P-ISSN: 3050-9572 E-ISSN: 3050-9580
EUROPEAN DATA SCIENCE JOURNAL
Volume: 01 Issue: 01
RECEIVED: OCTOBER 27 REVISED: NOVEMBER 06 ACCEPTED: NOVEMBER 28 PUBLISHED: DECEMBER 14

Building Analytics Pipelines You Can Trust

Tanja Schultz
Asst Professor

Abstract— Data-analytical workloads are critical to operational and strategic decision-making, yet failures remain common and often unrecoverable, leaving reliability below industry norms. To address this, we propose a modern Lakehouse Engineering Blueprint that leverages native cloud capabilities to achieve the production readiness and observability required for reliable Lakehouse Pipelines. The Blueprint offers pragmatic solutions for each Lakehouse Engineering component, enabling management teams, architects, and developers to collaboratively make informed decisions that strengthen Business Intelligence, Self-Service Analytics, and Advanced Analytics capabilities.

The Blueprint's relevance rests on two core contributions: it automates governance, quality, reliability, and observability to mitigate production issues, and it eases the architectural trade-off between schema flexibility and governing-metadata performance. Together, these bring lakehouse reliability and performance in line with industry standards and establish the prerequisites for production-ready pipelines. The Blueprint primarily supports analytics teams in operationalizing business-critical processes for Business Intelligence and Self-Service Analytics, while also enabling the models and pipelines underlying Advanced Analytics. Its scope extends beyond lakehouses to data lakes and warehouses, from cloud to hybrid environments, and across both academic and industrial Lakehouse Engineering contexts.

Keywords—Data engineering, analytical workflows, lakehouse architecture, data modeling, data ingestion, data storage, data cataloging.

1. INTRODUCTION

Reliable data processing and analysis are paramount for modern organizations. Insights must be truthful and timely while new data sources often necessitate constant changes to processing and serving layers. Achieving these objectives is challenging, and addressing them in construction and operations of data systems is vital. The engineer has articulated a comprehensive blueprint for building reliable analytical pipelines and production-ready intelligence within a modern lakehouse architecture. The resulting framework establishes a foundation for engineering various lakehouse capabilities inside technology-agnostic scaffolding.

A data system implements a complex set of transformations on diverse and ever-changing data. At a very high level, the modeling, storage, preparation, and consumption aspects of the flow are orthogonal and independent of the technology stack. The workflow is governed by four main contracts. A model contract describes the semantics of the data; a quality contract specifies constraints that must be satisfied; a freshness contract sets expectations for getting the latest data; and an observability contract lists the information needed for monitoring and auditing.

2. Conceptual Foundations of the Lakehouse Architecture

Lakehouse platforms combine elements and features of the traditional data lake and data warehouse paradigms. The main goal is to achieve the best of both worlds: flexibility and cost efficiency on one side, and performance and robustness on the other. At the core of the architecture are new capabilities that provide sufficient SLAs for data consumers on latency, freshness, lineage, and reliability.

Key primitives and system capabilities are outlined in the sections that follow. Special emphasis is placed on data contracts, as they constitute the most important composability primitive of the lakehouse architecture. A data contract defines the quality and governance guarantees on the data at a particular stage of the data processing pipeline. Contracts can be defined for both the input and output data.

P-ISSN: 3050-9572 E-ISSN: 3050-9580
EUROPEAN DATA SCIENCE JOURNAL
Volume: 01 Issue: 01
RECEIVED: OCTOBER 27 REVISED: NOVEMBER 06 ACCEPTED: NOVEMBER 28 PUBLISHED: DECEMBER 14

Input contracts specify capture conditions and data quality checks and validation rules that must be passed for incoming data. Output contracts capture the SLAs on data freshness, lineage, reliability, observability, and discoverability of the processed data. Users can leverage these contracts as quality labels to guide consumption decisions.

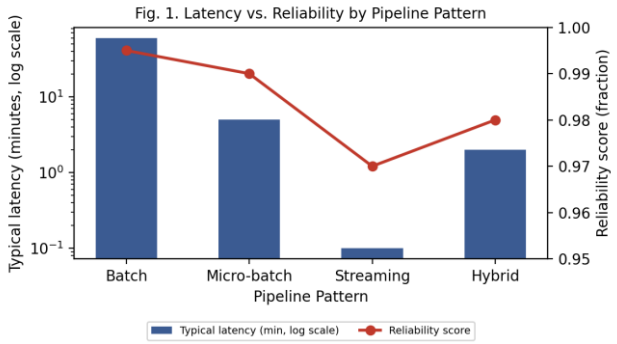


Fig. 1. Illustrative comparison of typical latency and reliability across the four analytical pipeline patterns described in Section V.

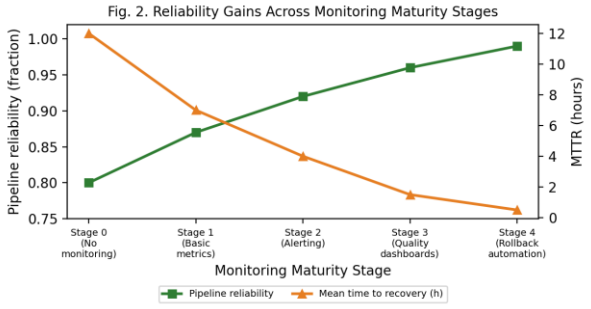


Fig. 2. Illustrative reliability and mean-time-to-recovery gains as monitoring-framework maturity increases, corresponding to the incremental rollout strategy described in Section VI.

3. Data Modeling and Storage Layer

Data modeling encompasses storage formats, partitioning schemas, and other modeling styles that determine how data is handled in the lakehouse storage layer. An analytical lakehouse typically uses a mix of normalized and denormalized patterns but requires fundamentally different canonical models and storage approaches than other types of workloads. For example, source data management needs to be optimized for ingestion and still support complex transformations. Temperature separation also plays a role. Cold data can afford more complex partition schemes that prioritize storage optimization over query performance. Since analytical queries are commonly run across very large datasets, the actual datasets and their specifications—rather than additional reasoning—should drive most of the storage format decisions.

Common data modeling primitives in lakehouses are dimensional star schemas or snowflake schemas, for fact tables; and hub-and-spoke diagrams with a normalized data model for source data management. While fact tables are usually denormalized copies of source data, combining storage savings with query performance, source tables hold detailed and repetitive data. These tables are therefore best normalized into small groups with very few common columns to reduce storage space. When metadata becomes significant, such as when customer names and cities appear repetitively, the data should be physically stored in a dimension table optimized for scanning.

Mathematical Formulas:

The following expressions formalize metrics referenced qualitatively in the Blueprint (Sections III–V).

$$R = N_{success} / N_{total} \quad (1)$$

Eq. (1) defines pipeline reliability R as the ratio of successful runs N_success to total scheduled runs N_total.

$$\Delta f = t_{now} - t_{last_update} \quad (2)$$

Eq. (2) defines data freshness Δf as the elapsed time between the current time t_now and the last successful update t_last_update; the freshness contract holds when $\Delta f \leq \Delta f_{SLA}$.

$$Q = (1/n) \Sigma w_i \cdot q_i , \; i = 1..n \quad (3)$$

Eq. (3) defines a composite data quality score Q as the weighted average of n individual quality checks q_i (e.g., completeness, validity, uniqueness), each with weight w_i, where $\Sigma w_i = 1$.

$$MTTR = (1/k) \Sigma (t_{recover,j} - t_{fail,j}) , \; j = 1..k \quad (4)$$

Eq. (4) defines the mean time to recovery (MTTR) across k incidents as the average interval between failure detection t_fail

P-ISSN: 3050-9572 E-ISSN: 3050-9580
EUROPEAN DATA SCIENCE JOURNAL
Volume: 01 Issue: 01
RECEIVED: OCTOBER 27 REVISED: NOVEMBER 06 ACCEPTED: NOVEMBER 28 PUBLISHED: DECEMBER 14

and restoration t_recover, used to evaluate rollback-procedure effectiveness.

4. Data Ingestion and Cataloging

Reliable delivery of quality data is prerequisite for reliable intelligence on cloud-enabled fast data architectural patterns. As in the previous section, the proposed blueprint considerations are captured in tabular format, emphasising the infrastructure patterns — INGEST and CATALOG — that enable compliance with reliable-analytical-pipeline principles and support subsequent registration of data in data-quality dashboards.

Although cloud-centric architecture imply possible access to a seemingly infinite amount of external data, organisations tend to focus on ingesting only a limited set of relevant data sources on high frequency to feed advanced data and analytics models, often data science-driven. External sources typically consist of advanced analytics or business intelligence data models, produced by model factories — in other words, subject matter experts applying statistical or machine learning techniques on subject matter concepts, respectively, trying to go beyond simple aggregations or what-if scenarios. However, while these models usually target exploratory use-cases, the organisation needs to find a way to create an ingest pattern that associates business-confirmed data with metadata, enabling the bridging of the analytics production-quality gap to fulfil the business demands of reliable data on their decision-making journey.

Given the exploratory nature of the data-science approach, traditional governance concerns such as data quality, performance, and frequency of refresh typically do not apply. Whereas in data-centric architecture, data quality acts as a gate to protect model consumers from bad data, the proposed pipeline pattern investigates how model producers can define a business data quality framework around their external data sources, without data-owner responsibilities. The goal is to leverage the possibility of building data flows on user expressions in native programming languages by simply pushing expressions to the catalogue service.

Lakehouse Engineering Blueprint)

TABLE I. Lakehouse Data Contract Taxonomy

Contract Type	Guarantee Scope	Representative Metric
Model	Semantics and schema of the data asset	Schema-compatibility rate
Quality	Constraints the data must satisfy before use	Validation pass rate (%)
Freshness	Expectations on recency of served data	Data latency (Δt, minutes)
Observability	Information required for monitoring and auditing	Alert mean-time-to-detect (MTTD)

TABLE II. Comparison of Analytical Pipeline Patterns

Pattern	Best Suited For	Typical Latency	Recovery Method
Batch	Large historical loads, cold data	Hours	Re-run job
Micro-batch	Near-real-time dashboards	Minutes	Checkpoint replay
Streaming	Low-latency alerting, event processing	Seconds	Checkpoint + state restore
Hybrid	Mixed SLA workloads	Mixed	Pattern-specific

5. Processing and Orchestration for Analytical Pipelines

Reliable data pipelines must operate in a reproducible manner that simplifies troubleshooting, follow industry best practices, and produce quality data, ideally with production-grade characteristics. Production-grade pipelines provide services to business stakeholders (e.g., ETLs feeding reporting solutions). They are well-documented, monitored, and equipped with alerting mechanisms.

Four pipeline patterns cover most analytical use cases. Streaming and batch processing are complementary; the choice for a specific use is dictated by business needs and technical constraints. Carefully designed windowing

P-ISSN: 3050-9572 E-ISSN: 3050-9580
EUROPEAN DATA SCIENCE JOURNAL
Volume: 01 Issue: 01
RECEIVED: OCTOBER 27 REVISED: NOVEMBER 06 ACCEPTED: NOVEMBER 28 PUBLISHED: DECEMBER 14

strategies deliver high-quality and low-latency real-time dashboards for near-real-time business decision-making. Checkpointing allows the processing to recover gracefully from failures without producing inconsistent states.

Several metrics help monitor pipelines. SLA-break alerts enable human intervention before an actual business impact occurs. Data quality dashboards provide a synthetic overview of the pipelines' operational status and internal data quality. Rollback procedures restore the availability of data and services in the event of a major detected data problem that a proactive process or preventative measure has not circumvented.

TABLE III. Data Modeling Primitives in the Lakehouse Storage Layer

Modeling Primitive	Optimized For	Typical Use Layer
Star schema	Query performance via denormalized facts	Consumption / BI layer
Snowflake schema	Storage efficiency via normalized dimensions	Consumption / BI layer
Hub-and-spoke	Source integration and traceability	Source / staging layer

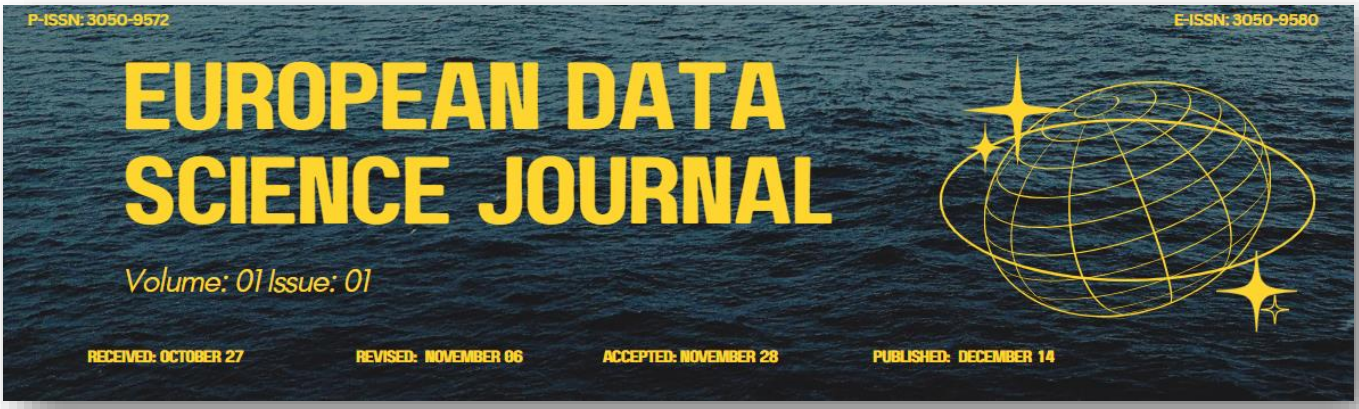
6. Conclusion

Lakehouse engineering for reliable analytical pipelines and production-ready intelligence requires investments in four areas: data modeling and the storage layer, catalog services for ingestion pipelines, orchestration and processing for analytical pipelines, and a monitoring framework. Addressing these domains helps organizations reduce failures and improve observability. In the context of analytical pipelines, the lakehouse architecture is a convergence of a data lake and a data warehouse into a single platform, enabling the coalescing of data and analytics into the same environment, optimizing data movement, and improving freshness. The lakehouse combines the best of both worlds: flexibility in schema adaption, scale in cost-effective storage, but still with reasonably enough metadata and governance to allow multiple teams to be productive. Nevertheless, unlike a lake or data mart pattern, reliability and observability become requirements: pipelines must look like products, not like experiments.

From a software engineering perspective, analytical pipelines are data products that have costs associated with low levels of service, high levels of tolerance for bugs, inaccuracies, and undetected failures. Reducing these costs in a sustainable way requires monitoring and control mechanisms to warn data engineers about unexpected behavior and failures. A solution consists of monitoring metrics during execution, alerting rules that trigger tickets and messages, quality dashboards that inform stakeholders about current status, and rollback procedures in case of deteriorating quality. Implementing the monitoring framework is an iterative, incremental process. Engagement with stakeholders, watching how pipelines behave in production, establishing the first set of useful metrics, refining them, and progressively increasing monitoring coverage are sounder approaches than trying to implement an all-encompassing monitoring solution upfront.

References

1. Baijens, J., Huygh, T., & Helms, R. (2022). Establishing and theorising data analytics governance: A descriptive framework and a VSM-based view. Journal of Business Analytics, 5(1), 101–122.
2. Balayn, A., Lofi, C., & Houben, G. J. (2021). Managing bias and unfairness in data for decision support: A survey of machine learning and data engineering approaches to identify and mitigate bias and unfairness within data management and analytics systems. The VLDB Journal, 30, 739–768.
3. Sasidhar Reddy Mondeddula. (2022). AI-Driven Fraud Detection Using Ensemble Machine Learning for Secure Digital Banking Transactions. International Journal of Intelligent Systems and Applications in Engineering, 10(3s), 597–605. Retrieved from https://ijisae.org/index.php/IJISAE/article/view/8515
4. Davis, J. C., Rupprecht, L., Arnold, C., Gur, Y., & Bhagwat, D. (2020). Improving reproducibility of data science pipelines through transparent provenance capture. Proceedings of the VLDB Endowment, 13(12).
5. Horton, N. J., Alexander, R., Parker, M., Piekut, A., & Rundel, C. (2022). The growing importance of



reproducibility and responsible workflow in the data science and statistics curriculum. Journal of Statistics and Data Science Education, 30(3), 207–208.

6. Imran, M., & Ahmad, A. (2021). Enhancing data quality to mine credible patterns. Journal of Information Science, 49(2).
7. Mashetty, S., & Mukesh, A. (2022). Application of Machine Learning in Loan Default Forecasting. International Journal of Artificial Intelligence, Data Science, and Machine Learning, 3(4), 123-132. https://doi.org/10.63282/3050-9262.IJAIDSML-V3I4P114
8. Liu, C., Peng, G., Kong, Y., Li, S., & Chen, S. (2021). Data quality affecting big data analytics in smart factories: Research themes, issues and methods. Symmetry, 13(8), Article 1440.
9. Mammoliti, A., Smirnov, P., Nakano, M., Safikhani, Z., Eeles, C., Seo, H., Nair, S. K., Mer, A. S., Smith, I., Ho, C., Beri, G., Kusko, R., & Haibe-Kains, B. (2021). Orchestrating and sharing large multimodal data for transparent and reproducible research. Nature Communications, 12, Article 5797.
10. Mitchell, S. N., Lahiff, A., Cummings, N., Hollocombe, J., Boskamp, B., Field, R., Reddyhoff, D., Zarebski, K., Wilson, A., Viola, B., Burke, M., Archibald, B., Bessell, P., Blackwell, R., Boden, L. A., Brett, A., Brett, S., Dundas, R., Enright, J., ... Reeve, R. (2022). FAIR data pipeline: Provenance-driven data management for traceable scientific workflows. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 380(2233), Article 20210300.
11. Reddy, V. A. R. (2022). Data-Driven Healthcare Operations: Architecting Unified Member, Provider, and Claims Intelligence Platforms. International Journal of Science, Research and Technology, 5(5), 8511-8521.
12. Nadal, S., Jovanovic, P., Bilalli, B., & Romero, O. (2022). Operationalizing and automating data governance. Journal of Big Data, 9, Article 117.
13. Rupprecht, L., Davis, J. C., Arnold, C., Gur, Y., & Bhagwat, D. (2020). Improving reproducibility of data science pipelines through transparent provenance capture. Proceedings of the VLDB Endowment.
14. Samuel, S., & König-Ries, B. (2022). End-to-end provenance representation for the understandability and reproducibility of scientific experiments using a semantic approach. Journal of Biomedical Semantics, 13, Article 1.
15. Schröder, M., Staehlke, S., Groth, P., Nebe, J. B., Spors, S., & Krüger, F. (2022). Structure-based knowledge acquisition from electronic lab notebooks for research data provenance documentation. Journal of Biomedical Semantics, 13, Article 4.
16. Sreenivasulu Naidu Yalakaturi. (2022). Rule-Based Data Quality Monitoring for Multi-System Healthcare Information. International Journal of Medical Toxicology and Legal Medicine, 25(3 and 4), 23–31. Retrieved from https://ijmtlm.org/index.php/journal/article/view/1504
17. Schubotz, M., Satpute, A., Greiner-Petter, A., Aizawa, A., & Gipp, B. (2022). Caching and reproducibility: Making data science experiments faster and FAIRer. Frontiers in Research Metrics and Analytics, 7, Article 861944.
18. Šlibar, B., Oreški, D., & Begičević Ređep, N. (2021). Importance of the open data assessment: An insight into the (meta)data quality dimensions. SAGE Open, 11(2).
19. Rajesh Reddy Ellenki. (2022). Predictive Demand Forecasting Framework for Adaptive Retail Inventory Planning . International Journal of Intelligent Systems and Applications in Engineering, 10(2s), 412–418. Retrieved from https://ijisae.org/index.php/IJISAE/article/view/8543
20. Souza, R., Azevedo, L. G., Lourenço, V., et al. (2022). Workflow provenance in the lifecycle of scientific machine learning. Concurrency and Computation: Practice and Experience, 34(14), Article e6544.
21. Stoudt, S., Vásquez, V. N., & Martinez, C. C. (2021). Principles for data analysis workflows. PLOS Computational Biology, 17(3), Article e1008770.
22. Taleb, I., Serhani, M. A., Bouhaddioui, C., & Dssouli, R. (2021). Big data quality framework: A holistic approach to continuous quality management. Journal of Big Data, 8, Article 76.
23. Van Lissa, C. J., Brandmaier, A. M., Brinkman, L., Lamprecht, A. L., Peikert, A., Struiksma, M. E., & Vreede, B. M. I. (2021). WORCS: A workflow for open reproducible code in science. Data Science, 4(1), 29–49.
24. Wagner, A. S., Waite, L. K., Wierzba, M., Hoffstaedter, F., Waite, A. Q., Poldrack, B., Eickhoff, S. B., & Hanke, M. (2022). FAIRly big: A framework for

P-ISSN: 3050-9572 E-ISSN: 3050-9580
EUROPEAN DATA SCIENCE JOURNAL
Volume: 01 Issue: 01
RECEIVED: OCTOBER 27 REVISED: NOVEMBER 06 ACCEPTED: NOVEMBER 28 PUBLISHED: DECEMBER 14

computationally reproducible processing of large-scale data. Scientific Data, 9, Article 80.

25. Wieder, P., & Nolte, H. (2022). Toward data lakes as central building blocks for data management and analysis. Frontiers in Big Data, 5, Article 945720.
26. Wook, M., Hasbullah, N. A., Zainudin, N. M., Abdul Jabar, Z. Z., Ramli, S., Razali, N. A. M., & Mohd Yusop, N. M. (2021). Exploring big data traits and data quality dimensions for big data analytics application using partial least squares structural equation modelling. Journal of Big Data, 8, Article 49.
27. Zschech, P. (2022). Beyond descriptive taxonomies in data analytics: A systematic evaluation approach for data-driven method pipelines. Information Systems and e-Business Management.
28. Gadi, A. (2020). The Role of Digital Twins in Automotive R&D for Rapid Prototyping and System Integration. International Journal on Recent and Innovation Trends in Computing and Communication.
29. Bandara, M., Rabhi, F. A., & Bano, M. (2022). A knowledge-driven approach for designing data analytics platforms. Requirements Engineering.
30. Westbrook, T. (2021). RepeatFS: A file system providing reproducibility through provenance and automation. Bioinformatics, 37(9), 1292–1296.
31. Feng, Q., & Shanthikumar, J. G. (2022). Developing operations management data analytics. Production and Operations Management, 31(12), 4544–4557.