributed	Public health focus	References
Galaxy	Partial	Yes	Partial	No	[Afgan et al. 2018]
IRMA	Yes	No	No	Partial	[Shepard et al. 2016]
Nextflow ecosystem	Partial	Yes	Partial	Partial	[Di Tommaso et al. 2017]
Proposed platform	Yes	Yes	Yes	Yes	This work

As summarized in Table 1, the proposed platform differs by combining specialized viral genomic analysis, web-based execution, distributed orchestration and explicit orientation toward public health genomic surveillance.

4. Materials and Methods

4.1. System Architecture

The proposed platform was designed to reduce technical barriers associated with executing bioinformatics pipelines, enabling users without command-line expertise to perform genomic analyses through an accessible web environment. The system integrates ViralFlow and MacWorP within a distributed and containerized architecture aimed at scalable workflow orchestration for genomic surveillance applications (Figure 1).

The underlying software stack supporting this architecture is summarized in Table 2, highlighting the technologies associated with interface, orchestration, execution and storage layers.

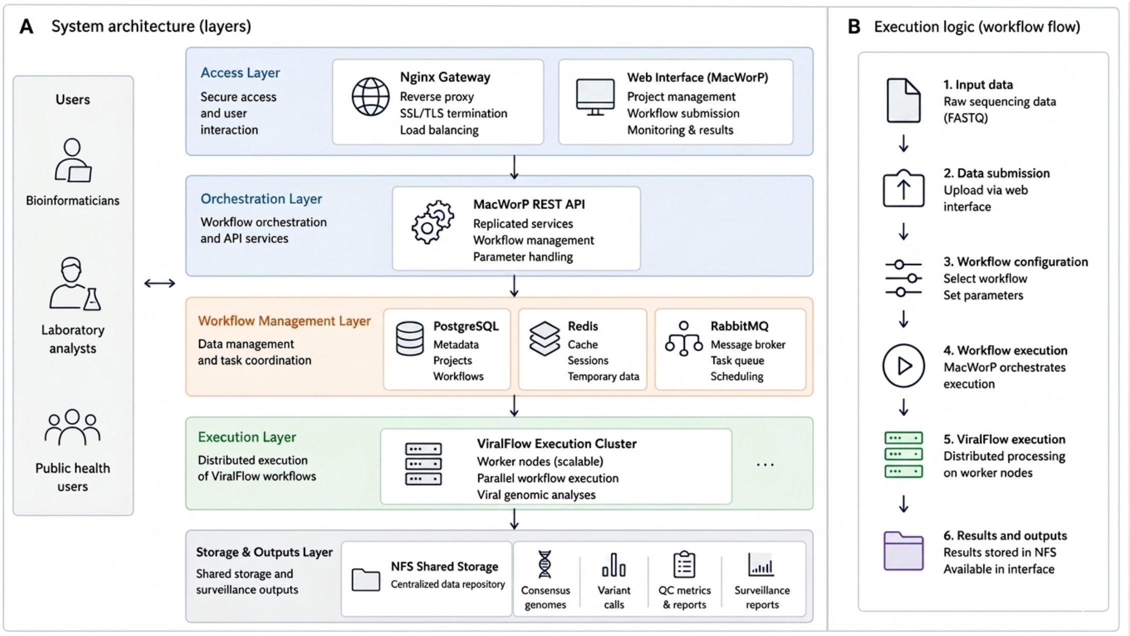

Figure 1. (A) Layered architecture coupling user access, workflow orchestration, distributed ViralFlow execution, and centralized storage. (B) End-to-end workflow from data submission to distributed processing and surveillance outputs.

As shown in Table 2, the integrated software stack combines complementary technologies for user interaction, distributed task coordination and scalable workflow execution, providing the computational foundation for modularity, reproducibility and portability of the proposed platform. The architecture adopts a modular multi-service design in which user requests are received through HTTPS and managed by Nginx acting as reverse proxy, SSL/TLS termination layer and load balancer. The frontend, implemented in Vue.js, provides graphical interfaces for project management, workflow parameterization and monitoring of analysis execution.

The backend is implemented as a Flask-based REST API deployed in replicated instances to support scalable request handling. System persistence and asynchronous task coordination rely on PostgreSQL, Redis and RabbitMQ, which support metadata management, caching and workflow scheduling. Bioinformatics analyses are executed through distributed worker nodes where ViralFlow is integrated as a managed workflow. Shared NFS storage provides centralized management of input data, workflow definitions and generated outputs, enabling efficient data exchange among computational services while supporting distributed execution. Together, these components establish a scalable computational environment coupling workflow orchestration and specialized viral genomics analysis in a unified platform.

4.2. Platform Integration and Workflow Implementation

The methodological core of this work was the integration of ViralFlow as an executable workflow within the MacWorP orchestration environment. In this architecture, ViralFlow performs genomic analysis tasks including read quality control, genome assembly, variant detection and lineage assignment, while MacWorP manages workflow execution, user interaction and distributed scheduling. Integration was implemented through structured

Table 2. Software stack supporting the ViralFlow–MacWorP integrated platform.

Layer	Technology	Function
Frontend	Vue.js	User interface and workflow submission
Backend/API	Flask	Workflow services and orchestration
Messaging	RabbitMQ	Task scheduling and asynchronous execution
Database	PostgreSQL	Metadata and project management
Cache	Redis	Session and temporary data management
Workflow Engine	Nextflow/ViralFlow	Viral genomic analysis execution
Storage	NFS	Shared repository for inputs and outputs

workflow definitions specifying input data, parameter configuration and computational requirements for ViralFlow execution inside MacWorP. To ensure compatibility with the ViralFlow runtime environment, an intermediate launcher mechanism was incorporated into the workflow execution layer, enabling activation of ViralFlow dependencies prior to workflow submission.

Containerized deployment was adopted throughout the system to ensure portability, reproducibility and simplified installation. The platform was designed to satisfy functional and non-functional requirements relevant to genomic surveillance applications, including automated pipeline execution, parameter management, usability, scalability, reliability and reproducibility.

Rather than functioning as a simple software wrapper, the integration enables ViralFlow to operate as a native workflow inside a broader distributed orchestration environment, extending accessibility while preserving analytical reproducibility.

Source code and installation documentation are publicly available through the project repositories:

- MacWorP installation guide: `https://github.com/cubimedrub/macworp`
- ViralFlow installation: `https://github.com/WallauBioinfo/ViralFlow`

4.3. Experimental Setup

Proof-of-concept deployment and integration experiments were conducted in a Linux environment running Ubuntu 22.04 LTS. The test platform consisted of an 12th gen intel core 17-12650hx16 processor with 32 GB DDR4 RAM operating at 3200 MHz.

This environment was used to validate deployment, workflow interoperability and distributed execution behavior during preliminary system evaluation. Representative viral genomic datasets were employed to verify end-to-end execution from data submission to workflow outputs. Although the present study focuses on architectural feasibility and preliminary operational evaluation, this computational environment also provides the basis for future scalability and performance benchmarking.

4.4. Use of artificial intelligence tools

AI tools were used solely for editorial revision, graphical support and figure generation. Language models assisted in improving textual clarity and consistency. Figures were generated from structured descriptions provided by the authors, with all scientific content, interpretation and validation performed by the authors. No AI tools were used for data analysis or generation of scientific conclusions.

5. Results and Discussion

5.1. Integration and Workflow Execution

To enable interoperability between MacWorP and ViralFlow, both systems were deployed within a shared computational environment and integrated through modifications in the workflow execution layer. A functional challenge identified during deployment involved incompatibilities in dependency activation during Nextflow workflow execution, which were resolved through an intermediate launcher mechanism that activates the ViralFlow computational environment before workflow submission. This integration enabled ViralFlow to be executed as a native workflow inside MacWorP, preserving software dependencies while allowing automated submission through the graphical interface. The resulting architecture supports user-driven workflow execution, distributed processing through worker nodes, and centralized data management using NFS-shared storage. Proof-of-concept analyses demonstrated successful execution of representative viral genomic workflows, including arboviral and SARS-CoV-2 datasets, confirming the technical feasibility of coupling specialized genomics pipelines with a workflow orchestration platform.

5.2. Operational Evaluation

A preliminary evaluation was performed to assess the operational gains enabled by the integration. Rather than focusing exclusively on analytical outputs, the evaluation considered system-level performance indicators relevant for workflow accessibility and deployment. Table 3 presents a qualitative operational comparison between the proposed platform and representative workflow systems.

Table 3. Preliminary evaluation of the integrated platform.

Metric	Standalone ViralFlow	Integrated ViralFlow–MacWorP	Nextstrain	IRMA
Workflow submission	Command line	Web interface	Command line	Command line
User interaction steps	Multiple commands	Guided submission	Multiple commands	Multiple commands
Distributed execution	Limited	Enabled	Enabled	Enabled
Centralized data management	No	Yes	No	No
Workflow reproducibility	High	High	High	High
Accessibility (non-experts)	Limited	Improved	Limited	Limited
Average runtime	00:02:38	00:04:02	00:01:52	00:00:33

The comparison suggests that the proposed integration extends the accessibility and orchestration capabilities of ViralFlow while preserving analytical reproducibility. From a systems perspective, the main contribution lies not in replacing existing platforms, but in combining distributed workflow execution, accessible interfaces and public health-oriented genomic analysis within a unified environment.

5.3. Architectural Implications and Limitations

The proposed architecture supports distributed execution, centralized workflow management and potential expansion toward institutional or multi-user genomic surveillance environments. Its modular design also favors integration of additional analysis workflows. Although the proof-of-concept demonstrates feasibility, further benchmarking is needed to quantify scalability, computational overhead and usability under operational deployment scenarios.

6. Conclusions and Future Works

We presented an integrated platform combining ViralFlow and MacWorP to automate viral genomic surveillance analyses through distributed workflow orchestration and accessible web-based execution. The proof-of-concept implementation demonstrated the feasibility of coupling specialized viral genomics pipelines with scalable workflow management, reducing technical barriers while preserving reproducibility. Preliminary evaluation suggest that the proposed architecture improves accessibility and supports distributed execution, positioning the platform as a modular framework for genomic surveillance automation in public health laboratories. Future work includes large-scale performance benchmarking, usability evaluation with end users, and expansion toward multi-server deployments and additional surveillance workflows.

Acknowledgements

This work was supported by the National Council for Scientific and Technological Development (CNPq, grant no. 404352/2024-3).

References

Afgan et al. 2018. Afgan, E. et al. (2018). The galaxy platform for accessible, reproducible and collaborative biomedical analyses. *Nucleic Acids Research*, 46(W1):W537–W544.

da Silva et al. 2024. da Silva, A. F., da Silva Neto, A. M., Aksenen, C. F., Jeronimo, P. M. C., Dezordi, F. Z., Almeida, S. P., Costa, H. M. P., Salvato, R. S., Campos, T. d. L., Wallau, G. d. L., et al. (2024). Viralflow v1. 0—a computational workflow for streamlining viral genomic surveillance. *NAR genomics and bioinformatics*, 6(2):lqae056.

Di Tommaso et al. 2017. Di Tommaso, P. et al. (2017). Nextflow enables reproducible computational workflows. *Nature Biotechnology*, 35(4):316–319.

Gardy and Loman 2018. Gardy, J. L. and Loman, N. J. (2018). Towards a genomics-informed, real-time, global pathogen surveillance system. *Nature Reviews Genetics*, 19(1):9–20.

Hadfield et al. 2018. Hadfield, J. et al. (2018). Nextstrain: real-time tracking of pathogen evolution. *Bioinformatics*, 34(23):4121–4123.

Shepard et al. 2016. Shepard, S. S. et al. (2016). Viral deep sequencing needs an adaptive approach: Irma. *BMC Genomics*, 17(1):708.