Computing Architectures and Data Systems
REVIEW | COMMISSIONED SYNTHESIS

Resilient Data Pipelines for Scientific Computing

Dylan Howard - Corresponding author

Abstract

This review examines resilient data pipelines for scientific computing. The organizing question is how pipelines should preserve provenance and scientific meaning through retries, partial writes, schema change, and infrastructure failure. Ten related scholarly sources are synthesized through a decision-centered framework spanning problem definition, mechanism, measurement, evaluation, implementation, and governance. The review does not invent experiments, pooled estimates, or unreported quantitative results. It instead evaluates the strength and transferability of the available evidence, with particular attention to declaring computational success while silent corruption invalidates downstream evidence. The resulting framework links technical or empirical performance to explicit use conditions and identifies tests that should precede wider adoption in long-running and distributed scientific workflows.

Keywords: scientific computing, data pipelines, resilience, provenance, fault tolerance

Synthesis lens	Principal question
Mechanism	the chain connecting evidence inputs, resilient data pipelines for scientific computing, intermediate outputs, and downstream decisions
Evaluation	construct validity, comparative performance, uncertainty, transfer across settings, and decision utility
Primary risk	declaring computational success while silent corruption invalidates downstream evidence

1. Introduction

Research on resilient data pipelines for scientific computing sits at the boundary between a promising component and a consequential decision. The core question is how pipelines should preserve provenance and scientific meaning through retries, partial writes, schema change, and infrastructure failure. Answering it requires more than assembling favorable results. It requires a chain of evidence that makes the problem boundary, mechanism, measurement, comparator, uncertainty, and accountable decision visible to the reader.

This article presents a structured narrative review of ten related publications. Sources were selected for topical relevance, traceable publication metadata, and complementary coverage of technical, empirical, and governance questions. No meta-analytic pooling is attempted because the included studies differ in designs, populations, system boundaries, and outcomes. The purpose is decision-oriented synthesis: to identify what each source can support, where transfer remains uncertain, and which evidence would justify the next stage of use.

The central risk is declaring computational success while silent corruption invalidates downstream evidence. A top-line metric or broad policy label can appear decisive while concealing the conditions that produced it. Accordingly, the synthesis separates observation from interpretation and implementation from validation. This separation is especially important when the same system creates benefits for one user or institution while transferring cost, risk, or administrative burden to another.

2. Evidence Base and Problem Boundary

A defensible review begins by defining the unit of analysis. For the present topic, the relevant boundary includes inputs, institutional or technical context, the mechanism producing an output, the person or system acting on that output, and the downstream outcome. Evidence falls outside the boundary when it measures only an intermediate proxy yet is used to claim an end-to-end benefit.

Zhu et al. (2016) addresses "Fault-Tolerant Scheduling for Real-Time Scientific Workflows with Elastic Resource Provisioning in Virtualized Clouds." In this synthesis, that work is used as a source on problem definition within resilient data pipelines for scientific computing. The connection is deliberately bounded: the cited study informs the evidence map, but it does not by itself establish readiness for long-running and distributed scientific workflows. That distinction keeps the review close to the published record and prevents an adjacent result from being converted into a broader causal claim.

Tang (2021) addresses "Reliability-Aware Cost-Efficient Scientific Workflows Scheduling Strategy on Multi-Cloud Systems." In this synthesis, that work is used as a source on conceptual boundary within resilient data pipelines for scientific computing. The connection is deliberately bounded: the cited study informs the evidence map, but it does not by itself establish readiness for long-running and distributed scientific workflows. That distinction keeps the review close to the published record and prevents an adjacent result from being converted into a broader causal claim.

Chen et al. (2015) addresses "Dynamic and Fault-Tolerant Clustering for Scientific Workflows." In this synthesis, that work is used as a source on measurement within resilient data pipelines for scientific computing. The connection is deliberately bounded: the cited study informs the evidence map, but it does not by itself establish readiness for long-running and distributed scientific workflows. That distinction keeps the review close to the published record and prevents an adjacent result from being converted into a broader causal claim.

Together, these works provide complementary entry points, but their conclusions should not be flattened into a single ranking. Differences in data generation, setting, temporal horizon, and outcome definition may be substantively meaningful. A review should therefore preserve those differences and state where analogy, rather than direct replication, connects one domain to another.

3. Mechanism and System Design

The operative mechanism is the chain connecting evidence inputs, resilient data pipelines for scientific computing, intermediate outputs, and downstream decisions. This formulation discourages component-only reasoning. A model, platform, policy, assay, or infrastructure service acquires practical meaning only when its interfaces and use conditions are specified. The evidence chain should describe what information is available, what transformation occurs, which assumptions make that transformation credible, and who is authorized to act.

Li et al. (2021) addresses "Real-time and dynamic fault-tolerant scheduling for scientific workflows in clouds." In this synthesis, that work is used as a source on system mechanism within resilient data pipelines for scientific computing. The connection is deliberately bounded: the cited study informs the evidence map, but it does not by itself establish readiness for long-running and distributed scientific workflows. That distinction keeps the review close to the published record and prevents an adjacent result from being converted into a broader causal claim.

Bala and Chana (2015) addresses "Autonomic fault tolerant scheduling approach for scientific workflows in Cloud computing." In this synthesis, that work is used as a source on representation choice within resilient data pipelines for scientific computing. The connection is deliberately bounded: the cited study informs the evidence map, but it does not by itself establish readiness for long-running and distributed scientific workflows. That distinction keeps the review close to the published record and prevents an adjacent result from being converted into a broader causal claim.

Ahmad et al. (2020) addresses "A fault-tolerant workflow management system with Quality-of-Service-aware scheduling for scientific workflows in cloud computing." In this synthesis, that work is used as a source on failure analysis within resilient data pipelines for scientific computing. The connection is deliberately bounded: the cited study informs the evidence map, but it does not by itself establish readiness for long-running and distributed scientific workflows. That distinction keeps the review close to the published record and prevents an adjacent result from being converted into a broader causal claim.

These studies also show why modular claims require interface tests. A component may perform well while the complete pathway fails because inputs drift, users interpret outputs differently, recovery semantics are ambiguous, or institutional incentives change. Design documentation should therefore include not only the intended pathway but also foreseeable deviations, degraded modes, and conditions under which the system should stop or defer.

4. Evaluation and Uncertainty

Evaluation should center on construct validity, comparative performance, uncertainty, transfer across settings, and decision utility. Average performance remains useful, but it should be accompanied by distributions, error categories, relevant subgroups or operating regimes, and uncertainty at the point where a decision changes. Comparators must isolate the value of the proposed addition rather than compare a mature intervention with an intentionally weak baseline.

Alaie et al. (2022) addresses "A hybrid bi-objective scheduling algorithm for execution of scientific workflows on cloud platforms with execution time and reliability approach." In this synthesis, that work is used as a source on comparative evaluation within resilient

data pipelines for scientific computing. The connection is deliberately bounded: the cited study informs the evidence map, but it does not by itself establish readiness for long-running and distributed scientific workflows. That distinction keeps the review close to the published record and prevents an adjacent result from being converted into a broader causal claim.

Alaei et al. (2020) addresses "An adaptive fault detector strategy for scientific workflow scheduling based on improved differential evolution algorithm in cloud." In this synthesis, that work is used as a source on uncertainty within resilient data pipelines for scientific computing. The connection is deliberately bounded: the cited study informs the evidence map, but it does not by itself establish readiness for long-running and distributed scientific workflows. That distinction keeps the review close to the published record and prevents an adjacent result from being converted into a broader causal claim.

External validation should introduce meaningful shifts in setting, time, population, workload, market structure, or environmental conditions. A nominally independent sample can still be too similar to test transfer. Conversely, a failed transfer is scientifically informative when the changed conditions are measured and the mechanism of failure is examined rather than hidden in an aggregate score.

5. Translation and Governance

Translation to long-running and distributed scientific workflows depends on more than technical readiness. Decision rights, documentation, maintenance, monitoring, appeal, and recovery are part of the intervention. The governance requirement is traceable evidence, named responsibility, version control, monitoring, appeal, and explicit revalidation. Each control should be connected to a named failure mode and should identify an owner, evidence source, review interval, and escalation path.

Khaldi et al. (2019) addresses "Fault tolerance for a scientific workflow system in a Cloud computing environment." In this synthesis, that work is used as a source on implementation within resilient data pipelines for scientific computing. The connection is deliberately bounded: the cited study informs the evidence map, but it does not by itself establish readiness for long-running and distributed scientific workflows. That distinction keeps the review close to the published record and prevents an adjacent result from being converted into a broader causal claim.

Li et al. (2022) addresses "Fault-tolerant scheduling and data placement for scientific workflow processing in geo-distributed clouds." In this synthesis, that work is used as a source on governance within resilient data pipelines for scientific computing. The connection is deliberately bounded: the cited study informs the evidence map, but it does not by itself establish readiness for long-running and distributed scientific workflows. That distinction keeps the review close to the published record and prevents an adjacent result from being converted into a broader causal claim.

A credible implementation plan also defines sunset and revalidation conditions. New data, software updates, institutional restructuring, population change, or shifts in incentives can invalidate previously reasonable assumptions. Monitoring should therefore test the validity of the evidence chain, not merely whether a service remains online or a headline metric remains stable.

6. Research Agenda

Future studies should preregister or otherwise predefine the intended decision, principal outcomes, exclusions, and analysis plan when feasible. They should report missingness, sensitivity analyses, negative or null findings, and the cost of errors to differently situated stakeholders. Reproducible artifacts should include sufficient metadata to reconstruct data provenance, model or analytical versions, parameter choices, and the sequence from evidence to conclusion.

Cross-disciplinary work needs a shared object of explanation rather than a collection of adjacent vocabularies. For resilient data pipelines for scientific computing, that shared object is the complete decision pathway. Technical, clinical, social, environmental, or economic expertise should each change the design or interpretation of the study. When evidence remains incomplete, the useful output is a bounded hypothesis, a transparent uncertainty statement, or a request for additional measurement.

7. Conclusion

The literature supports a disciplined approach to resilient data pipelines for scientific computing: define the decision, preserve system boundaries, test consequential failure modes, connect uncertainty to action, and assign accountable control. The strongest conclusion is not that one method is universally superior. It is that credible use in long-running and distributed scientific workflows requires an auditable link from source evidence to design choice, evaluation result, and governed decision.

Declarations

Ethics approval: Not applicable. This article synthesizes previously published literature and reports no new research involving humans, animals, human tissue, or identifiable sensitive data.

Consent to participate and consent for publication: Not applicable.

Funding: No external funding is reported for this commissioned synthesis.

Competing interests: The authoring group declares no competing interests for this commissioned synthesis.

AI-assisted writing: Generative AI supported drafting, source organization, and language editing. Accountable authors and editors must verify every claim, citation, and disclosure before publication.

Data, code, and materials availability: No new dataset or code was generated. All evidence is identified in the reference list.

Licence: This manuscript is prepared for publication under the Creative Commons Attribution 4.0 International licence (CC BY 4.0).

References

Ahmad, Z., Nazir, B., & Umer, A. (2020). A fault-tolerant workflow management system with Quality-of-Service-aware scheduling for scientific workflows in cloud computing. International Journal of Communication Systems, 34(1). https://doi.org/10.1002/dac.4649

Alaei, M., Khorsand, R., & Ramezanpour, M. (2020). An adaptive fault detector strategy for scientific workflow scheduling based on improved differential evolution algorithm in cloud. Applied Soft Computing, 99, 106895. https://doi.org/10.1016/j.asoc.2020.106895

Alaie, Y. A., Shirvani, M. H., & Rahmani, A. M. (2022). A hybrid bi-objective scheduling algorithm for execution of scientific workflows on cloud platforms with execution time and reliability approach. The Journal of Supercomputing, 79(2), 1451-1503. https://doi.org/10.1007/s11227-022-04703-0

Bala, A., & Chana, I. (2015). Autonomic fault tolerant scheduling approach for scientific workflows in Cloud computing. Concurrent Engineering, 23(1), 27-39. https://doi.org/10.1177/1063293x14567783

Chen, W., Silva, R. F. D., Deelman, E., & Fahringer, T. (2015). Dynamic and Fault-Tolerant Clustering for Scientific Workflows. IEEE Transactions on Cloud Computing, 4(1), 49-62. https://doi.org/10.1109/tcc.2015.2427200

Khaldi, M., Rebbah, M., Meftah, B., & Smail, O. (2019). Fault tolerance for a scientific workflow system in a Cloud computing environment. International Journal of Computers and Applications, 42(7), 705-714. https://doi.org/10.1080/1206212x.2019.1647651

Li, C., Liu, J., Wang, M., & Luo, Y. (2022). Fault-tolerant scheduling and data placement for scientific workflow processing in geo-distributed clouds. Journal of Systems and Software, 187, 111227. https://doi.org/10.1016/j.jss.2022.111227

Li, Z., Chang, V., Hu, H., Hu, H., Li, C., & Ge, J. (2021). Real-time and dynamic fault-tolerant scheduling for scientific workflows in clouds. Information Sciences, 568, 13-39. https://doi.org/10.1016/j.ins.2021.03.003

Tang, X. (2021). Reliability-Aware Cost-Efficient Scientific Workflows Scheduling Strategy on Multi-Cloud Systems. IEEE Transactions on Cloud Computing, 10(4), 2909-2919. https://doi.org/10.1109/tcc.2021.3057422

Zhu, X., Wang, J., Guo, H., Zhu, D., Yang, L. T., & Liu, L. (2016). Fault-Tolerant Scheduling for Real-Time Scientific Workflows with Elastic Resource Provisioning in Virtualized Clouds. IEEE Transactions on Parallel and Distributed Systems, 27(12), 3501-3517. https://doi.org/10.1109/tpds.2016.2543731