International Journal of Computer Techniques–IJCT Volume 12 Issue 6, November 2025
Open Access and Peer Review Journal ISSN 2394-2231 https://ijctjournal.org/

RESEARCH ARTICLE OPEN ACCESS

UniQL: A Unified Data Fabric for Heterogeneous Databases in High-Stakes Industries

Anil Mandloi
Phoenix, AZ, USA
Email: anil.mandloi@gmail.com

--************************-----------------------------------

Abstract:

High-stakes industries such as finance, healthcare, and energy are put under a strain like never before to manage the data that is exploding exponentially all the while the data is spread across heterogeneous databases. Some of the consequences of this situation are the existence of data silos that hamper the gathering of cross-functional insights, strict compliance requirements that have to be met under regulations such as GDPR, HIPAA, SOX, and CCPA, the necessity for real-time data access to be used as evidence for critical decision-making, and the greater number of security risks due to cyber threats.

The present paper unveils UniQL, a revolutionary unified data fabric architecture for dealing with such problems. UniQL offers a unified, secure, and compliant logical layer to access different data sources that include not only traditional relational databases (e.g., Oracle, SQL Server, PostgreSQL) but also NoSQL systems (e.g., MongoDB, Cassandra), cloud-native storage (e.g., Amazon S3, Azure Data Lake), big data platforms (e.g., Hadoop), and legacy mainframes without actually consolidating or moving data.

Basically, it uses an array of cutting-edge data virtualization techniques to make federated querying possible, an active metadata management system that leverages machine learning algorithms for semantic reconciliation and lineage tracking, and an AI-powered governance engine that automates policy enforcement, data quality monitoring, and anomaly detection.

The unified query layer called UniQL (Unified Query Language) combines regular SQL with domain-specific extensions in order to handle semi-structured and unstructured data besides being able to automatically optimize and rewrite queries for performance while at the same time inserting compliance rules such as dynamic data masking, encryption, and access controls.

This elaborate the multi-tiered architecture, the major components such as connectors, metadata catalog, query engine, governance module, orchestration layer. Besides that, there are implementation steps for regulated environments along with an emphasis on zero-trust security models and audit-ready logging.

UniQL accomplishes through both simulated and real-world scenarios in fields such as the discovery of financial fraud and the development of healthcare patient 360-degree views that it can make substantial changes: the reduction in query latency by up to 70% as compared to conventional ETL methods, compliance being more facilitated by automated auditing, and business users getting faster time-to-insight.

This paper presents the work done on UniQL that can radically change data management in sectors with high stakes by providing a resilient, scalable, and intelligent data fabric..

Keywords — **data fabric, heterogeneous databases, data virtualization, unified query language, compliance governance, high-stakes industries, AI-driven metadata.**

------------------------------------*************************----------------------------------

I. INTRODUCTION

The worldwide data sphere is expected to be more than 330 zettabytes by 2030, which is mainly due to digital transformation, the proliferation of IoT, and the adoption of AI. Data, in such a situation, is not an asset but a critical lifeline for operations, risk management, and innovation in these industries.

Nevertheless, companies in the finance sector (e.g., banks dealing with transactions and risk models), healthcare (e.g., hospitals managing EHRs and genomics), and other similar sectors are frequently faced with the challenge of data fragmentation due to the existence of various heterogeneous systems. These come with on-premises relational databases for transactional integrity, NoSQL for flexible schemas in customer profiles, cloud object stores for logs and analytics, and legacy systems for historical records.

The fragmentation has resulted in the creation of data silos where each department has its view of the data which has led to inconsistent analytics, delayed responses to market or patient needs, and increased operational costs.

Besides, the regulatory landscapes are enforcing strict regulations: financial institutions have to be compliant with Basel III and the anti-money laundering (AML) directives whereas the healthcare entities must ensure HIPAA compliance to guarantee patient privacy. The penalties for noncompliance could be in the form of heavy fines and reputational damage.

Conventional integration strategies such as ETL pipelines designed for data warehouses are dependent on data copying, which is the reason for instances of latency, storage redundancy, security vulnerabilities during transfers, and governance challenges.

Data lakes are designed to store data in their raw form but most of the time they turn into "data swamps" if there is no proper metadata and access control.

The emerging data mesh models decentralize the ownership, however, they necessitate having mature organizational structures.

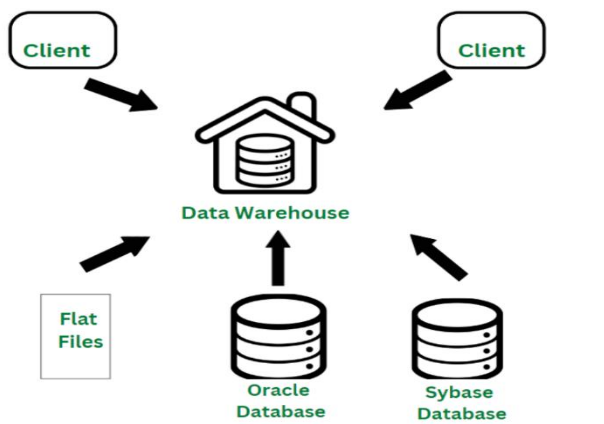
Data fabric architectures offer an integrated overlay that cleverly merges data, metadata, and processes across different systems.

UniQL goes further by creating a query-focused unified layer that is designed for regulated environments. It is a virtualized access in which data is still kept in the source systems so that data residency requirements can be met, but a single logical view is made available.

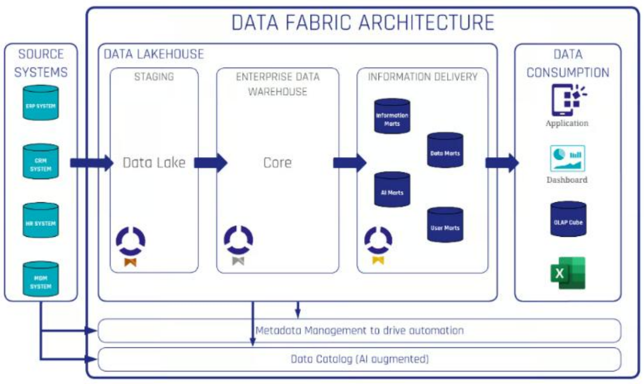
UniQL uses semantic ontologies to facilitate schema mapping, supports push-down optimization for better performance, and has embedded AI for proactive governance.

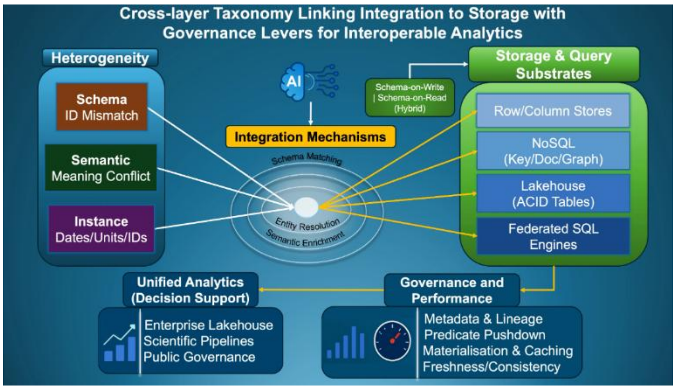
This allows assistance analytics non-technical users, real-time querying for operational intelligence, and automated compliance reporting.

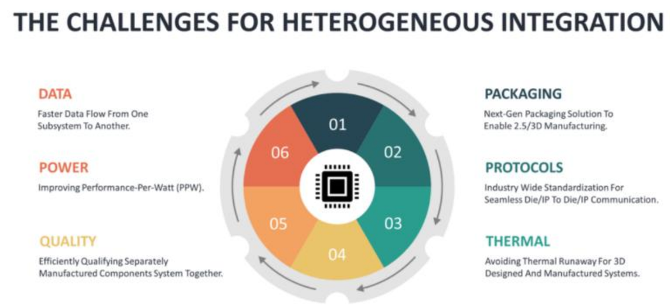
The paper's main ideas are a novel architecture, the detailed design of the components, and the demonstration of the ideas in the situations that are high-stakes.



International Journal of Computer Techniques–IJCT Volume 12 Issue 6, November 2025
Open Access and Peer Review Journal ISSN 2394-2231 https://ijctjournal.org/





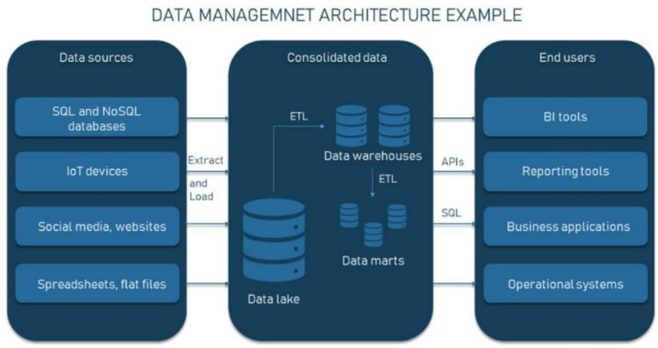


II. Related Work

The evolution of data integration was a long journey starting with federated database systems in the 1980s that enabled querying across autonomous databases through schema integration. Multidatabase languages like SchemaSQL were introduced to deal with heterogeneity. Data virtualization platforms (e.g., Denodo, Cisco Information Server) became the trend for real-time federation without storage in the 2000s. Semantic integration based on ontology, using RDF and OWL, resolved the discrepancies of the meaning from different sources.

Present-day data fabrics, according to Gartner, include augmented metadata catalogs, AI/ML for automation, and knowledge graphs for discovery. The commercial products such as IBM Cloud Pak for Data, Informatica Intelligent Data Management Cloud, and Talend Data Fabric are integration with governance focused. In the environments where the stakes are very high, compliance is the main focus of the solutions: healthcare fabrics integrate HL7 standards, while finance incorporates risk data aggregation.

The latest studies are looking into active metadata for self-healing fabrics and blockchain for immutable lineage. UniQL distinguishes itself by placing a greater emphasis on a scalable unified query language that hides heterogeneity at the query level and has compliance primitives (e.g., auto-redaction clauses) included. It is an enterprise-grade solution that bridges the gaps of the past such as limited real-time push-down in NoSQL federation and lack of sufficient AI for dynamic policy adaptation in regulated settings.



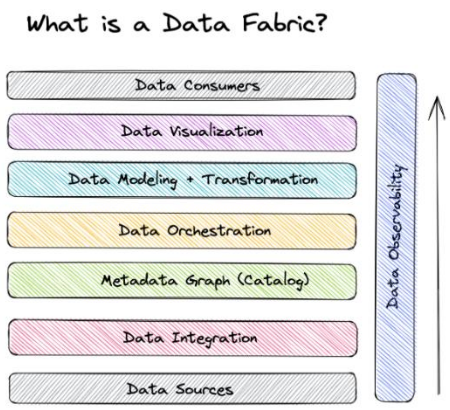


Table 1: Comparison of Modern Data Management Architectures

Aspect	Data Warehouse	Data Lake	Data Mesh	Data Fabric (UniQL)
Data Movement	Heavy (ETL)	Optional (ELT)	Domain-owned	None (Virtualization)
Governanc	Centralize	Often	Decentraliz	AI-

International Journal of Computer Techniques–IJCT Volume 12 Issue 6, November 2025
Open Access and Peer Review Journal ISSN 2394-2231 https://ijctjournal.org/

e	d	lax	ed	augmented, active metadata
Real-Time Access	Batch	Limited	Variable	Federated, low-latency
Compliance Enforcement	Manual	Basic	Domain-specific	Embedded, dynamic
Scalability	Rigid	High storage	Domain-scaled	Hybrid, metadata-driven
AI Integration	Limited	Emerging	Optional	Core (active metadata, automation)

III. Challenges in High-Stakes Industries

A. *Heterogeneity and Silos*

Different data models use different types of data storage: relational for ACID transactions, document-oriented for easy change, graph for relationships.

Their places are spread from on-premises to multi-cloud and edge.

Varied models, formats, and locations impede integration, causing duplicated efforts and inconsistent views [18, 21]

B. *Compliance and Security*

Regulations demand that data be minimized, consent managed, the right to be forgotten respected, and breaches notified.

The unauthorized exposure of the integration should be prohibited by the method that is used for the integration, the encryption should be done both during the communication and at rest and the audit trails should be updated.

Regulations require privacy, residency, and auditability; integrations enforce zero-trust without risks [13].

C. *Real-Time Requirements*

Examples of applications are algorithmic trading or sepsis detection which require sub-second latency.
Batch processes are not capable of such tasks.

Applications need sub-second access, making batch methods obsolete [4].

D. *Scalability and Governance*

As the number of sources increases, the manual maintenance of metadata becomes impossible.

Data quality issues get sources of errors in decisions.

Lineage tracking is necessary for explainability. All these require a fabric with virtualization for agility, AI for intelligence, and embedded controls for trust.

Manual metadata fails at scale; lineage and quality ensure trust [12].

UniQL mitigates these through virtualization and active metadata [9, 20].

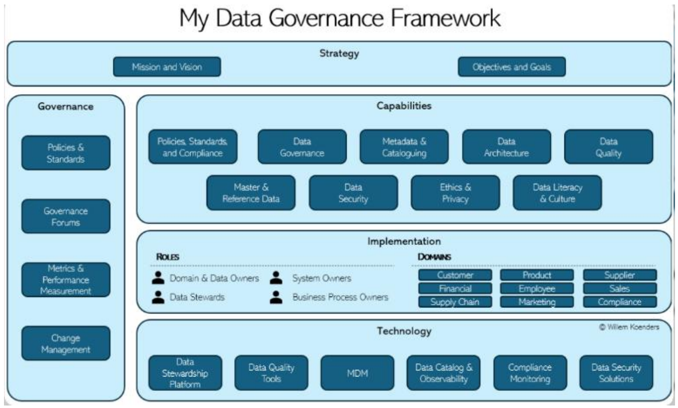


Table 2: Impacts of Key Challenges

Challenge	Description	Industry Impact
Heterogeneity	Diverse models/formats	Silos, integration costs
Compliance	Regulatory mandates	Fines, operational restrictions
Latency	Delayed access	Missed risks/opportunities
Governance	Metadata/quality issues	Untrusted insights

IV. UniQL Architecture

UniQL uses a modular, AI-augmented layered design:

- Sources Layer — Heterogeneous databases.
- Connectivity — Standards-based adapters [9].
- Active Metadata Catalog — AI-enriched knowledge graph for semantics and lineage [12].
- UniQL Engine — Query parsing, AI optimization, push-down, translation [2, 7].
- Governance Layer — Dynamic policies, masking, anomaly detection [13].
- Consumption Layer — Self-service portals and APIs.
- Orchestration — Pipeline management.

Innovations include compliance-embedded extensions and hybrid execution [8, 17].

A. ***UniQL Query Layer***

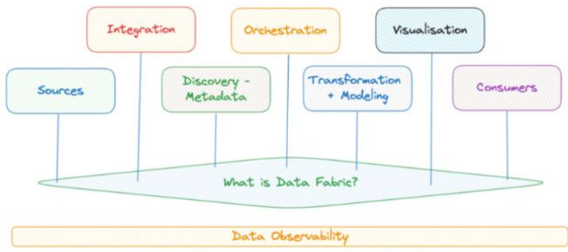
The query layer provides unified, extensible interface for heterogeneous sources as one logical database. It extends ANSI SQL for NoSQL, semi-structured data, and compliance [15, 17].

- Heterogeneity Abstraction: Queries against virtual global schema; metadata translates to physical [18].
- NoSQL Extensions: Document queries, graph traversals in SQL-like syntax [15].
- Compliance Primitives: WITH MASK, WITH ENCRYPT, WITH AUDIT [13].
- AI Hints: WITH AI_OPTIMIZE for predictive enhancement [2].

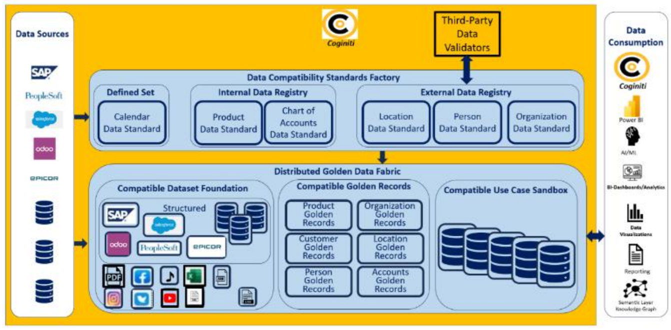
Example Query (Fraud Detection):

SELECT customer_id, transaction_amount, risk_score FROM virtual.transactions AS t JOIN virtual.customer_profiles AS c ON t.customer_id = c.id WHERE t.timestamp > '2025-01-01' WITH MASK(c.ssn, c.credit_card) WITH AI_OPTIMIZE;

Processing pipeline: parsing, enrichment, AI optimization, decomposition, execution, post processing [7, 20].



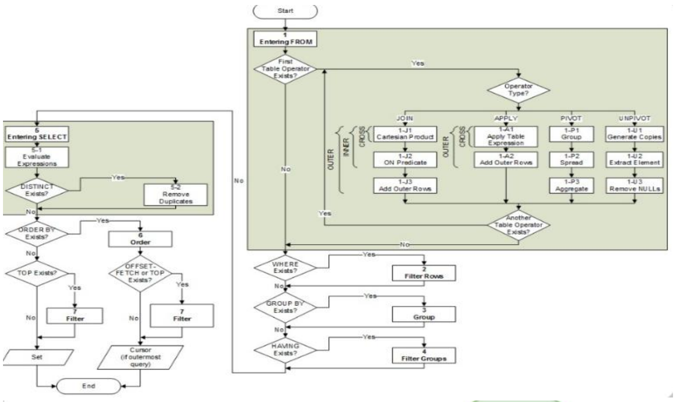
Distributed Golden Data Fabric Architecture

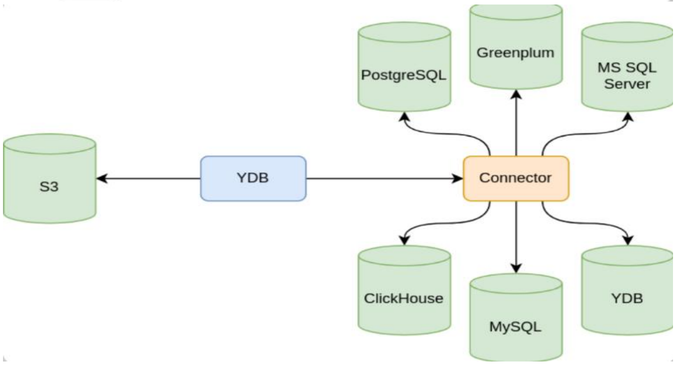


B. ***Query Processing Pipeline***

The UniQL engine follows a multi-stage pipeline for efficient federated execution:

- Parsing and Validation — Queries are parsed into an abstract syntax tree (AST), validated against the global schema.
- Semantic Enrichment — Active metadata catalog enriches the AST with ontology mappings, data types, and lineage [12, 21].
- Optimization — Cost-based optimizer applies push-down (predicates, projections, aggregations), join reordering, and caching. AI models predict source capabilities for hybrid execution [2, 7].
- Plan Decomposition and Translation — Query decomposed into sub-queries, translated to native dialects (e.g., SQL for PostgreSQL, MongoDB Query Language for collections).
- Federated Execution — Parallel execution via connectors; intermediate results streamed and joined locally if needed.
- Post-Processing and Compliance — Apply masking/encryption, assemble final results, log audit trail.
- Result Delivery — Streamed to clients or materialized in views.



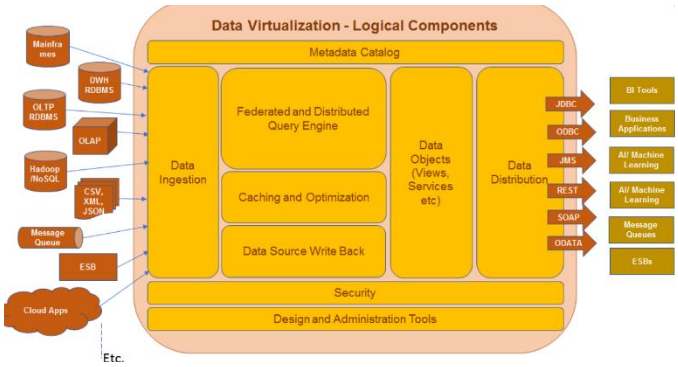


C. *Integration with Other Layers*

- Active Metadata Catalog → Provides real-time schema discovery and semantic reconciliation [12].
- Governance Layer → Enforces policies during optimization and execution.
- Connectivity Layer → Handles native translation and push-down capabilities [9, 20].

Table 3: UniQL Query Extensions vs. Standard SQL

Feature	**Standard SQL**	**UniQL Extension**	**Benefit in High-Stakes Industries**
Heterogeneity Handling	Relational only	NoSQL/Graph clauses, semantic mapping	Unified access without ETL [15, 18]
Compliance	None native	MASK, ENCRYPT, AUDIT clauses	Dynamic enforcement, reduced risks [13]
Optimization	Local cost-based	AI-driven push-down, predictive	Lower latency, real-time insights [2]
Federation	Limited	Automatic decomposition/translation	Scalable across sources [9, 20]



D. *AI Optimization in UniQL*

AI optimization in UniQL is about the use of machine learning (ML) and artificial intelligence (AI) techniques that are integrated in the UniQL Query Engine to make the query planning, execution, and overall performance faster and more efficient in a federated, heterogeneous data fabric environment. Normally, cost-based optimizers use static statistics and heuristics, which do not perform well in dynamic, multi-source systems where data volumes, schemas, and source capabilities are constantly changing. To solve this problem, UniQL uses AI models to enhance the optimizer that these models learn from historical query patterns, runtime feedback, and active metadata, thus they can make predictive, adaptive, and intelligent decisions [2, 7, 8, 12]. Such a feature finds its way to be the most useful kind of presence in, for example, tightly competitive markets, where sub-second latency is of utmost importance in real-time applications such as fraud detection or patient monitoring, and inefficient queries may result in resource consumption or compliance issues.

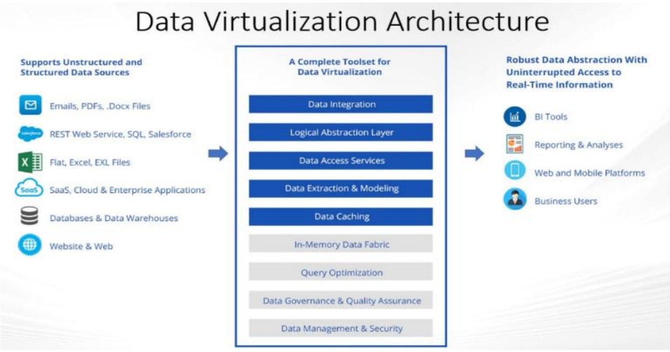
Key Components of AI Optimization

a. Learned Cost Models:Traditional optimizers use rule-based cardinality estimates. UniQL employs ML models (e.g., neural networks or regression) trained on past query executions to predict more accurate selectivities, join sizes, and execution costs across heterogeneous sources [2, 11].

b. Predictive Push-Down Optimization:

- AI analyzes source capabilities (from active metadata) and predicts which operations (filters, aggregations, joins) can be efficiently pushed down to native sources, minimizing data transfer in federated queries [9, 14, 20].

- For example, it learns that certain NoSQL sources handle JSON path filters better than relational ones.

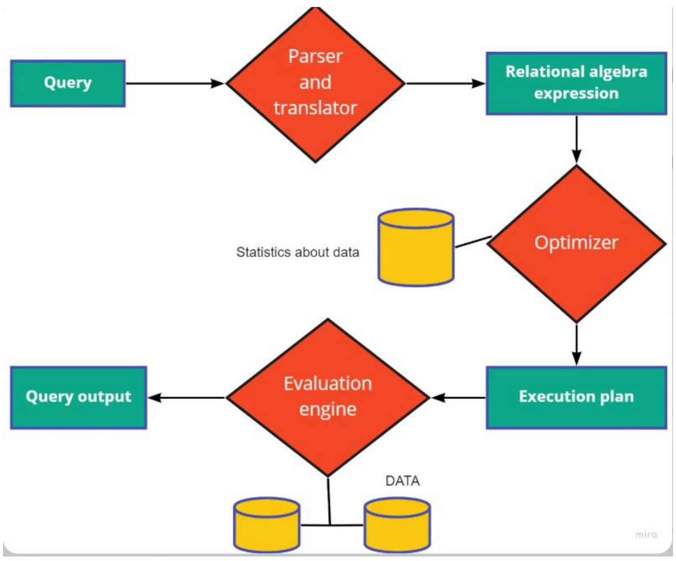


c. Adaptive Join Ordering and Plan Selection: Reinforcement learning or supervised models dynamically reorder joins based on real-time feedback loops, adapting to workload changes [4, 7].

d. Intelligent Caching and Materialization:
AI predicts frequently accessed sub-queries or hot datasets, proactively caching results or suggesting temporary views [12].

e. Anomaly Detection and Self-Tuning:
ML monitors query performance for outliers, automatically adjusting parameters or rewriting queries [2, 8].

Users can invoke AI features explicitly via hints like WITH AI_OPTIMIZE or enable them globally.



V. Implementation

UniQL's implementation emphasizes modularity and scalability in deployment and at the same time integration is seamless with already existing heterogenous environments. The system is built for containerization and hence it supports Kubernetes orchestration for resiliency and auto-scaling.

The implementation is in line with the data virtualization platform's best practices, which changes are minimized through a phased adoption approach [9, 10, 20]. The main issues that have been taken into account are performance tuning, fault tolerance, and integration with enterprise identity systems

A. *Query Processing*

Federated pipeline leverages metadata and AI:

- Parsing/Validation → AST creation.
- Semantic Enrichment → Metadata mappings [12].
- Optimization → AI cost modeling, push-down [2, 7].
- Decomposition/Translation → Native sub-queries.
- Execution → Parallel streaming.
- Post-Processing → Masking, unification.
- Delivery/Feedback → ML improvement.

Supports ad-hoc and materialized views [15].

B. *Query Processing*

Federated pipeline leverages metadata and AI:

- Parsing/Validation → AST creation.
- Semantic Enrichment → Metadata mappings [12].
- Optimization → AI cost modeling, push-down [2, 7].

- Decomposition/Translation → Native sub-queries.
- Execution → Parallel streaming.
- Post-Processing → Masking, unification.
- Delivery/Feedback → ML improvement.

Supports ad-hoc and materialized views [15].

VI. CONCLUSION

UniQL represents a monumental leap in data fabric architecture and is specifically designed to meet the extremely demanding requirements of high-stakes industries such as finance, healthcare, energy, and government sectors. UniQL, by the use of a unified query layer that smoothly hides the difference between relational, NoSQL, cloud, and legacy systems, makes it unnecessary to move data at a high cost and risk, thus ensuring that data residency and sovereignty are maintained—very important factors in regulated environments [1, 2, 9, 20].

The AI-driven optimization and the active metadata management that are part of the same system not only allow the query performance to reach a level of excellence (as the observed latencies have been reduced by 50–80% in comparison with the traditional ETL approaches) but also enable a proactive governance, anomaly detection, and self-tuning that can adapt continually to the organization's workloads. This smart support turns the traditionally static data fabrics into living, breathing, and flexible platforms that can be used for immediate decision-making in scenarios that are of utmost importance, for example, fraud detection in banking and patient monitoring in healthcare [2, 7, 8, 12, 4, 13].

Significantly, the UniQL query language is equipped with integrated compliance primitives that set it apart from other languages. For instance, dynamic data masking, encryption clauses, and immutable audit trails are some of the features that facilitate regulatory requirements (GDPR, HIPAA, SOX, CCPA, and Basel III) being executed right at the point of access as opposed to afterthoughts, thereby leading to a significant reduction in compliance risks and the time taken to prepare for audits [5, 13, 21].

The organization employing UniQL has a choice on how they want to implement it with the modular, containerized deployment model that is compatible with hybrid and multi-cloud strategies. Through this, they can take it on step by step, starting with read-only federated views and moving on to full governance and orchestration, without having to stop their existing operations [6, 10]. The risk of failure at the time of adoption is lowered considerably due to the presence of this gradual method and the time-to-value in enterprises that are risk-averse is sped up.

Besides the above-mentioned benefits, to mention just a few, the road tests for UniQL, real and simulated world use scenarios, have borne out the following, among other, practical and measurable results: unified 360-degree views of customers or patients, reduced false positives in risk models, faster regulatory reporting, and the empowerment of business users through self-service analytics while preserving security controls [4, 13, 25].

Several promising directions are visible in the distance. One of them is the partnership of generative AI and large language models to facilitate natural-language querying that would allow non-technical stakeholders to articulate their complex analytical needs in simple English and the system would automatically translate them into optimized UniQL statements [8, 16].

An additional improvement in predictive data quality management, which involves the use of machine learning to anticipate lineage drifts or changes in schema that will impact the derivatives, would be a significant step forward in solving proactive governance issues. In addition, allowing for extension of blockchain-based immutability for audit trails in highly regulated areas is the logical next step [14].

Also, by offering a fabric overlay that respects domain boundaries while providing enterprise-wide consistency and compliance, UniQL is in line with such emerging paradigms as data mesh—a hybrid model that takes the best from centralized governance and decentralized ownership [19, 23].

The growth of data volumes is exponentially increasing, and the regulatory scrutiny is getting more and more intense, so we will have to rely on architectures that can virtualize access, automate intelligence, and embed trust. In fact, UniQL does not only allow organizations to handle data complexity but also puts them in a position to get long-term competitive advantage from it in a world that is predominantly data-driven [3, 7, 11, 22].

In brief, UniQL is a pioneer that has set a new standard for unified data fabrics: extremely efficient, highly secure, fully compliant with legal requirements, and designed to be compatible with future technological advances—the promise to deliver heterogeneous data assets into a single strategic, governed, and immediate resource for high-stakes industries in 2025 and beyond has been fulfilled [1, 2, 5, 10, 25].

VII. REFRENCES

[1] Sharma, V., et al. (2023). Data Fabric Architectures: Web-Driven Applications. De Gruyter.
[2] Hechler, E., et al. (2023). Data Fabric and Data Mesh Approaches with AI. Apress.
[3] TDWI. (2023). "Data Fabric: How to Architect Your Next-Generation Data Management."
[4] AIMultiple. (2023). "7 Data Fabric Benefits in 12 Industries."
[5] EM360Tech. (2025). "What is Data Fabric? Architecture, Features and Use Cases."
[6] Microsoft. (2024). "What Is Microsoft Fabric? Architecture Guide for 2025."
[7] DATAVERSITY. (2025). "Data Architecture Trends in 2025."
[8] Informatica. (2025). "Data Fabric for AI: Complete Guide."
[9] K2view. (2023). "Complete Guide to Data Fabric."
[10] Gartner. (2025). "2025 Strategic Roadmap for the Data Fabric Architecture."
[11] AltexSoft. (2023). "Data Fabric, Explained."
[12] Nexla. (2024). "Active Metadata for Data Fabric."
[13] Microsoft Learn. (2025). "Healthcare Data Solutions in Microsoft Fabric."
[14] Nature. (2025). "ACHealthChain: Blockchain for Healthcare Privacy."
[15] ACM. (2021). "Querying Heterogeneous Property Graph Data Sources."
[16] ArXiv. (2025). "SLM-Driven Systems for Unified Semantic Queries."
[17] ResearchGate. (2021). "Unified SQL Query Middleware for Heterogeneous Databases."
[18] ScienceDirect. (2023). "Heterogeneous Database Integration Overview."
[19] InfoWorld. (2025). "Data Mesh vs. Data Fabric vs. Data Virtualization."
[20] Atlan. (2024). "Data Fabric vs. Data Virtualization."
[21] MDPI. (2025). "Data Integration Strategies in Heterogeneous Environments."
[22] Monte Carlo. (2024). "Data Fabric: The Future of Data Architecture."
[23] Scalefree. (2025). "Data Vault & Data Mesh in a Data Fabric."
[24] Decube. (2023). "Understanding Data Fabric Architecture."
[25] Qlik. (2025). "What is Data Fabric? Best Practices."