From Legacy Systems to Scalable Cloud Platforms: Building Modern Data Pipelines with Data Engineering Strategies, Scaling Trust, Compliance, and Performance in Public Health

Mani Kanta Pothuri

Manikantpothuri1@gmail.com

Abstract:
Public health data motivates impactful decisions. Healthcare centers often process data using legacy systems. These are monolithic with silos and lack flexibility. Latency in responses is a common side effect of these models. Shifting into scalable cloud-supported data platforms allows real-time analytics and increases trust by ascertaining regulatory compliance, thereby increasing performance. The paper studies key data engineering strategies such as modular ETL dataflows. Robust data modeling, metadata-supported design, automatic quality manifestation, and governance architectures. The functionalities are proposed to modernize citizen health data modeling and management frameworks. This study outlines regarding use of cloud native tools and architecture trends to support delivering scalable and resilient outcomes. Meticulous data privacy management, security, and uptime needs are addressed. Finally, strategies to develop stakeholder trust and regulatory compliance are discussed in dynamic data environments.

Keywords: Legacy systems, ETL dataflows, Scalable cloud, Governance architectures, Dynamic data environments.

1. INTRODUCTION

Healthcare systems across the world generate huge volumes of diversified datasets in the form of electronic health records, vaccination data, surveillance feeds about syndromes, demographic registries, and genomic sequencing information. Conventionally, these datasets are overseen in secluded legacy systems using on-premise mainframes, proprietary content servers, and batch ETL (Extracting, Transforming, and Loading) processes [1]. These systems are limited to data scalability, integration fragility, and the absence of real-time data processing capabilities. The challenges to adapt to emerging compliance demands like GDPR, HIPAA, and other local data regulations also occur with traditional processing frameworks.

Transition from rigid legacy architecture to flexible cloud platforms is beyond a simple technological upgrade. This is a shift in user mindset. Citizen health organizations need to evolve from basic infrastructures to cloud-native ecosystems for addressing dynamic data demands and policy enforcement changes [2]. Using a modular pipeline pattern, metadata-motivated quality control, and strong governance standards, the scalable cloud architecture capably delivers responsive outcomes while managing trust, compliance, and performance requirements.

2. CHALLENGES OF LEGACY SYSTEMS

2.1 Public health regulations

Managing public health data requires stringently aligning with legal and ethical guidelines. These regulations are developed to secure patient privacy and ensure security.
HIPAA: This is the basic law that governs the utilization and disclosure of secure health information (PHI [3]). This is based on the privacy rule yay establishes national standards to secure PHI. Security rule sets up

standards for securing PHI. The breach notification rule requires entities to share notices to affected users and the government for managing data breach incidents.

GDPR: GDPR has been formulated by the European Union to manage data privacy. Although not exclusively developed for health data, this regulation has a major influence on public health initiatives. Personal data processing by entities needs to be legal, fair, and transparent. Data minimization is another principle that involves the collection of absolutely required data only. Personal data needs to be redacted after completion of the purpose and erased from associated systems.

Internal regulations: Different nations implement exclusive data regulations to protect personal information, and the Electronic Documents Act (PIPEDA) is enforced in Canada and the Australian Privacy Act. Healthcare systems need to follow such international standards for compliance with related guidelines.

2.2 Issues with legacy system features

Silos in rigid infrastructure: Legacy systems store data in different blocks according to the exclusive service and programs addressed by specific data, such as vaccination tracking, case reports, and pandemic statistics [4]. Integration of this data requires customized interfaces and manual exporting. Such rigidity delays data access and impedes cross-functional analytics. These issues inhibit support for evolving applications such as pandemic outbreak detection and tracking.

Scalability and performance impasse: On-premise data servers are limited to their capability and buckle under continuous data volume requirements, such as a surge in epidemic cases [5]. Batch-wise ETL data pipelines are processed once a day or in a week. Public health management needs continuous and real-time responses.

Trust, privacy, and compliance issues: Data involves sensitive health content that needs to be restricted with access regulations, auditability, and compliance with regulatory guidelines like HIPAA (Health Information Portability and Accountability Act, GDPR (General Data Protection Regulations), and other local guidelines [6]. Legacy time logs and monitoring have limited traceability. This increases difficulties in demonstrating compliance with the auditing board and responding to data breach incident investigations.

3. STRATEGIC CLOUD ARCHITECTURE APPLICATION

3.1 Type of cloud platform

Selecting a suitable advanced cloud platform such as AWS, Azure, or Google Cloud allows implementing managed services for computing, storage, and orchestrating data. The managed services decrease operational overages and support with inherent scalability. AWS is effective for managing Kinesis, Redshift, Glue, EMR, and lake creation. GCP is effective for dataflow, Big Query, Cloud SQL, Dataproc, and Dataplex. Azure is also efficient by supporting Event Hubs, Data Factory, and Synapse Analytics [7]. Migration of data and processes into managed services empowers healthcare agencies to emphasize compliance, trust, and domain-based logic rather than installing patches in the infrastructure or planning data storage capacity.

3.2Core architectural trends

Event-based ingestion is promoted using data streaming platforms like Kinesis gather data from source systems like EHRs as events depicting test results and case scenarios.

Landing and structured zone secures direct event logs with immutability using time stamps and landing zones indicating the cloud object storage versions [8]. The Upstream structured conversions sanitize, remove duplicates, and normalize like tiered data zones.

Data Lakehouse secures content into usable formats with schematic catalogues and optimized queries [9]. Lakehouse empowers interactive SQL and batch-wise processes of data streams.

Serving layer curates data products such as tables, data marts, Application program interfaces, and learning features. The elements are exposed by manifesting access regulations through access-regulated query engines such as Redshift Secure views.

Centralized data event logs promote traceability, data lineage metadata, quality dashboards, and detection of anomalies are encapsulated across insertion and transformation levels.

The stages involved in data processing are depicted in following Figure 1.

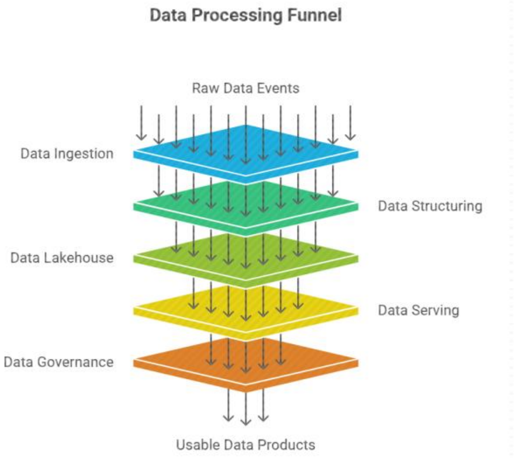

Figure 1 Data processing funnel

4. DATA ENGINEERING STRATEGIES OR ADVANCED DATA PIPELINES

Modern data engineering is beyond the simple mobilization of data. This involves developing strong, scalable, and reliable systems [10]. These are effective for managing huge volumes of data using the following key strategies.

4.1 Modular ETL versus ELT pipelines

The conventional approach is to extract the data, followed by transforming and loading. This involves cleaning and transforming data before loading it into warehouses. An advanced strategy in this regard is extraction and loading, followed by transformation. This model is effective for loading raw data directly into cloud-oriented lakes, and conversions occur later. ELT is very pliant and enables a schema-on-read process, which is about deciding the structure of data while conducting analysis [10]. Version-based data pipelines supporting Continuous integration and deployment (CI/CD) practices ensure consistent data flow across development, staging, and management. These pipelines emit logs while running with metrics and lineage elements such as row counts, error levels, and drift in schema.

Schema-on-read and Intent-Based Design: Implementing the schema-on-read format for raw data is effective in expediting the ingestion process. This imposes schema limitations in the conversion of structured zones [11]. Transforming express intent involves implementing valid results. Using tools such as Apache Iceberg, Delta Lake, and BigQuery's implementation supports emergence and versioning. The schema drift would be identified, escalated after adding an entry.

Metadata Pipelines: These allow maintaining centralized metadata catalogs such as Glue data, Hive, or Dataplex. Automated dataflows capture metadata for discovering new sources, creating schema definitions, and manifesting standards such as naming consistencies. These allow propagation of data by motivating artifacts such as documentation [12]. This data is also associated with regulatory compliance and tagging using sensitivity levels, retention policies, and governance categorization.

Automatic data quality and validity: Adding rules while processing data pipelines is important for checking null values, validating ranges, and managing referential integrity. Data anomalies and schema are also automatically detected. Any violations in these rules, re-route the records into a constrained zone with visuals for analysis and reconciliation [13]. Tools such as deequ or Soto are effective by providing frameworks to define and implement quality checks.

Privacy engineering and access control: Sensitive datasets are segregated, manifesting access through granular-level policies. These could be sensitive personal identifiers that could be tokenized or hashed while ingesting [14]. Pseudonymized identities only go downstream. The access permissions that are de-identified, and an integrated view appears for analysis. The protected health records (PHI) are limited to authorization. Logs allow tracking of access data and activities. Encryption is applicable while data is at rest as well as in transit using key service management.

4.2 Batch processing versus real-time processing

Batch processing includes processing a huge volume of content following a schedule. This is effective for processes to generate timely data reports and execute high-level analytics. Apache Spark is the most used tool for this process.

Stream processing activities include processing content continuously. This is important for applications to generate immediate perceptions, like monitoring patient information and tracking epidemic outbreaks [14]. Technologies such as Apache Kafka and Flink are regularly implemented for stream processing.

4.3 Data governance and quality

An Advanced data pipeline needs to incorporate methods for managing data governance and quality.

- Data Lineage is about tracking the origin and transit path of data from source to the endpoint.
- Data validation involves executing validation checks to ascertain completeness, accuracy, and consistency.
- Master data management includes developing one authorized source for key data like patient identification details and facility codes.
- Scalability and automation are useful for developing advanced data pipelines that are developed for continuous scalability [15]. The processes include the following tools.

Containerization: Implementing Docker or Kubernetes technologies for packaging applications and dependencies is known as containerization. This increases the portability and scalability of cloud data.

Cloud native services: Leveraging managed facilities using cloud service vendors, using AWS Glue, Google Dataflow, and Azure Data Factory for automated management, scaling, and resource handling.

Orchestration: Implementing tools such as Apache Airflow or Perfect to schedule and monitor supports in managing complicated data workflows.

4.4 Cloud platforms for public health

Cloud domains such as AWS, Google, and Microsoft Azure support with tools and architecture required for developing advanced data systems depicted in **Figure 2.**

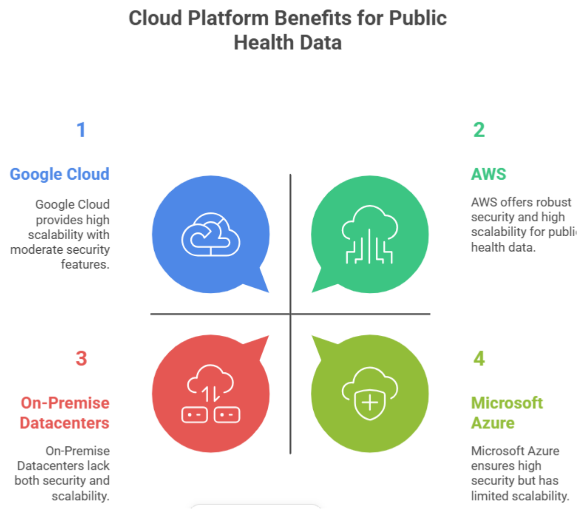
Cloud Platform Benefits for Public Health Data
1
Google Cloud
Google Cloud provides high scalability with moderate security features.
2
AWS
AWS offers robust security and high scalability for publi health data.
3
On-Premise Datacenters
On-Premise Datacenters lack both security and scalability.
4
Microsoft Azure
Microsoft Azure ensures high security but has limited scalability.

Figure 2 Cloud platforms for public health

Security and compliance: Cloud service vendors support with a strong set of security features and work with certification for different compliance-based standards such as HIPAA and GDPR [4]. These support the pursuit of developing a secure and compliant system seamlessly. Facilities such as AWS or Azure support with granular access regulations to secure sensitive content.

Scalability and performance: Cloud services are effective for unlimited computing and storage of resources according to user needs. These are important for managing population health in case of a surge in data volumes due to epidemic occurrences or vaccinations conducted on a huge scale [16]. Cloud support offered by Amazon S3 or Azure Synapse Analytics is developed to manage huge datasets with maximum performance.

Cost efficiency: Implementing models for payments as per usage is effective to manage costs rather than installing architecture and managing on-premise datacenters. These enable public health support or pursuits to assign resources effectively.

Innovation and collaboration: Cloud support allows accessibility to advanced technologies such as Machine Learning, Artificial Intelligence, and advanced analytics [17]. These allow public health institutions and practitioners to efficiently promote population health by delving deeper into the data available and achieving better results.

5. DATA GOVERNANCE AND REGULATORY COMPLIANCE

Using Metadata tagging ascertains about dataset association with rules such as retention period, allowed data areas, and anonymization stages. The following elements are important for effective data governance and complying with regulations.

- The data pipelines manifest tag awareness about data above the retention period for purging or archival.
- Data transformation is associated with lineage involving versions, triggers according to a user schedule, inputs, and schemas [18]. These are critical for managing compliance following auditable pipelines.
- Continuous penetration tests, encryption key changes, access evaluations, and SIEM monitoring ensure that the database follows standards.
- Public and internal dashboards could be used to display metrics regarding data quality, accuracy, and pipeline execution success, along with compliance elements.
- Differential privacy gateways and query restrictions could be used to generate aggregated insights into individual record protection.
- Implementing shared architecture provides an overview regarding compliance reports and risk evaluation, as institutional confidence has been fostered among stakeholders.

6. PERFORMANCE AND SCALABILITY FRAMEWORK FOR HEALTHCARE DATA

Figure 3. depicts the proposed framework for high performing and scalable healthcare data operations and governance by shifting from legacy systems.

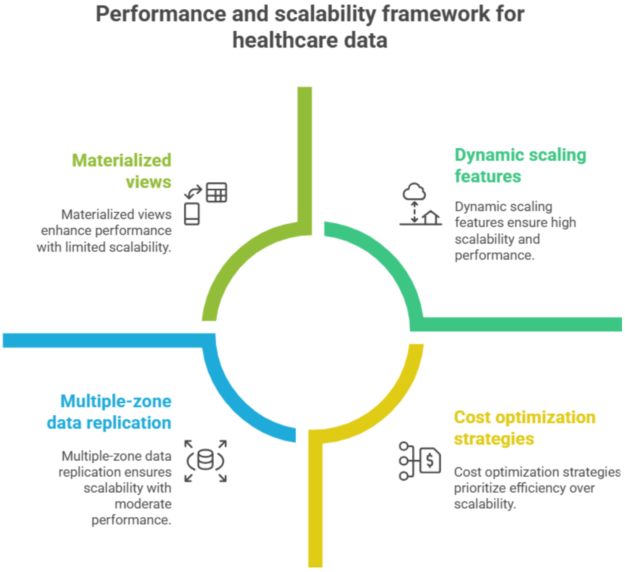
Performance and scalability framework for healthcare data
Materialized views
Materialized views enhance performance with limited scalability.
Dynamic scaling features
Dynamic scaling features ensure high scalability and performance.
Multiple-zone data replication
Multiple-zone data replication ensures scalability with moderate performance.
Cost optimization strategies
Cost optimization strategies prioritize efficiency over scalability.

Figure 3 Performance and scalability framework for healthcare data

- Cloud platforms are equipped with dynamic scaling features and serverless query engines.
- The elastic computer clusters are effective for managing streaming facilities and adjusting capabilities to address demand.
- Optimizing costs involves intelligently segregating the database, cache query outcome, and automatic idle resource shutdown [18].

- Frequently executed datasets and pre-unified tables develop materialized views by delivering fewer latent queries.
- Cloud service providers such as AWS SageMaker support ML processes, ascertaining consistency in feature values across training and serving processes.
- Implementing multiple-zone data replication and cross-region support, addressing datacenter outages.
- The Pipeline orchestration tools allow automating failover and scheduling rehearsals, ascertaining that recovery plans are implemented efficiently.

6.1 Real-time healthcare incident monitoring instance

Public health agencies are involved with tracking diverse events after vaccination campaigns using continuous data streams from clinics and diagnostic centers [10]. Implementing cloud-based architecture involves the following elements.

- Data ingestion happens with event logging through Pub/Sub or Kinesis. The raw events are in a version-based object store.
- Payloads are validated by normalizing codes, such as adverse event tables.
- This follows the deduplication of patient information with privacy mapping and segregation according to severity.
- Query schema drift allows detecting new incident types and triggers alerts schema updates.
- De-identified and collaborated daily cases are analyzed using sliding-window dashboards.
- Feature store supports learning models for high-risk profiling and follow-up [19]. All the information accessed is logged, and the compliance dashboard shares notifications to data analysts regarding privacy window expirations.

7. FUTURE DIRECTIONS

- Deploying adaptive governance models is effective in keeping up with changing regulations.
- Connecting with IoT and wearable health devices to improve time-series healthcare data involves storage, processing, and analysis [4].
- Ensuring data privacy and following ethical standards by using federated learning, differential privacy, and synthetic data [20].
- The process supports running analytics without needing centralized data.
- Applying post-quantum cryptography to secure healthcare data.
- Collaborating with automated intrusion detection and response systems for stronger protection [13].
- Regularly reviewing frameworks against compliance standards to reduce bias and support fairness in healthcare data pipelines.
- Focusing on sustainability when designing cloud-native platforms for managing healthcare data.

8. CONCLUSION

Public healthcare systems and organizations require implementing advanced techniques for tracking data in alignment with policies as an ongoing process. Migration of data into the cloud empowers interoperability and security. Building reliable and efficient data pipelines is a prerequisite for healthcare data management systems. Cloud platforms can oversee a wide range of data loads, ensuring systems remain stable and available even during unexpected events. Integrated, high-level data processing becomes more accessible, empowering practitioners and researchers. Successfully modernizing these systems goes beyond simply updating technology. Establishing clear rules and oversight for data management fosters dependability and security. In conclusion, an effectively planned approach to security and public trust is vital for successful implementation. Initiative-taking platform development and integration with AI and supervised learning increases the efficiency of public health data.

REFERENCES:

[1] A. Aljaloud and A. Razzaq, "Modernizing the Legacy Healthcare System to Decentralize Platform Using Blockchain Technology," MDPI Technologies, vol. 11, no. 4, pp. 1-17, 2023.

[2] O. Alkhubouli, H. M. Lala, A. A. AlHabshy and K. A. ElDahshan, "Enhancing data warehouses security," International Journal of Advanced Computer Science and Applications (IJACSA), vol. 15, no. 3, pp. 1-23, 2024.

[3] B. Althani, "Migration challenges of legacy software to the cloud: a socio-technical perspective," Information & Technology Management, vol. 12, no. 2025, pp. 1-10, 2024.

[4] G. S. Bolla, "Integration Architecture Fundamentals for Healthcare Systems: A Framework for Seamless Interoperability," Journal of Computer Science and Technology Studies, vol. 7, no. 3, pp. 106-111, 2025.

[5] A. Bönisch, D. Kesztyüs and T. I. Kesztyüs, "FAIR+R: Making clinical data reliable through qualitative metadata," Studies in Health Technology and Informatics, vol. 310, no. 1, p. 99–103, 2024.

[6] N. Chennupati, "Zero-touch transformation: AI-driven middleware for autonomous integration of," World Journal of Advanced Engineering Technology and Sciences, vol. 15, no. 2, pp. 1444-1453, 2025.

[7] M. Bregonzio, A. Bernasconi and P. Pinoli, "Advancing healthcare through data: the BETTER project's vision for distributed analytics," Front. Med.,, vol. 11, no. 1, pp. 1-10, 2024.

[8] N. Bugshan, I. Khalil, A. P. Kalapaaking and M. Atiquzzaman, "Intrusion Detection-Based Ensemble Learning and Microservices for Zero Touch Networks," IEEE Communications Magazine, vol. 61, no. 6, pp. 86 - 92, 2024.

[9] V. G. Yogeshappa, "Designing Cloud-Native Data Platforms for Scalable HealthcareAnalytics," International Journal of Research Publication and Reviews, vol. 6, no. 3, pp. 3784-3791, 2025.

[10] Putzier, Khakzad, Dreischarf, Thun, Trautwein and Taheri, "Implementation of cloud computing in the German healthcare system," npj Digital Medicine, vol. 7, no. 12, pp. 1-10, 2024.

[11] S. Chundru and P. K. Maroju, "Architecting Scalable Data Pipelines for Big Data: A Data Engineering Perspective," International Journal of Intelligent Systems and Applications in Engineering, vol. 12, no. 23S, p. 1855–1870, 2024.

[12] S. A. Ionescu, V. Diaconita and A. O. Radu, "Engineering Sustainable Data Architectures for Modern Financial Institutions," Electronics, vol. 14, no. 8, pp. 1-15, 2024.

[13] M. Kalkatawi, "Beyond the upgrade: unraveling the complexities of health information system migration," Discover Health Systems, vol. 4, no. 7, pp. 1-10, 2025.

[14] E. Dritsas and M. Trigka, "A Survey on the Applications of Cloud Computing in the Industrial Internet of Things," Big Data And Cognitive Computing, vol. 9, no. 2, pp. 1-14, 2025.

[15] K. Gierend, S. Freiesleben, D. Kadioglu, F. Siegel, T. Ganslandt and D. Waltemath, "The Status of Data Management Practices Across German Medical Data Integration Centers: Mixed Methods Study," JMIR Publications, vol. 25, no. 1, pp. 1-32, 2023.

[16] A. Islam, H. Karimipour and T. R. Gadekallu, "An Explainable AutoML-Driven Meta-Learning Scheme for Intrusion Prevention in Zero-Touch Networks Within Carbon Intelligent IIoT," IEEE Internet of Things Journal, vol. 1, no. 1, pp. 1-22, 2025.

[17] D. D. Jayaram, "Bridging Legacy Systems with Modern Platforms: A Scalable Approach," International Journal of Research in Computer Applications and Information, vol. 8, no. 1, pp. 3192-3210, 2025.

[18] P. Yanamadala, "Demystifying cloud-native enterprise architecture: A framework for digital transformation in complex organizations," World Journal of Advanced Research and Reviews, vol. 26, no. 1, pp. 1919-1928, 2025.

[19] O. Ogunwole, E. C. Onukwulu, M. O. Joel, E. M. Adaga and A. I. Ibeh, "Modernizing Legacy Systems: A Scalable Approach to Next-Generation Data Architectures and Seamless Integration," International Journal of Multidisciplinary Research and Growth Evaluation, vol. 4, no. 1, pp. 901-909, 2025.

[20] S. Singh, B. Pankaj, Nagarajan and N. P. Singh, "Blockchain with cloud for handling healthcare data: A privacy-friendly platform," Materials Today: Proceedings, vol. 62, no. 7, pp. 5021-5026, 2022.