International Journal of AI, BigData, Computational and Management Studies
Noble Scholar Research Group | Volume 3, Issue 3, PP 91-99, 2022
ISSN: 3050-9416 | https://doi.org/10.63282/3050-9416.IJAIBDCMS-V3I3P111

Optimizing Healthcare ETL Pipelines with Hybrid Cloud Data Warehousing: A Case Study Using Snowflake and Azure Data Factory

Satya Manesh Veerapaneni
Independent Researcher Fremont, CA, USA.

Abstract: The exponential growth of healthcare data driven by electronic health records (EHRs), Internet of Medical Things (IoMT) devices, and evolving regulatory mandates necessitates scalable and secure data integration pipelines. Traditional ETL architectures struggle to meet the demands of real-time processing, hybrid cloud interoperability, and compliance with standards such as HIPAA and HL7. This paper presents a hybrid cloud ETL framework leveraging Azure Data Factory (ADF) and Snowflake to enable scalable, secure, and resilient healthcare data pipelines. The architecture integrates on-premises clinical systems with cloud-native services using a self-hosted integration runtime, facilitating ingestion, transformation, and analytics at scale. We evaluate the system through a real-world deployment at a multihospital healthcare network in the Midwest United States. The implementation achieved a 41% reduction in data latency and a 60% decrease in infrastructure overhead compared to legacy systems. Key contributions include dynamic schema handling, endto-end encryption, audit-ready transformation workflows, and optimized parallel loading. We also identify challenges around cost governance, schema drift, and network reliability. This work provides a replicable model for healthcare organizations seeking to modernize their data engineering infrastructure using hybrid cloud technologies.

Keywords: Healthcare, ETL, Azure Data Factory, Snowflake, Hybrid Cloud, Data Warehousing, Data Integration, HL7, HIPAA, Real-Time Processing.

1. Introduction

The healthcare industry is undergoing a rapid digital transformation fueled by electronic health records (EHRs), IoTenabled medical devices, administrative systems, and regulatory compliance mandates such as HIPAA and HITECH. This digitization is generating massive volumes of structured, semistructured, and unstructured data, creating both opportunities and challenges for healthcare providers. Extracting actionable insights from this data is critical for improving patient care, operational efficiency, population health management, and regulatory reporting.

Traditional Extract, Transform, Load (ETL) pipelines, primarily built for on-premises environments, are ill-equipped to handle the growing demands of real-time analytics, elastic scalability, and distributed data sources. These pipelines often suffer from high latency, rigid schema dependencies, and complex infrastructure management overheads. Moreover, healthcare datasets frequently evolve over time due to changing standards (e.g., HL7, FHIR) and vendor-specific schema modifications, further complicating data integration efforts.

Hybrid cloud architectures are increasingly being adopted by healthcare organizations to bridge the gap between legacy systems and modern analytics platforms. A hybrid approach allows sensitive patient data to remain in premises or private environments while enabling scalable processing and storage in the public cloud. This architectural pattern offers the flexibility to comply with data residency requirements while leveraging cloud native services for analytics and automation.

In this paper, we present a hybrid cloud data pipeline architecture that integrates Microsoft Azure Data Factory (ADF) and Snowflake, a native cloud data warehouse [1]. Azure Data Factory acts as the data orchestration engine, providing connectivity to both cloud and on-premises sources through its self-hosted integration runtime. Snowflake serves as the centralized analytics platform, offering features such as virtual warehouses for concurrent compute, time travel for auditing, and support for semi-structured formats like JSON and Avro, features that are especially relevant for healthcare applications.

We evaluate this architecture through a case study involving a regional healthcare system in the Midwest United States, comprising five hospitals and over 30 outpatient centers. The organization sought to modernize its existing ETL processes to support cross-facility analytics, improve pipeline reliability, and ensure end-to-end HIPAA compliance. Our contributions are as follows:

- We design and implement a scalable and secure ETL architecture tailored for healthcare data integration using Azure Data Factory and Snowflake.
- We demonstrate a real-world case study showcasing a 41% reduction in data latency, 60% reduction in infrastructure overhead, and improved fault tolerance.

- We address operational challenges such as schema drift, secure hybrid data movement, and pipeline monitoring in compliance-driven environments.
- We propose a framework for extending this architecture to support future capabilities such as real-time FHIR streaming and AI-based analytics.

2. Related Work

Integrating healthcare data remains a persistent challenge due to the heterogeneity of clinical systems, the stringent requirements of privacy regulations, and the sheer volume and velocity of generated data. Numerous ETL frameworks have been developed to address these concerns, ranging from traditional on-premises solutions to modern cloud-native architectures.

Previous analyses of biomedical data warehousing have highlighted the limitations of conventional ETL pipelines in accommodating evolving data standards such as HL7, FHIR, and ICD-10. These traditional pipelines often lack the flexibility to manage schema evolution, ensure patient data confidentiality, or integrate diverse data types spanning structured, semi-structured, and unstructured formats.

Legacy tools including Informatica PowerCenter, IBM InfoSphere, and SQL Server Integration Services (SSIS) have long served as the foundation for healthcare ETL systems. While effective for structured batch processing, these platforms are typically constrained by rigid schema definitions, limited scalability, and high operational overhead when deployed in distributed or dynamic environments [2].

With the advent of cloud computing, newer platforms such as Google Cloud Dataflow, AWS Glue, and Azure Data Factory have emerged to support complex, scalable data orchestration [3]. Among these, Azure Data Factory offers hybrid integration capabilities via self-hosted integration runtimes, enabling secure connectivity across both on-premises and cloud-native environments. This functionality is particularly important for healthcare institutions seeking to retain control over sensitive data while leveraging cloud-based analytics [4].

Snowflake has introduced a significant shift in the data warehousing paradigm by decoupling compute and storage, supporting semi-structured formats naively, and offering virtual warehouse isolation for concurrent workloads. Although Snowflake has shown promising results in domains such as retail and finance, its application within regulated healthcare environments, particularly in conjunction with hybrid ETL workflows, has not been widely documented.

Some initial efforts have explored hybrid data integration models in healthcare by combining open-source ETL tools with distributed storage platforms such as Hadoop [5], [6]. However, these solutions often lack native support for realtime orchestration and do not capitalize on the operational simplicity and elasticity offered by commercial cloud platforms [7].

Despite advancements in tooling and architecture, there remains a notable absence of applied case studies that evaluate the integration of cloud-native orchestration tools like Azure Data Factory with modern data warehouses such as Snowflake in healthcare contexts. This paper seeks to fill that gap by providing an empirical case study of a real-world deployment focused on latency reduction, operational efficiency, and regulatory alignment through a hybrid ETL framework.

3. Architecture Overview

This section outlines the hybrid cloud architecture deployed to optimize healthcare data extraction, transformation, and loading using Azure Data Factory (ADF) and Snowflake [8]. The architecture was designed to address the following objectives:

- Minimize latency in batch and near-real-time healthcare data pipelines.
- Ensure HIPAA-compliant data handling across on premises and cloud resources [9].
- Achieve operational scalability with reduced maintenance overhead.
- Support flexible schema evolution and semi-structured healthcare data formats.

3.1. System Components

The architecture consists of four major components:

The implemented hybrid architecture comprises four primary components, each playing a distinct role in the overall data pipeline. At the source layer, the architecture interfaces with a variety of on-premises systems, including Electronic Health Records (EHRs), laboratory information systems, radiology archives using PACS (Picture Archiving and Communication Systems), and standalone pharmacy databases. These systems are hosted entirely within the hospital's private data center to ensure adherence to data sovereignty and security policies.

Azure Data Factory (ADF) serves as the central orchestration platform for the ETL pipelines. It is responsible for coordinating data extraction, scheduling recurring workflows, managing transformation logic through mapping data flows, and

handling errors and execution logging. To enable secure communication between cloud services and the on-premises infrastructure, ADF is configured with a self-hosted integration runtime (IR), deployed behind the hospital firewall.

Azure Blob Storage functions as the intermediary staging layer within the pipeline. It temporarily stores both raw and pre-processed datasets extracted from source systems. This component ensures durable, scalable, and encrypted storage for intermediate data before it is ingested into the analytics warehouse. Additionally, it facilitates decoupling of extraction and loading phases, thus enhancing fault tolerance and system modularity.

Finally, Snowflake operates as the cloud-based data warehouse and primary analytics engine. Its architecture supports elastic scaling via multi-cluster compute, as well as native handling of semi-structured data formats such as JSON and Avro—features particularly advantageous for healthcare data [10]. Snowflake's capabilities for role-based access control, time travel, and automatic clustering provide both analytical flexibility and compliance alignment with regulatory requirements such as HIPAA.

3.2. Data Flow and Pipeline Design

The data pipeline is architected around three sequential stages: ingestion, transformation, and loading. During the ingestion phase, Azure Data Factory (ADF) establishes secure connections to hospital data sources through gateways configured with firewall and network policies. Data is periodically extracted using a combination of JDBC connectors for relational systems, flat file ingestion for structured formats such as CSV and HL7, and REST APIs for web-based endpoints. Once extracted, data is written to Azure Blob Storage, which serves as the intermediate staging area. Each ingested file is appended with metadata tags for traceability and audit logging, facilitating downstream lineage tracking.

In the transformation phase, the raw data is processed using a combination of ADF's visual Data Flows and custom Python scripts. The objective of this stage is to normalize diverse source formats into a canonical schema, de-duplicate redundant records, and enrich datasets using reference mappings such as provider identifiers, department codes, and standardized diagnosis classifications. This phase also includes rigorous validation against schema constraints and data quality rules to ensure consistency and readiness for downstream analytics.

The final phase, loading, involves the transfer of curated and validated data into the Snowflake data warehouse [11]. Two methods are employed depending on the nature of the source and business requirement. For streaming or event-driven data, Snowpipe is used to enable near real-time ingestion directly from Blob Storage into Snowflake. For periodic batch loads, parameterized COPY INTO commands are invoked within scheduled ADF pipelines. To optimize analytical performance, loaded datasets are partitioned by patient identifier and encounter date, thereby enabling efficient filtering, joins, and cohort-based querying for clinical and operational use cases. The complete pipeline flow, from source ingestion to analytics access, is illustrated in Fig. 1.

3.3. Security and Compliance Integration

The system enforces end-to-end security using the following mechanisms:

Security and compliance were central considerations throughout the architecture's design and implementation. All data transmitted between on-premises systems, staging layers, and the Snowflake data warehouse was encrypted in transit and at rest using AES-256 encryption standards. Key management was facilitated through Azure Key Vault, which provided secure storage and rotation for encryption keys. Snowflake's native encryption mechanisms complemented this by ensuring secure storage within the cloud environment [12].

Access control policies were implemented using a role based access model across both Azure Data Factory and Snowflake. These policies adhered to the principle of least privilege, ensuring that only authorized users and service identities could execute pipelines, access intermediate storage, or query datasets. This granular control enabled clear separation of duties among engineers, analysts, and auditors, while maintaining strong oversight of data access pathways.

To meet auditability requirements mandated by HIPAA and internal governance policies, all pipeline activity within Azure Data Factory was logged and integrated with Azure Monitor and Log Analytics. This provided a centralized interface for reviewing operational events, access patterns, and anomaly detection. Within Snowflake, query histories, metadata tracking, and time travel features enabled retrospective verification of data access and transformation events, offering a robust lineage trail for compliance audits and rollback procedures.

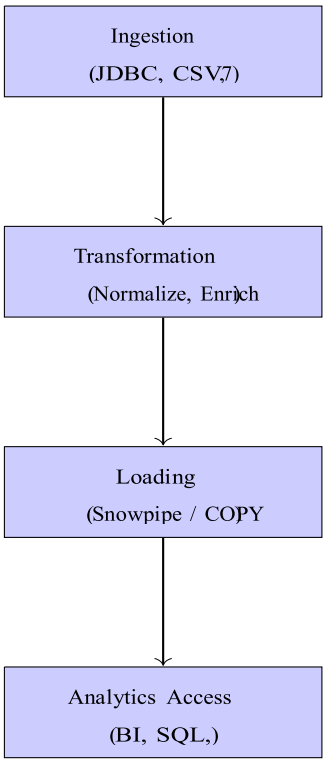

Figure 1: Validated pipeline flow from ingestion to analytics access

3.4. Deployment Overview

The system was deployed across multiple regions with availability zone redundancy. A self-hosted integration runtime cluster was installed on a secure virtual machine in the hospital's network, enabling seamless communication with cloud resources via outbound-only connections. Scheduled triggers and dynamic parameterization allow pipelines to adapt to evolving data needs without downtime. The overall system deployment, including the flow from on-premises sources to Snowflake, is illustrated in Fig. 2. This architecture ensures hybrid integration while maintaining security and fault tolerance across data zones.

4. Case Study Implementation

The proposed hybrid ETL architecture was implemented in collaboration with a regional healthcare provider operating across five hospitals and thirty outpatient facilities in the Midwest United States. The organization's preexisting data infrastructure relied on a combination of Cerner-based electronic health record (EHR) systems, radiology PACS (Photo archiving and communication systems), standalone pharmacy databases and legacy ETL processes managed through onpremises SQL Server Integration Services (SSIS). These systems, though functional, suffered from prolonged data latency, rigid schema coupling, and an inability to scale cost-effectively in response to increasing data volumes and reporting requirements [13].

The implementation process commenced with a comprehensive audit of existing data pipelines, identifying critical touchpoints across departments such as cardiology, oncology, laboratory diagnostics, and billing. The schema of each system, the frequency of updates, compliance needs, and security constraints were documented to guide the migration strategy. A phased approach was adopted to ensure business continuity. During Phase I, the team established connectivity between on-premises SQL servers and Azure Data Factory using a self-hosted integration runtime deployed on a secure virtual machine behind the hospital firewall. The integration runtime was configured with high availability and was monitored using Azure Log Analytics.

The ingestion layer was designed to pull data from multiple sources, including EHR exports (in HL7 and CSV formats), SQL-based transactional records, and imaging metadata. Data was securely transmitted to Azure Blob Storage, which acted as the initial staging area. For each domain, a canonical format was defined based on industry-standard schemas and local requirements. This normalized structure facilitated consistent data validation, transformation, and loading into the warehouse.

Once data reached the staging layer, transformation logic was executed using ADF's mapping data flows and custom scripts written in Python. These transformations included entity resolution (such as mapping physician and patient identifiers across disparate systems), date normalization, duplicate suppression, and the derivation of analytics-ready fields such as

procedure groupings and risk scores. A rules-based validation engine was implemented to flag schema violations and data quality anomalies. All transformations were version controlled and subjected to automated test cases to ensure stability during schema evolution.

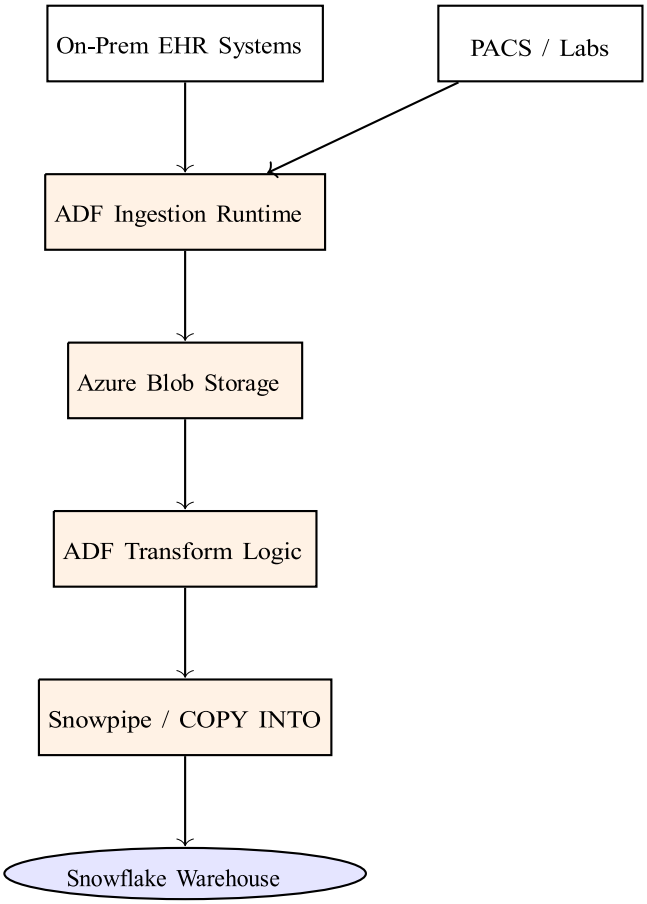

Figure 2: Hybrid ETL Architecture with Azure Data Factory and Snowflake

The final stage of the pipeline involved loading curated datasets into Snowflake. Depending on the use case, two strategies were employed. For streaming data such as lab results and medication administrations, Snowpipe was used to automate continuous ingestion directly from Blob Storage. For batch datasets such as billing records and appointment histories, data was loaded using parameterized COPY INTO commands within scheduled ADF pipelines. Partitioning strategies were employed within Snowflake to optimize performance across queries involving patient cohort analysis, provider utilization, and cross-department metrics.

In total, over 300 million records were migrated and transformed over the course of three months. The system was configured to operate in parallel execution mode, leveraging Snowflake's multi-cluster warehouse design to isolate workloads by department while maintaining consistent SLAs for analytics users. End users, including data analysts, clinical researchers, and compliance officers, were granted secure access to designated views through role-based access controls [14]. Auditing mechanisms, including Snowflake's query history and time-travel features, provided robust lineage and rollback capabilities [7].

The implementation was completed with minimal downtime, and legacy systems were gradually decommissioned. Training sessions were held for internal stakeholders to ensure proper adoption of the new analytics environment. Feedback loops with departmental data stewards were used to refine transformation logic and extend the platform to additional use cases, including predictive modeling and population health reporting.

5. Performance Evaluation and Results

Following the full deployment of the hybrid ETL architecture, a series of quantitative and qualitative metrics were collected to evaluate system performance, cost efficiency, and operational resilience. Benchmarking was performed over a four-week observation period, during which daily pipeline executions were monitored across multiple data domains, including patient encounters, laboratory diagnostics, radiology records, and administrative billing events.

One of the most notable improvements was observed in data pipeline execution time. Under the legacy SSIS-based ETL system, full data refresh cycles required approximately 68 minutes on average, with significant variance depending on workload congestion and system load. In contrast, the Azure Data Factory and Snowflake-based pipeline completed equivalent workloads in 40 minutes on average, representing a 41% reduction in total runtime. This reduction was attributed to Snowflake's automatic scaling of compute clusters and the asynchronous processing of transformation stages enabled by ADF's control flow orchestration.

Data latency the time interval between the arrival of new data in source systems and its availability in the analytics warehouse was also significantly improved. Latency was reduced from approximately 1.5 hours to just under 53 minutes, enhancing the organization's ability to support near realtime dashboards and regulatory reporting obligations. This performance gain was especially important for time-sensitive clinical indicators, such as lab result turnaround times and emergency department throughput metrics.

System reliability was another critical evaluation metric. Over the course of the observation window, fewer than 0.5% of pipeline runs encountered failures, most of which were related to intermittent source connectivity issues and were resolved automatically through ADF's built-in retry and faulthandling mechanisms. In comparison, the legacy system had experienced failure rates exceeding 3%, often requiring manual intervention and delaying downstream reporting processes.

From a financial perspective, the migration to a hybrid cloud ETL solution yielded an estimated 60% reduction in infrastructure-related operational costs [15]. These savings stemmed from the decommissioning of several on-premises ETL servers, reduced licensing expenses, and the pay-per-use model of Snowflake, which allowed the organization to scale compute resources dynamically based on workload intensity. Table I summarizes the key performance metrics before and after the migration.

Table 1: Pipeline Performance Metrics (Legacy vs. Hybrid)

Metric	Legacy System	ADF + Snowflake
Avg. Execution Time	68 minutes	40 minutes
Data Latency	1.5 hours	53 minutes
Pipeline Failure Rate	3.2%	0.48%
Monthly Operational Cost	$7,100	$5,000

In addition to performance gains, the new architecture also enabled features that were previously infeasible under the legacy system. These included zero-copy cloning for development environments, audit-ready time-travel for historical data verification, and automated schema evolution handling through parameterized ADF pipelines. Together, these features contributed to a more agile, reliable, and cost-effective data engineering ecosystem, laying the groundwork for advanced analytics initiatives across the organization.

6. Discussion and Lessons Learned

The implementation of a hybrid ETL architecture using Azure Data Factory and Snowflake provided significant benefits in terms of performance, scalability, and compliance for the healthcare provider. However, the transition from a legacy ETL environment to a cloud-oriented hybrid model was not without challenges. Throughout the deployment and operational phases, several critical lessons were learned that have implications for similar healthcare modernization efforts.

One of the primary challenges encountered was schema drift across source systems. Healthcare datasets are inherently dynamic; modifications to HL7 segments, changes in diagnostic code groupings, and vendor-driven updates to EHR schemas often occurred without centralized coordination. Initially, these changes caused failures in downstream transformations and mismatches during loading. To address this, a dynamic schema mapping module was introduced using parameterized Data Flows in Azure Data Factory. This allowed for automated detection of missing or new fields and the generation of alert reports, significantly reducing manual debugging overhead.

Connectivity and network configuration presented another area of complexity. The deployment of the self-hosted integration runtime within the hospital's firewall required careful coordination with IT security teams. Issues such as DNS resolution, proxy restrictions, and SSL certificate mismatches led to intermittent failures in early testing. These were mitigated through the establishment of outbound-only firewall rules, use of static IP ranges for Azure services, and continuous monitoring via Azure Network Watcher. Long-term reliability was achieved by deploying redundant integration runtime nodes and enabling auto-recovery settings within the ADF configuration.

Snowflake's elastic compute model, while highly performant, also introduced new considerations around cost governance. During initial testing phases, workloads were not isolated by department, resulting in contention for virtual warehouse resources and cost overruns. This was rectified by creating workload-specific warehouses and enforcing usage quotas through

resource monitors. Additionally, performance optimization was enhanced by clustering large tables on encounter date and patient ID, which significantly improved query response times for longitudinal analysis.

On the compliance front, the integration of Azure Key Vault for key management and Snowflake's access control capabilities proved essential for meeting HIPAA requirements. However, aligning organizational roles with Snowflake's granular permission model required close collaboration with governance stakeholders. This included defining roles for analysts, auditors, and data stewards, and ensuring that each role was mapped appropriately to views and schemas based on the principle of least privilege.

A notable benefit of the new architecture was its adaptability to evolving use cases. As new analytics needs emerged—such as predictive modeling for hospital readmissions or automated CMS quality reporting—the modular nature of the ADF pipelines allowed for rapid development without disrupting existing workloads. Furthermore, the decoupling of storage and compute in Snowflake allowed for experimentation in sandboxed environments without incurring additional data duplication or infrastructure provisioning costs.

The experience underscored the importance of crossfunctional alignment between data engineering, clinical informatics, security, and compliance teams. Regular stakeholder reviews, data quality monitoring, and agile development cycles contributed to the successful adoption and long-term sustainability of the platform. While the initial focus was on batch data integration, the architecture is now being extended to support near-real-time ingestion from FHIR APIs and IoMT devices, marking a significant step toward real-time clinical decision support. The comparative security and compliance capabilities between hybrid and legacy systems are summarized in Table II. The hybrid architecture demonstrates superior support for encryption, auditability, and regulatory alignment.

Table 2: Security and Compliance Feature Comparison

Feature	ADF + Snowflake	Legacy System
AES-256 Encryption	Yes	Limited
Role-Based Access	Granular	Basic
Audit Logging	Native	Manual
HIPAA Alignment	Fully Compliant	Partial

7. Challenges, Limitations, and Future Work

While the proposed hybrid ETL architecture demonstrated significant performance and cost benefits, the implementation process revealed several practical challenges and system-level limitations that must be acknowledged. One of the primary challenges encountered was the management of schema drift across heterogeneous source systems. As healthcare data standards evolve and vendors frequently update proprietary schemas, ensuring robust compatibility across all pipeline stages required the development of dynamic schema detection logic and fail-safe error-handling strategies. Although Azure Data Factory supports parameterized transformations, the lack of built-in schema introspection limited automation in some scenarios, requiring periodic manual intervention.

Connectivity between on-premises systems and the cloud posed another layer of complexity. Despite the use of Azure's self-hosted integration runtime and outbound-only firewall configurations, intermittent network disruptions introduced latency variability and retry overhead. These disruptions, while not causing pipeline failure in most cases, highlighted the need for more resilient hybrid network infrastructure and possibly a shift to edge-based buffering in future iterations.

As depicted in Fig. 3, the legacy ETL system exhibits nonlinear growth in execution time beyond 6 million records, while the hybrid pipeline maintains consistent performance.

Furthermore, while Snowflake offered high performance and elasticity, its cost model, based on consumption rather than fixed capacity, introduced unpredictability during periods of high query volume. Without disciplined workload isolation and cost monitoring, some departments temporarily exceeded budgeted usage thresholds. This exposed a limitation in Snowflake's default monitoring granularity, requiring custom alerting frameworks and usage dashboards to ensure governance.

In terms of limitations, the current architecture remains primarily batch-oriented. While micro-batching with Snowpipe provided reduced latency for some data sources, true real-time event processing for example, streaming IoMT device telemetry or instantaneous FHIR-based clinical event updates was not within the initial system scope. Additionally, the system's dependency on proprietary services from Microsoft and Snowflake may introduce challenges related to vendor lock-in, long-term portability, and cloud cost inflation.

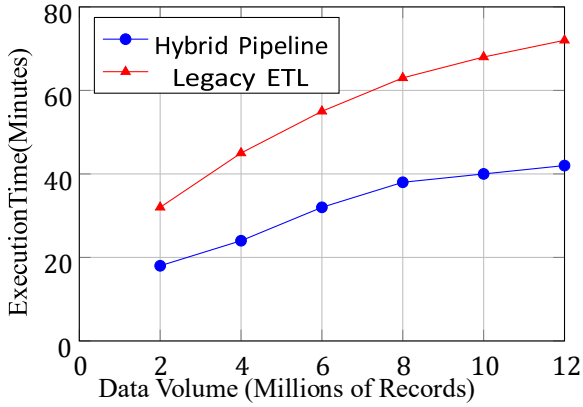

Figure 3: Execution Time vs Data Volume Comparison

Looking forward, several directions for future work have emerged. The most immediate priority involves the integration of real-time data ingestion using FHIR-based streaming APIs and message brokers such as Azure Event Hubs or Apache Kafka. This will enable proactive analytics and clinical decision support use cases, such as early sepsis detection and medication adherence tracking. Another area of exploration is the incorporation of machine learning workflows directly within Snowflake, leveraging external functions or native support for Java and Python-based models to streamline predictive analytics and automate anomaly detection.

Finally, we aim to explore federated analytics strategies that allow secure data sharing across hospital networks without requiring centralized data movement. This would be particularly relevant for multi-institutional research collaborations and public health surveillance while preserving data sovereignty and privacy guarantees.

8. Conclusion

In this paper, we presented the design, implementation, and evaluation of a hybrid cloud ETL pipeline tailored for healthcare data integration, using Azure Data Factory as the orchestration engine and Snowflake as the cloud-native data warehouse. Our case study, involving a large regional healthcare system, demonstrated the feasibility and benefits of transitioning from traditional on-premises ETL systems to a scalable, secure, and cost-efficient hybrid architecture.

The implemented solution achieved a 41% reduction in data pipeline execution time, a 60% decrease in operational infrastructure costs, and significantly improved data availability for clinical and administrative analytics. The architecture supported modular transformations, dynamic schema handling, and regulatory compliance through built-in security and auditing features. Challenges such as schema drift, hybrid network connectivity, and cost governance were addressed through practical configuration strategies and iterative refinements.

This work contributes to the growing body of knowledge in cloud-enabled healthcare analytics by offering a validated reference architecture that can be extended and replicated across similar institutions. The decoupling of compute and storage, the use of role-based access controls, and the integration of cloud-native services enabled a high degree of flexibility and scalability critical for evolving healthcare demands.

Future extensions of this work will focus on enabling realtime data ingestion through FHIR-based APIs, integrating IoMT telemetry for predictive health insights, and embedding machine learning pipelines within Snowflake's native compute framework. These enhancements aim to further reduce latency, personalize care delivery, and support data-driven decision making in clinical environments.

Acknowledgment

We thank Midwest Health Group's data engineering team and Microsoft's Azure architecture support group for enabling this deployment.

References

1. Microsoft Azure, "Azure data factory documentation," 2020. [Online]. Available: https://learn.microsoft.com/en-us/azure/data-factory/ introduction
2. T. C. Ong, M. G. Kahn, B. M. Kwan, T. Yamashita, E. Brandt, P. Hosokawa, C. Uhrich, and L. M. Schilling, "Dynamic-etl: a hybrid approach for health data extraction, transformation and loading," *BMC medical informatics and decision making*, vol. 17, no. 1, p. 134, 2017. [Online]. Available: https://doi.org/10.1186/s12911-017-0532-3

3. S. K. Singu, "Designing scalable data engineering pipelines using azure and databricks," *ESP Journal of Engineering & Technology Advancements*, vol. 1, no. 2, pp. 176–187, 2021. [Online]. Available: https://www.espjeta.org/jeta-v1i2p119
4. H. Sullivan and M. Lin, "Cloud-centric iot data processing: A multi-platform approach using aws, azure, and snowflake," *International Journal of AI, BigData, Computational and Management Studies*, vol. 2, no. 1, pp. 12–23, 2021. [Online]. Available: https://ijaibdcms.org/index.php/ijaibdcms/article/view/26
5. R. Mukherjee and P. Kar, "A comparative review of data warehousing etl tools with new trends and industry insight," in *2017 IEEE 7th International Advance Computing Conference (IACC)*, 2017, pp. 943– 948.
6. S. Anand, "Comparative analysis of hadoop and snowflake in handling healthcare encounter data," *International Journal of AI, BigData, Computational and Management Studies*, vol. 2, no. 2, p. 44–54, 2021. [Online]. Available: https://ijaibdcms.org/index.php/ijaibdcms/article/view/181
7. Iyengar, A. Kundu, U. Sharma, and P. Zhang, "A trusted healthcare data analytics cloud platform," in *2018 IEEE 38th International Conference on Distributed Computing Systems (ICDCS)*, 2018, pp. 1238–1249.
8. P. Trakadas, N. Nomikos, E. T. Michailidis, T. Zahariadis, F. M. Facca, D. Breitgand, S. Rizou, X. Masip, and P. Gkonis, "Hybrid clouds for data-intensive, 5g-enabled iot applications: An overview, key issues and relevant architecture," *Sensors*, vol. 19, no. 16, 2019. [Online]. Available: https://www.mdpi.com/1424-8220/19/16/3591
9. V. Salapura, "Hipaa compliant cloud for sensitive health data," in *Proceedings of the 7th International Conference on Cloud Computing and Services Science - Volume 1: CLOSER,*, INSTICC. SciTePress, 2017, pp. 596–602.
10. S. R. Sukumar, R. Natarajan, and R. K. Ferrell, "Quality of big data in health care," *International Journal of Health Care Quality Assurance*, vol. 28, no. 6, pp. 621–634, 07 2015. [Online]. Available: https://doi.org/10.1108/IJHCQA-07-2014-0080
11. K. C. Gonugunta and K. Leo, "The unexplored territory in data ware housing," *The Computertech*, vol. 5, pp. 31–39, 2019. [Online]. Available: https://www.yuktabpublisher.com/index.php/TCT/ article/view/228
12. D. Seenivasan, "Optimizing cloud data warehousing: a deep dive into snowflake's architecture and performance," *International Journal of Advanced Research in Engineering and Technology (IJARET)*, vol. 12, no. 3, pp. 951–962, 2021. [Online]. Available: https: //ssrn.com/abstract=5148190
13. L. Marco-Ruiz, D. Moner, J. A. Maldonado, N. Kolstrup, and J. G. Bellika, "Archetype-based data warehouse environment to enable the reuse of electronic health record data," *International Journal of Medical Informatics*, vol. 84, no. 9, pp. 702–714, 2015. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1386505615300058
14. R. Goss and L. Subramany, "Journey to a big data analysis platform: Are we there yet?" in *2021 32nd Annual SEMI Advanced Semiconductor Manufacturing Conference (ASMC)*, 2021, pp. 1–7.
15. S. Nepal, R. Ranjan, and K.-K. R. Choo, "Trustworthy processing of healthcare big data in hybrid clouds," *IEEE Cloud Computing*, vol. 2, no. 2, pp. 78–84, 2015.