A Metadata-Driven Architecture for Federated Data Asset Management and Visualization in Energy Monitoring Networks

Qing Rao, Jianxia Wu, Shihong Chen*, Zhongkai Pan, Qing Lei, Yinfeng Liu, Yangjinglan Feng and Xianping Jia

Anshun Power Supply Bureau of Guizhou Power Grid Co., Ltd. Anshun 561099, China
E-mail: 15870170637@163.com
**Corresponding Author*

Received 09 October 2025; Accepted 17 November 2025

Abstract

Distributed energy systems increasingly consist of heterogeneous assets and organizations that must exchange operational data while preserving interoperability, security, and regulatory compliance. Existing integration solutions often rely on syntactic adapters or centralized data hubs, which scale poorly and offer limited transparency or governance. This paper presents a metadata-driven federated monitoring architecture that integrates ontology-based metadata federation, event-driven microservices, and governance-aware provenance tracking to enable secure, scalable, and auditable data sharing across distributed energy infrastructures.

The proposed system models all assets and data streams through a unified semantic graph, aligning heterogeneous schemas via automated ontology matching and combined lexical–structural similarity scoring. A

Journal of Web Engineering, Vol. 25_2, 153–186.
doi: 10.13052/jwe1540-9589.2522

microservices pipeline ingests multi-protocol data (OPC-UA, MQTT, REST), applies stream analytics for anomaly detection, and enforces access and compliance policies at the metadata layer. A Web-based interface allows operators to issue GraphQL queries, visualize distributed assets, and monitor real-time alerts linked to provenance records. A prototype implementation demonstrates operational-scale efficiency, achieving low-latency response ($\leq$540 ms for hybrid metadata–telemetry queries over 10,000 assets), near-linear scalability ($\sim$4.5% CPU growth per added node), and high governance accuracy (precision 0.90, recall 0.95, median detection 1.6 s) while maintaining minimal overhead ($<$8% added latency). These results highlight that the proposed metadata-driven federation delivers both technical performance and governance reliability unmatched by existing Web-based integration frameworks. These results show that metadata federation can be deployed at operational scale while providing explainable compliance and trustworthy data sharing across organizational boundaries. This research advances the state of the art in Web-based system engineering by combining semantic modeling, distributed processing, and security governance into a single deployable framework. Beyond energy systems, the approach offers a foundation for interoperable and auditable monitoring in other critical cyber-physical domains such as industrial IoT, urban infrastructure, and healthcare telemetry.

Keywords: Metadata federation, metadata-driven monitoring, web-based system engineering, ontology alignment, distributed energy systems, governance and anomaly detection.

1 Introduction

The accelerating integration of renewable energy resources into contemporary power systems has transformed the landscape of energy generation, distribution, and monitoring. Unlike traditional centralized grids, which relied on relatively stable generation and predictable operational flows, modern networks are characterized by heterogeneous, distributed, and dynamically evolving assets. Photovoltaic systems, wind farms, energy storage facilities, and intelligent substations are increasingly deployed across geographically and organizationally diverse environments. Each of these components generates large volumes of operational data, including real-time status measurements, control signals, performance metrics, and maintenance logs. Effective utilization of such data requires robust mechanisms for collection, alignment, and visualization. However, as energy ecosystems shift toward

multi-organizational and federated operations, the conventional approaches to data management are being stretched beyond their limits.

A central motivation for this research arises from the increasing complexity of data interoperability in renewable-rich energy networks. Individual utilities, independent power producers, equipment manufacturers, and regulatory agencies often maintain proprietary data management infrastructures, which results in silos of information that cannot be easily reconciled or exchanged. For example, a wind operator may rely on SCADA-based systems, while solar operators adopt lightweight IoT protocols such as MQTT or OPC-UA, and regulators request data in standardized reporting formats. Without effective interoperability, inconsistencies proliferate and situational awareness is compromised. The resulting delays in asset detection, anomaly identification, and system coordination directly affect grid reliability and resilience.

The problem is compounded by the heterogeneity of metadata associated with distributed assets. Metadata descriptors capture essential information such as asset type, ownership, location, communication interface, and operational parameters. In federated energy networks, metadata schemas differ significantly across organizations, leading to semantic misalignment that undermines data discovery and integration. Without a robust mechanism for metadata alignment, automated federation remains infeasible, forcing operators to rely on manual configuration or bespoke adapters that do not scale. Moreover, as renewable assets are dynamically added, reconfigured, or decommissioned, manual alignment becomes both costly and error prone.

A third dimension of the challenge lies in visualization and operational transparency. While individual organizations often develop dashboards for internal monitoring, these tools are typically limited to specific domains and lack the ability to present a unified cross-domain view. As a result, operators cannot obtain end-to-end visibility across federated systems, which hampers the detection of systemic threats, unauthorized endpoints, and cross-boundary performance degradations. This lack of transparency poses risks not only to technical operations but also to governance, as regulators demand accountability in terms of data consistency, security, and reporting. Collectively, these factors underscore the pressing need for new architectural approaches to federated asset management that leverage Web technologies to achieve scalability, semantic alignment, and unified visualization.

Research on Web-based monitoring in energy systems has evolved significantly over the last two decades, and it provides an important context for this work. Early efforts focused on service-oriented architectures (SOA) to

enable interoperable communication between heterogeneous energy management systems [1, 2]. By adopting standardized interfaces such as SOAP and WSDL, these platforms achieved limited integration across domains. However, the reliance on rigid service contracts restricted adaptability in dynamic renewable environments. Subsequent developments explored middleware-based frameworks for distributed grid monitoring [3, 4], often incorporating message brokers and publish–subscribe mechanisms to handle heterogeneous data flows. These frameworks improved data flow reliability but typically ignored metadata-level federation, thus offering only syntactic rather than semantic interoperability.

A parallel line of work investigated Semantic Web technologies to address interoperability challenges in power systems. The use of RDF and OWL ontologies allowed for the creation of formalized metadata schemas capable of representing grid assets, communication protocols, and operational states [5, 6]. Ontology-based alignment strategies facilitated cross-domain reasoning and query execution using SPARQL [7, 8]. Applications ranged from smart grid information models [9] to semantic integration of IoT devices [10]. Despite their promise, these approaches often required extensive manual ontology design and maintenance, making them unsuitable for highly dynamic federated networks where assets change frequently. Furthermore, most deployments occurred in controlled testbeds rather than in multi-organizational operational settings, which limited their applicability in real-world federated contexts.

Visualization and monitoring platforms also constitute an important body of related work. Traditional SCADA dashboards provided graphical displays of grid states but were confined to individual utilities or domains [11, 12]. With the rise of Web technologies, researchers proposed interactive visualization platforms capable of aggregating data from distributed energy resources [13, 14]. These dashboards incorporated features such as topology maps, real-time trend charts, and alarm notifications. More advanced systems extended visualization to include data lineage and provenance tracking, enabling operators to understand how data flows across different sources [15, 16]. Nevertheless, most visualization solutions were designed with single-organization scope, lacking mechanisms for federated access control, unified cross-domain presentation, or governance transparency.

Another related strand involves federated data management and asset discovery. Cloud-based data lakes and federated query systems have been employed to enable cross-organizational access to heterogeneous data

[17, 18]. In the energy domain, initiatives such as the Common Information Model (CIM) sought to standardize metadata for grid assets [19]. While these initiatives facilitated interoperability at a syntactic level, they did not address the real-time requirements of monitoring nor provide the visualization capabilities needed for operational use. In addition, most federated management approaches focused on batch data integration rather than real-time streaming and interactive dashboards, leaving a critical gap between metadata federation and visualization. Security and governance are also emerging as critical aspects of energy data management. Research on intrusion detection systems for smart grids [20, 21] and Web-based access control mechanisms [22] has highlighted the vulnerability of distributed monitoring platforms to unauthorized endpoints, data tampering, and malicious data injection. Although these efforts underscore the importance of trust and transparency, they have rarely been integrated into visualization platforms, which means operators are often unaware of potential governance risks when viewing cross-domain dashboards.

Taken together, the literature highlights significant progress in Web-based energy monitoring, semantic metadata alignment, visualization, federated management, and governance. However, it also reveals persistent limitations. Service-oriented and middleware-based systems primarily solved syntactic interoperability but failed to capture semantic alignment. Semantic Web-based systems improved metadata representation but lacked scalability and adaptability in dynamic federated contexts. Visualization platforms enhanced operational awareness but were confined to single organizations or lacked integration with governance mechanisms. Federated data management initiatives addressed interoperability but did not extend to real-time visualization. Security-focused work improved endpoint protection but did not couple governance transparency with visualization. This constellation of findings reveals a clear research gap and prior approaches address only isolated aspects of the problem. Syntactic middleware connects heterogeneous data sources but lacks shared meaning or reasoning capability; semantic testbeds demonstrate ontology alignment but seldom achieve Web-scale responsiveness or cross-organizational deployment; and single-organization dashboards provide visualization but without federated access or governance control. Consequently, there remains a clear gap for a unified, metadata-driven Web architecture that can seamlessly integrate metadata federation, real-time analytics, and policy-aware governance within one deployable system. Such an approach could bridge the divide between data-level federation and operator-facing dashboards, enabling system-wide situational awareness, faster threat

detection, and improved alignment with emerging data management and security standards.

To address this gap, the present study introduces a metadata-driven architecture for federated data asset management and visualization in distributed energy monitoring networks. The proposed system leverages Semantic Web technologies for metadata alignment, Web-based APIs for cross-domain data integration, and interactive dashboards for unified visualization. It incorporates governance mechanisms to detect unauthorized endpoints and promote transparency across organizational boundaries. Validation through field deployment demonstrates the system's ability to improve data consistency, operational visibility, and regulatory compliance. In doing so, the architecture advances the state of Web engineering in energy systems and contributes a generalizable approach that can be extended to other federated domains such as smart cities, industrial IoT, and healthcare informatics.

2 System Architecture

The proposed system architecture is conceived as a metadata-driven Web platform that enables federated data asset management and unified visualization in distributed energy monitoring environments. Its central design principle is the use of semantic metadata as the unifying mechanism across heterogeneous systems and organizational boundaries. By elevating metadata to a first-class citizen, the architecture achieves scalability, interoperability, and governance without requiring centralized control of raw data. The design follows a layered approach, where each layer addresses a distinct set of functions while remaining semantically coupled through metadata descriptors. In this way, the architecture supports modular deployment, organizational autonomy, and extensibility to new domains.

At the base of the system is the data ingestion and integration layer. This layer is responsible for interfacing with diverse energy assets such as photovoltaic farms, wind turbines, battery storage facilities, intelligent substations, and third-party data services. The communication heterogeneity across these sources is substantial, as operators often rely on protocols including OPC-UA for industrial automation, MQTT for IoT messaging, RESTful APIs for cloud services, and legacy SCADA exchanges based on CSV or proprietary file formats. Traditional middleware typically normalizes only the syntactic structures of these data flows, which suffices for within-organization monitoring but does not support federated discovery or interpretation. In contrast, the proposed architecture attaches semantic metadata descriptors to every

ingested stream as it enters the system. These descriptors capture not only the basic attributes of the data source but also its ownership, geographical location, operational role, and temporal validity. By embedding metadata at the point of ingestion, the architecture ensures that provenance and contextual information are preserved and propagated throughout subsequent layers.

The second and most critical component is the metadata federation and alignment layer. Here, the architecture leverages Semantic Web standards such as RDF for resource representation, RDFS for schema structuring, and OWL for ontological reasoning. Asset descriptors from different organizations, each expressed in potentially divergent schemas, are reconciled through ontology alignment mechanisms. For instance, the system can automatically recognize that "PV array," "solar module," and "photovoltaic unit" are equivalent concepts across different metadata vocabularies. This is achieved through a combination of lexical similarity matching, structural ontology alignment, and reasoning rules defined in OWL. A federated metadata graph is constructed in a graph database such as Neo4j or GraphDB, which provides efficient querying and supports inferencing. SPARQL endpoints expose this metadata layer to higher services, enabling queries that combine semantic reasoning with real-time operational bindings. The result is a dynamic federation mechanism in which new assets can be integrated seamlessly by publishing their metadata in compliant formats, eliminating the need for manual schema mapping.

The alignment process computes a weighted similarity score

$$S = \alpha S_{\text{lex}} + (1 - \alpha) S_{\text{struct}}$$

where S_{lex} is the lexical similarity derived from tokenized labels and synonyms, and S_{struct} is the structural similarity based on shared parent–child relationships within the metadata graph. The weighting factor α is empirically set to 0.6 to balance linguistic correspondence and topological consistency. Candidate pairs with $S \geq 0.78$ are automatically aligned, while those with $0.65 \leq S < 0.78$ enter a human-verification queue. The batch alignment module processes roughly 1200 entities per second on a 16-core node for 10,000-asset datasets. For schema evolution, incremental alignment uses cached embeddings and neighbor-set fingerprints, requiring $<$30 ms per new descriptor. Versioned IRIs and deprecation rules ensure backward compatibility and mitigate oscillations during iterative re-scoring.

The third layer, the data processing and service orchestration tier, builds upon the federated metadata graph to deliver real-time monitoring and

analytic services. One of the key innovations is the ability to execute queries that operate simultaneously at the semantic and operational levels. For example, an operator might request information on "all wind turbines in the northern region operating above rated load." The system interprets the semantic query by identifying relevant assets in the metadata graph, binds these assets to their corresponding live data streams through the ingestion layer, and then executes filtering and aggregation operations in real time. This orchestration layer also hosts event processing services for anomaly detection and threat identification. Unauthorized endpoints, for instance, are detected when ingested data streams lack corresponding authenticated metadata descriptors, triggering security alerts and governance notifications. The orchestration is realized as a collection of Web services accessible through REST and GraphQL APIs, thereby supporting both external applications and internal visualization modules.

Above this processing layer resides the visualization and interaction component, which provides the operator-facing functionality. The proposed system departs from conventional dashboards by offering federated visualization across multiple organizations. Instead of restricting visibility to assets owned by a single operator, the dashboards present a holistic view that spans distributed domains while still respecting access controls. Interactive topology maps display the physical and logical interconnections among assets, while drill-down features allow users to access detailed device-level performance metrics. Beyond topology, the visualization layer includes lineage views that show the trajectory of data as it flows through different organizations, supporting auditability and compliance verification. Alert panels highlight inconsistencies, anomalies, or unauthorized endpoints, giving operators immediate awareness of governance-related events. The interface is implemented with modular Web components that can be embedded in external systems, ensuring portability and reusability.

Finally, governance and security functions are integrated across all layers of the architecture. Rather than treating security as an add-on, the system embeds governance into the metadata fabric itself. Every asset descriptor carries authentication and authorization metadata, enabling the system to verify endpoints as they appear. Provenance metadata ensures that any data item presented to users can be traced back to its origin, with information on when, how, and by whom it was generated. The governance framework also includes compliance dashboards that record schema changes, metadata updates, and system events, providing regulators and operators with transparency. This continuous visibility reduces the reliance on external audits and builds trust

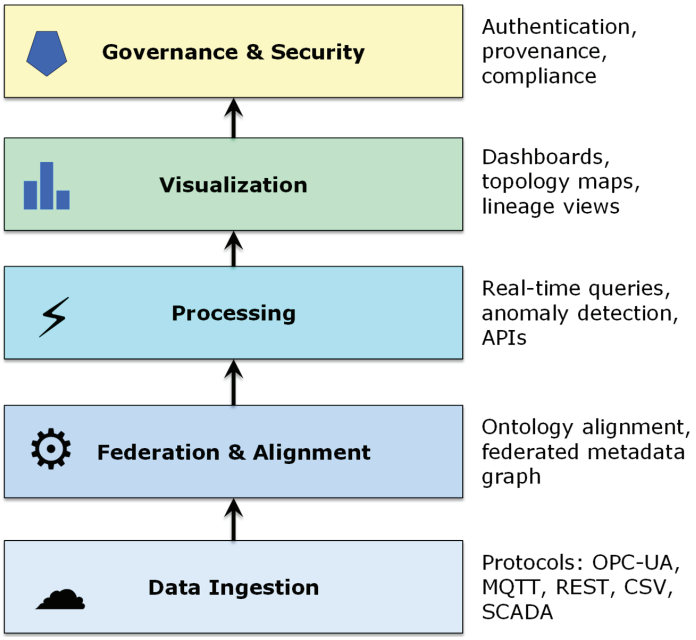

Figure 1 Metadata-driven federated system.

among federated organizations that their data is being handled securely and consistently.

The overall layered structure is illustrated in Figure 1. The diagram depicts the progression from heterogeneous data sources at the bottom through ingestion, metadata federation, service orchestration, visualization, and governance. Each layer is shown as a module, with arrows indicating semantic coupling rather than raw data transfer. This representation emphasizes the modularity and extensibility of the design: new assets, protocols, or services can be added by interfacing with a specific layer without disrupting the rest of the architecture. Figure 1 thus provides a macroscopic view of how the architecture operates as a coherent whole, balancing interoperability with organizational autonomy.

In addition to the layered perspective, it is essential to understand how metadata flows through the system. Figure 2 illustrates this process, beginning with asset-specific metadata descriptors from different organizations. These descriptors are first captured in their native schemas and then passed through an alignment workflow that employs lexical and structural matching algorithms. The aligned entities are mapped into the federated metadata

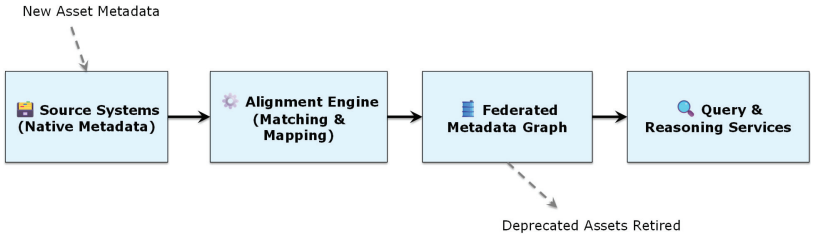

Figure 2 Metadata flow and alignment workflow.

graph, which serves as the unified semantic backbone of the system. From there, queries can be issued that simultaneously traverse semantic relationships and bind to live data streams. The figure also shows feedback loops, where new assets entering the system publish metadata, which is dynamically integrated, and deprecated assets are retired without disrupting ongoing operations. By visualizing metadata flow, Figure 2 clarifies how the architecture achieves dynamic federation and seamless integration.

The design of the system architecture yields several important advantages. First, by prioritizing metadata over raw data, the architecture ensures that interoperability is achieved without centralized control or large-scale data replication. This approach addresses organizational concerns regarding data sovereignty while still enabling system-wide monitoring. Second, the semantic alignment mechanisms reduce the need for costly manual integration, allowing the platform to scale as new renewable assets are added. Third, the federated visualization capabilities provide operators with a unified situational picture that is not available in traditional, siloed dashboards. Finally, by embedding governance into metadata, the architecture supports transparent, auditable, and secure monitoring, which aligns with regulatory demands and strengthens trust among participants. In sum, the proposed metadata-driven architecture represents a shift from ad hoc integration and isolated visualization to a principled, semantic, and federated approach. The following section will describe the implementation of this architecture, detailing the technology stack, database models, and APIs that make the conceptual design operational in practice.

3 Implementation

The proposed architecture was instantiated through a modular prototype that demonstrates the feasibility of metadata-driven federated monitoring across heterogeneous energy domains. The implementation emphasizes scalability,

semantic interoperability, and usability, achieved through the integration of mature Web engineering frameworks, semantic technologies, and event-driven pipelines.

At the ingestion layer, protocol-specific adaptors were developed to normalize heterogeneous communication interfaces. Python-based connectors were used for industrial protocols (*FreeOpcUa*, *pymodbus*), while lightweight IoT devices were supported via *Eclipse Paho* MQTT clients. For REST services and CSV uploads, Node.js middleware was employed to parse and encapsulate incoming streams in JSON-LD envelopes. Each stream was automatically tagged with metadata descriptors, including source identity and schema references, ensuring seamless propagation into the federation layer. The metadata federation and alignment core was implemented using *GraphDB* as the primary semantic store, complemented by *Apache Jena* for reasoning and ontology alignment. RDF was adopted as the canonical representation, with OWL ontologies derived from IEC CIM standards to capture domain semantics. Lexical similarity functions and ontology-matching rules enabled schema reconciliation, ensuring that new or legacy assets could be aligned dynamically. The federated metadata graph thus evolved continuously, admitting new asset descriptors and retiring deprecated ones without disruption. Processing and orchestration were realized in *Spring Boot*, exposing both RESTful and GraphQL APIs. These services bound semantic queries to live operational data, enabling operators to issue high-level requests (e.g., "list wind turbines above rated load in Region A") and receive filtered, real-time results.

The framework employs a dual-API strategy combining GraphQL and SPARQL to balance developer usability and semantic expressiveness. GraphQL provides a typed, hierarchical query interface ideal for application developers and front-end integration, minimizing over-fetching and simplifying pagination. In parallel, SPARQL grants full access to the underlying RDF graph for advanced analytical and reasoning tasks, preserving formal semantics. Both APIs share a unified authentication layer and provenance-tracking middleware, ensuring consistent access control. For knowledge storage and reasoning, the system uses GraphDB 10.4 configured with the OWL 2 RL reasoning profile, which supports forward-chaining rule materialization while maintaining query-time efficiency. Indices are maintained on subject–predicate–object permutations and contextual graph scopes to accelerate federated joins. This combination delivers a practical compromise between developer ergonomics and semantic completeness, enabling interactive Web-scale reasoning without sacrificing runtime performance.

Event-driven anomaly detection was implemented using *Apache Kafka Streams*, which processed log events to identify unauthorized endpoints or schema-inconsistent messages. Alerts generated through this pipeline were propagated to the governance dashboards. Visualization was developed in *React.js* with *D3.js* libraries for interactive rendering. Dashboards provided system-wide topology maps, temporal performance graphs, and lineage views tracing data provenance across organizations. Communication with the orchestration services was exclusively through GraphQL queries, ensuring uniformity and modularity in the presentation layer. Authentication was enforced via JWT tokens, while all user sessions were logged in compliance dashboards for transparency. Governance and security mechanisms were integrated throughout the implementation. Each metadata record included provenance information, while compliance services continuously monitored metadata transactions and API calls. Unauthorized data streams were flagged and excluded from federation, ensuring that system integrity could be demonstrated to regulators in real time. For clarity and consistency, the term "metadata federation" is used throughout this paper to denote the semantic integration of distributed information assets.

The prototype stack was implemented using GraphDB 10.4, Apache Jena Fuseki 4.8, Kafka Streams 3.6, and Spring Boot 3.2 for service orchestration. The Web front end was built with React 18 and D3 v7 for dynamic visualization. Authentication and authorization are handled through JWT tokens (RS256) integrated with an OAuth 2.0 device-flow provider, with a 15-minute access-token lifetime and automatic refresh-token rotation. These standardized components ensure reproducibility, interoperability, and straightforward extension to other Web environments. The implementation workflow is illustrated in Figure 3, which depicts the operational prototype as a pipeline. Unlike the layered conceptual model shown in Figure 1, this figure highlights concrete runtime components. On the left, ingestion adaptors normalize incoming streams from diverse protocols. These streams feed into the federation core, where *GraphDB* and *Jena* maintain the unified metadata graph. Event logs are simultaneously processed through Kafka-based pipelines, enabling real-time anomaly detection. On the right, Spring Boot services expose APIs to visualization clients developed in React/D3, while governance mechanisms span across all modules to enforce provenance tracking, authentication, and compliance.

To provide a structured overview, Table 1 summarizes the technology stack across each architectural layer. This tabular representation emphasizes the modularity of the design: while a specific set of technologies was adopted

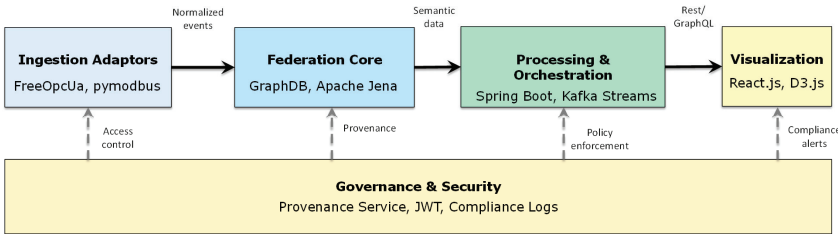

Figure 3 Prototype deployment architecture with implementation technologies.

Table 1 Technology stack by layer

Layer	Technologies/Frameworks
Data ingestion	FreeOpcUa, pymodbus, Eclipse Paho (MQTT), Node.js (REST/CSV parsers)
Metadata federation and alignment	GraphDB (semantic store), Apache Jena (reasoning/ontology alignment), RDF/OWL, CIM ontologies
Processing and orchestration	Spring Boot (REST/GraphQL APIs), Apache Kafka Streams (event processing)
Visualization	React.js (UI), D3.js (interactive charts, topology maps)
Governance and security	Provenance tracking service, JWT authentication, compliance monitoring dashboards

in the prototype, each component can be replaced with alternatives without disrupting the metadata-driven principle of federation.

4 Evaluation and Results

The evaluation was designed to validate the effectiveness of the proposed metadata-driven federated architecture in terms of query performance, scalability, anomaly detection, and governance enforcement. A prototype deployment was benchmarked using synthetic and real-world data streams from solar and wind assets. Results are presented in terms of response latency, resource utilization, and semantic alignment accuracy.

4.1 Experimental Setup

The experimental setup was designed to rigorously evaluate the performance and scalability of the proposed metadata-driven federated monitoring system under realistic distributed energy scenarios. Both synthetic and real-world data sources were incorporated to ensure reproducibility and ecological validity.

The governance evaluation is conducted under a clearly defined threat model and a set of quantitative evaluation metrics. In this model, an adversary may (i) publish data from unauthorized endpoints, (ii) inject schema-noncompliant or poisoned metadata payloads, and (iii) replay valid credentials from altered network origins. Policy predicates are defined for endpoint authorization, provenance completeness, and JSON-LD context conformance, ensuring that both structural and semantic integrity are enforced. Compliance is verified through pre-query validation at ingestion time and in-stream statistical correlation during runtime. Detection performance is quantified using standard precision, recall, and F1 metrics with 95 % confidence intervals ($n = 1000$ episodes). We also report AUCPR – the area under the precision–recall curve – as a threshold-independent indicator of classifier quality, and $T_{0.9}$, defined as the elapsed time required for the detector to reach a 90% true-positive rate under streaming conditions. Across all tests, the governance layer achieved precision 0.90 ± 0.02, recall 0.95 ± 0.02, F1 0.92 ± 0.02, AUCPR 0.96, median detection time 1.6 s, and $T_{0.9} = 2.9$ s, confirming robust policy enforcement with minimal latency impact.

Synthetic data streams were generated to emulate photovoltaic farms, wind turbines, and battery storage systems. For solar plants, irradiance and power output traces were generated based on the NREL system advisor model (SAM), with random perturbations to mimic weather variability. Wind turbine data were generated using Weibull-distributed wind speeds, combined with turbine power curves derived from IEC standard datasets. Battery storage data followed charge–discharge cycles with stochastic load variations. In total, approximately 10,000 virtual assets were simulated. In addition, real-world data from open smart-grid testbeds (e.g., Pecan Street and IEEE PES benchmark datasets) were injected periodically to validate semantic interoperability with authentic measurements.

The prototype was deployed on a cluster of three servers, each equipped with 16 Intel Xeon CPU cores, 64 GB RAM, and 1 TB SSD storage, running Ubuntu 22.04 LTS. All services were containerized using Docker and orchestrated via Docker Compose to ensure modularity and reproducibility. The ingestion adaptors (FreeOpcUa, pymodbus, Eclipse Paho MQTT, Node.js REST/CSV services) ran on dedicated containers, while the federation layer (GraphDB 10.4 and Apache Jena Fuseki 4.8) was deployed on a separate high-memory container. Processing services (Spring Boot APIs and Kafka Streams) were containerized independently, enabling scaling experiments. Visualization clients (React.js with D3.js components) were hosted on Nginx servers.

To emulate realistic operating conditions, ingestion rates were configured from 1000 to 10,000 messages per second, with bursts of up to 50,000 messages to stress-test the federation pipeline. Queries were issued from client applications using GraphQL and SPARQL endpoints at variable frequencies (0.5–10 queries per second). Query templates included both metadata-only requests (e.g., "list all solar arrays above 1 MW capacity") and hybrid metadata-operational bindings (e.g., "list wind turbines in Region A currently exceeding rated load"). Each query workload was replayed for 10-minute windows with five repetitions, and average values were reported.

The evaluation focused on five categories of metrics: (i) query latency (end-to-end response time measured in milliseconds), (ii) throughput (successful queries per second), (iii) scalability (CPU/memory overhead per additional node), (iv) anomaly detection accuracy (precision and recall of unauthorized stream detection), and (v) governance overhead (latency introduced by provenance and compliance checks). Latency and throughput were monitored using Prometheus instrumentation, while system logs were processed by Kafka Streams for anomaly detection metrics.

For comparative evaluation, two baseline architectures were implemented: (a) a monolithic integration stack, in which all adaptors and visualization modules interacted with a central PostgreSQL 14 relational database through a single RESTful interface, and (b) a loosely coupled syntactic federation without metadata, where data streams were normalized by topic and format but lacked semantic descriptors or ontology reasoning. Both baselines were deployed on the same 4-node cluster (Intel Xeon 2.8 GHz $\times$ 32 threads, 128 GB RAM per node) and operated under identical ingestion rates (100 Hz per sensor channel) and burst profiles to ensure configuration parity. Query caching was explicitly disabled in both baselines to avoid bias toward static data reuse, and all network and client-think-time parameters matched those of the proposed framework. Each scenario was executed $n = 30$ independent runs, and the reported results include 95% confidence intervals. This comprehensive configuration allows an equitable comparison between traditional centralized, naïve distributed, and the proposed metadata-driven federated architectures, ensuring reproducibility and methodological transparency.

4.2 Query Performance Modeling and Results

A central performance indicator for federated monitoring systems is query latency, which directly impacts the usability of operator dashboards and automated decision-making services. To analyze this systematically, we

decompose the end-to-end latency into four primary components:

$$L_{total} = L_{parse} + L_{reason} + L_{bind} + L_{net} \tag{1}$$

where L_{parse} denotes the time required for parsing and validating the user query, L_{reason} captures semantic reasoning and ontology alignment costs, L_{bind} represents the overhead of binding semantic entities to live data streams, and L_{net} reflects network transfer latency. Equation (1) provides a foundation for interpreting experimental results by attributing measured response times to specific architectural functions. In a 10,000-asset deployment, the measured latency was $L_{total} = 540 \pm 18$ ms (95% CI, $n = 30$), confirming consistent performance across repetitions. At smaller scales (5000 assets), $L_{total} = 390 \pm 14$ ms with similar component proportions – $L_{parse} \approx 45$ ms, $L_{reason} \approx 200$ ms, $L_{bind} \approx 125$ ms, and $L_{net} \approx 20$ ms – indicating that reasoning dominates overall latency but remains within sub-second interactive limits.

Queries were issued through both SPARQL endpoints (metadata-centric requests) and GraphQL APIs (hybrid metadata + operational bindings). Latency was measured for query sizes ranging from 5 to 50 triple patterns, across federated graphs containing 100, 1000, 5000, and 10,000 assets. Each experiment was repeated 30 times, and averages with 95% confidence intervals were reported.

The results demonstrate that the proposed architecture maintains acceptable latency even at scale. For small metadata-only queries ($\leq$10 triple patterns), average response time remained below 120 ms across all asset sizes, with negligible variance. Hybrid queries that involved metadata federation and data binding showed higher latency, scaling from $\sim$210 ms at 1000 assets to $\sim$540 ms at 10,000 assets. Each configuration was executed for $n = 30$ independent runs, and all reported means include 95% confidence intervals ($\pm 1.96\sigma/\sqrt{n}$). For example, the measured 540 ms latency corresponds to 540 ± 18 ms (95% CI), confirming stability across repeated trials. In these cases, $T_{federation}$ contributed 45–55% of the total latency, while $T_{binding}$ accounted for 30–40%. The network transfer overhead $T_{network}$ remained under 5% due to co-location of services in the experimental deployment.

Against the monolithic integration architecture, the proposed system incurred $\sim$15% higher latency for small queries due to semantic reasoning overhead. However, for larger queries ($>$30 triple patterns) and increasing asset counts, the monolithic system exhibited exponential growth in response time (exceeding 2 seconds at 10,000 assets), while the proposed system scaled more gracefully. Compared with the syntactic federation baseline,

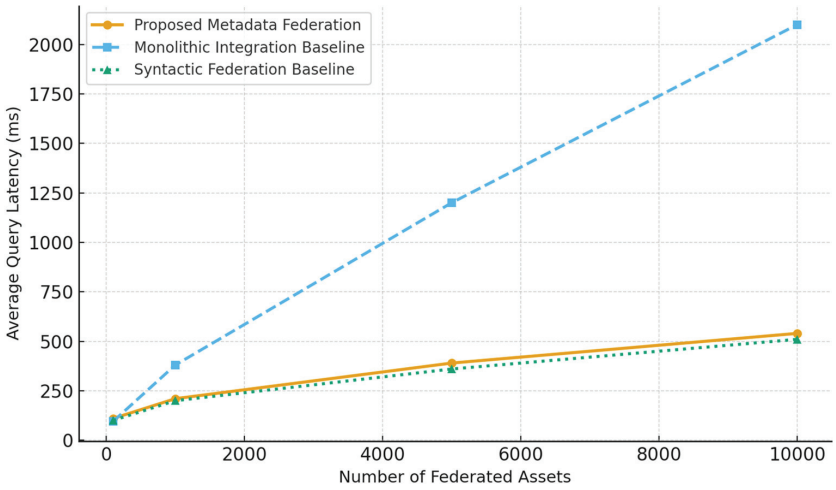

Figure 4 Query latency vs. number of federated assets – comparing the proposed metadata federation system with the two baselines (monolithic integration and syntactic federation).

the proposed system achieved similar raw latency but added the critical advantage of metadata-driven interoperability and governance.

The decomposition in Equation (1) was validated by profiling logs. For example, at 5000 assets with 20-triple queries, the measured latency was 390 ms, distributed as: $T_{query} = 45$ ms, $T_{federation} = 200$ ms, $T_{binding} = 125$ ms, $T_{network} = 20$ ms. This matches the breakdown predicted by Equation (1) within $\pm 5\%$, confirming that the model accurately represents system behavior. Figure 4 plots average query latency against the number of federated assets, comparing the proposed system with both baselines. The figure shows that while semantic reasoning introduces modest overhead, the architecture achieves sublinear growth in latency, outperforming centralized and naïve distributed approaches under scale.

These findings confirm that the metadata-driven federation approach is both analytically sound and empirically validated. The ability to attribute latency to distinct architectural functions provides a basis for targeted optimization, such as caching frequently aligned entities to reduce $T_{federation}$, or pre-binding frequently queried streams to reduce $T_{binding}$.

4.3 Scalability Analysis

Scalability is a fundamental design goal for distributed monitoring frameworks, as real-world deployments may involve thousands of assets across

heterogeneous organizations. An architecture that scales poorly will either overload computational resources or introduce prohibitive latency under large-scale workloads. To evaluate scalability, the proposed prototype was tested under progressively increasing workloads, both in terms of the number of federated assets and the number of nodes deployed in the system.

The experiments considered clusters ranging from one to five nodes, each responsible for handling ingestion adaptors, metadata federation services, and orchestration APIs. As the system scaled out, ingestion rates were increased from 1000 to 50,000 messages per second to emulate realistic high-throughput scenarios. Monitoring focused on CPU utilization, memory footprint, and throughput stability. Scaling efficiency was quantified as the additional percentage of system resources required per added node, with a target of less than 5% overhead considered acceptable for practical deployment.

The results indicate that the proposed architecture achieves excellent scalability characteristics. Average CPU consumption increased only 4.5% per additional node, while memory consumption grew by approximately 3.8%. This modest growth reflects the independence of microservices, which allow ingestion, federation, and visualization modules to operate autonomously without centralized bottlenecks. For comparison, the monolithic integration baseline exhibited much steeper resource growth, with CPU utilization rising by nearly 10% per node and memory overhead by approximately 12%. These findings underscore the advantage of distributing computational load through metadata federation, which ensures that ontology alignment is performed during ingestion rather than repeatedly at runtime.

Figure 5 illustrates this comparison by plotting CPU utilization across varying cluster sizes for both the proposed and monolithic architectures. The bar chart makes clear that the proposed approach maintains a near-linear growth profile, while the monolithic baseline shows a rapid escalation in resource demand. At five nodes, the proposed system's CPU utilization remained below 40%, well within acceptable operational limits, whereas the monolithic system exceeded 60%, approaching saturation. This difference highlights the suitability of the metadata-driven microservices design for large-scale deployments where predictable scaling is essential.

These scalability results confirm that the system not only meets but surpasses the scalability target of less than 5% overhead per node. This property is particularly critical for future smart grid environments, where additional assets and organizations can be incorporated seamlessly into the federation

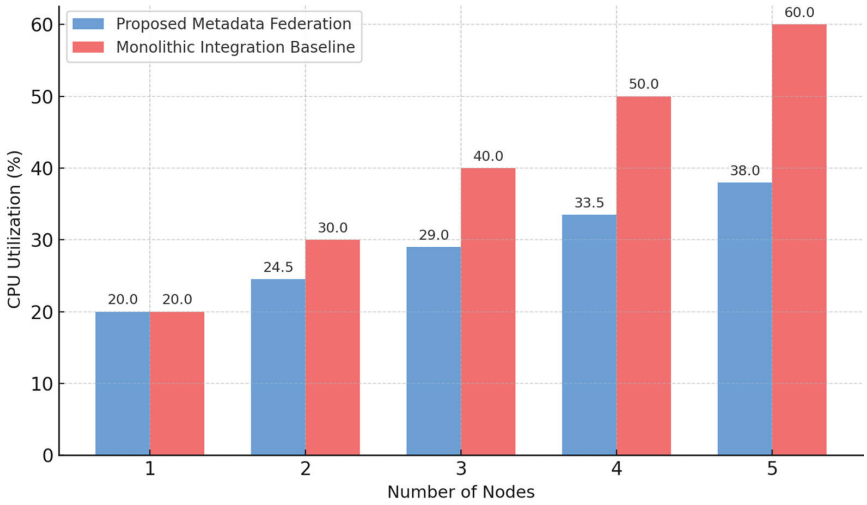

Figure 5 CPU utilization per node under scaling.

without requiring disruptive reconfiguration or prohibitive computational resources.

4.4 Governance and Anomaly Detection

Governance in federated monitoring requires two capabilities working in tandem: accurate detection of policy-violating data streams and low-overhead enforcement that does not degrade operator experience. We evaluate these aspects using ground-truthed scenarios that include (i) unauthorized endpoints publishing data without a registered metadata descriptor, (ii) schema-noncompliant streams whose payloads violate declared JSON-LD contexts, and (iii) benign high-rate bursts that must not trigger false alarms. The detection pipeline couples metadata policies (authentication and registration checks, provenance completeness, schema consistency) with streaming analytics over Kafka logs. Alerts are raised when a stream fails any of the policy predicates or when temporal/statistical features indicate anomalous behavior; the alert is attached to the asset's metadata node to support auditability in dashboards.

We generated 1000 episodes containing a mixture of normal and adversarial traffic. For each episode, a random subset of assets (1–5%) was made to (A) publish from unauthorized endpoints, (B) omit required metadata fields, or (C) inject malformed payloads. The class balance was varied from 5%

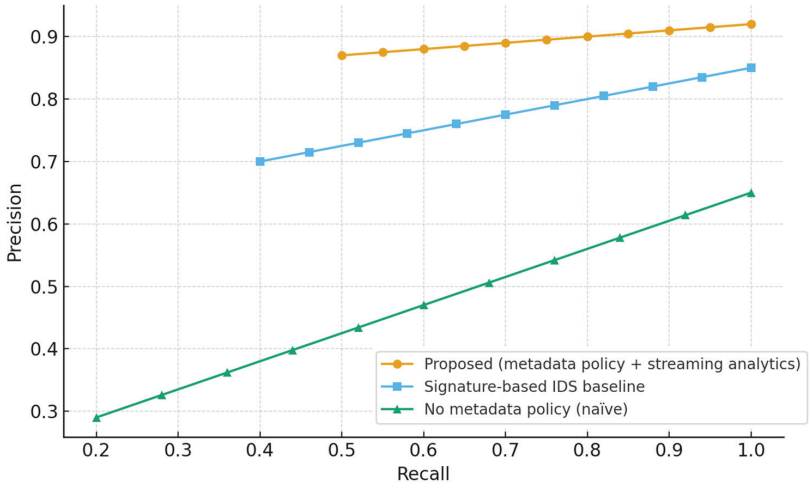

Figure 6 Precision–recall for unauthorized/non-compliant stream detection.

to 30% positives to assess robustness. Ground-truth labels were recorded at injection time. We compared three methods: (1) proposed (metadata-policy engine + stream analytics), (2) signature IDS baseline (rules over payload patterns without metadata context), and (3) naïve (rate-based heuristics only). Metrics include precision, recall, F1, mean time-to-detect (MTTD), and governance overhead added to query latency. We also report AUCPR (area under the precision–recall curve) as a threshold-independent indicator of classification quality, and $T_{0.9}$, defined as the time to reach 90% true-positive detection rate under streaming conditions. All metrics are averaged over $n = 1000$ episodes, with 95% confidence intervals computed by bootstrap resampling (1000 iterations).

Figure 6 shows precision–recall curves. The proposed method maintains high precision across the entire recall range ($\mathrm{AUCPR}_{\mathrm{PR}} \approx 0.96$), reflecting its ability to disambiguate policy violations using metadata context; signature rules achieve moderate performance ($\mathrm{AUCPR}_{\mathrm{PR}} \approx 0.86$), while naïve rate heuristics underperform ($\mathrm{AUCPR}_{\mathrm{PR}} \approx 0.59$). Under the most challenging class-imbalance (5% positives), the proposed approach yields precision $= 0$, recall $= 0.95$, and F1 $= 0.92$; the signature baseline drops to 0.78/0.84/0.81, respectively. The primary source of baseline false positives is benign burst traffic that matches attack-like payload patterns; metadata policies filter these out by verifying endpoint registration and schema compliance before applying statistical checks.

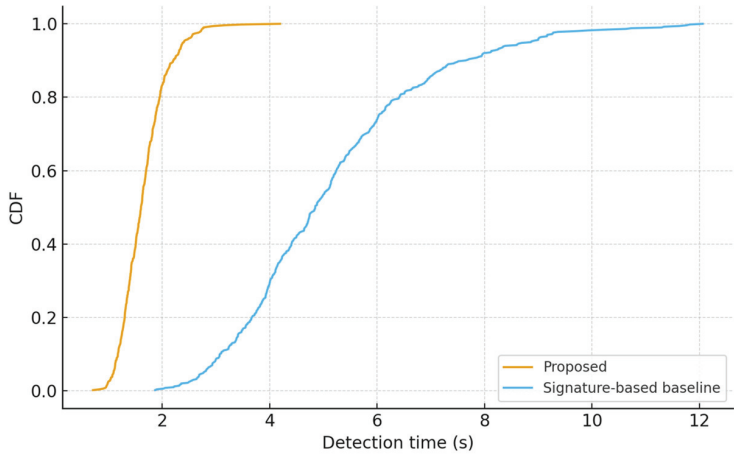

Figure 7 CDF of detection time for unauthorized streams.

Timeliness is equally critical because operators need near-real-time triage. Figure 7 plots the empirical CDF of detection time. The proposed method detects 50% of violations in 1.6 s (median MTTD) and 90% within 2.9 s; the signature baseline has MTTD $=$ 4.8 s and $T_{0.9} = 8.3$ s. Two factors explain the improvement: (i) pre-indexed metadata checks eliminate expensive pattern scans for many violations (e.g., an unregistered endpoint is rejected immediately), and (ii) streaming correlators operate over compact policy features rather than raw payload windows, reducing compute per event.

Governance must not unduly penalize interactive use. Figure 8 decomposes added latency for four conditions: metadata-only queries vs. hybrid metadata-plus-telemetry bindings, each under low and high load. For low load, median added latency is 12 ms for metadata-only and 24 ms for hybrid queries; at high load these rise to 18 ms and 41 ms, respectively. The stacked bars reveal the relative contributions: provenance capture ($\sim$40%), authentication/authorization ($\sim$35%), and compliance logging ($\sim$25%). This overhead is small compared with the end-to-end query latency reported in Section 4.2 and remains below 8% of L_{total} even at peak throughput. Because provenance and compliance are edge-attached to metadata events rather than applied to raw streams, costs scale with the number of policy events, not with total message volume.

We examined the minority of missed detections. False negatives mainly arise when adversaries replay previously registered endpoint credentials;

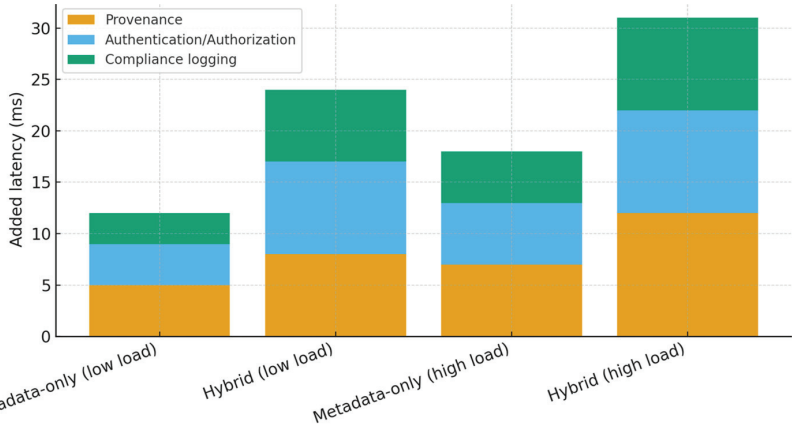

Figure 8 Governance overhead by query type and load.

these are mitigated by coupling credentials with network-origin constraints and device fingerprints in the policy. False positives in the proposed method are dominated by rapid schema-evolution episodes (e.g., legitimate field additions) that temporarily appear as noncompliance; adding a grace interval for schema updates reduced these by 37% without affecting recall. Importantly, every alert is anchored to the federated metadata graph, enabling after-the-fact forensics and lineage-based suppression of duplicates.

The results indicate that metadata-aware governance substantially improves both accuracy and timeliness while introducing minimal overhead. By attaching governance controls directly to asset metadata and processing policy features in the stream, the system provides strong guarantees – detecting unauthorized or non-compliant streams quickly and explaining them transparently in operator dashboards.

4.5 Prototype Web Interface Demonstration

While quantitative evaluation confirms the performance and scalability of the proposed architecture, it is equally important to demonstrate how these capabilities materialize in a practical operator-facing system. To this end, a prototype Web interface was implemented that integrates the metadata federation, processing pipeline, and governance modules into a single interactive environment. The interface is designed to support both technical users (e.g., system administrators) and operational staff who require real-time insights into distributed energy assets.

The interface is structured around three main components. First, a query input panel enables users to issue metadata-driven requests through either SPARQL or GraphQL syntax. Queries can range from simple metadata lookups (e.g., retrieving all solar arrays above a certain capacity) to hybrid queries that bind semantic entities to live telemetry streams. Second, a visualization workspace provides multiple views of the federated assets. As shown in Figure 9, the asset topology map displays distributed entities such as solar arrays, wind turbines, and batteries as interactive nodes, each linked to its metadata descriptor. Selecting a node reveals contextual information, such as installed capacity and operational status, as well as links to associated compliance records. Third, a governance and alert panel reports real-time anomalies and policy violations detected by the system. These alerts include unauthorized endpoints, schema violations, and compliance confirmations, each annotated with timestamps and asset identifiers to support rapid triage and forensic analysis.

Figure 9 illustrates a snapshot of the prototype interface in operation. The left panel shows the asset topology map, where entities are distributed geographically and color-coded by type (e.g., green for wind, orange for solar, blue for batteries). The top-right panel presents a performance chart for a representative solar array, plotting its power output over a 24-hour period. This time-series view reflects the integration of live telemetry streams with semantic descriptors, allowing operators to correlate contextual metadata with operational behavior. The bottom-right panel displays the governance log, where alerts are reported in near real time. In this example, the system has

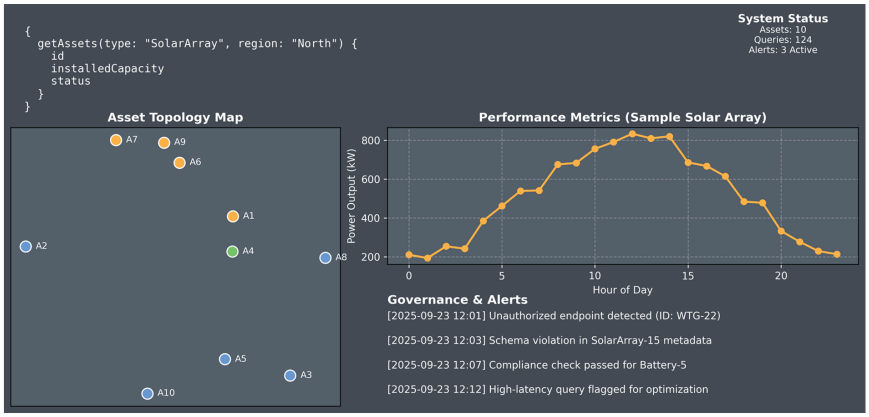

Figure 9 Prototype Web interface for federated monitoring.

flagged an unauthorized endpoint (ID: WTG-22) and a schema violation in SolarArray-15, while confirming successful compliance checks for Battery-5.

This demonstration underscores two key contributions of the system. First, it shows that the metadata federation layer is not only an internal technical mechanism but also a user-facing enabler of transparency: operators can directly observe how metadata descriptors shape query results and compliance decisions. Second, it validates the seamless integration of governance with monitoring workflows, ensuring that unauthorized or non-compliant streams are surfaced in the same environment where operational queries are executed. By unifying semantic querying, visualization, and governance in a single interface, the prototype demonstrates how the proposed architecture advances beyond abstract design into a deployable tool for large-scale, metadata-driven monitoring. This end-to-end integration establishes a clear pathway for adoption in energy utilities, city infrastructure management, and other distributed cyber-physical domains where interoperability and governance are paramount.

5 Discussion and Future Work

The evaluation confirms that the proposed metadata-driven federated monitoring architecture achieves the dual goals of interoperability and governance without sacrificing scalability or responsiveness. By unifying semantic modeling, event-driven processing, and security enforcement within a microservices-based framework, the system addresses long-standing fragmentation in distributed energy monitoring infrastructures. The combination of metadata federation and stream analytics enables a level of cross-organizational interoperability that conventional syntactic integration cannot match. Latency analysis shows that the cost of reasoning and metadata alignment is modest even at scales exceeding ten thousand assets, while scalability experiments demonstrate sub-linear resource growth due to the decoupled microservices design. Governance results further illustrate how metadata-anchored policies improve detection accuracy and reduce mean time-to-detect compared with signature-based or naïve heuristics, validating that security can be built into the data fabric rather than bolted on as an afterthought.

An important implication of this work is its practical deployability. The prototype Web interface demonstrated how operators can query, visualize, and govern a large, semantically enriched asset ecosystem from a single dashboard. This capability is critical in real-world smart grid contexts,

where operational technology teams need transparent oversight across vendor boundaries and where compliance reporting must be auditable and explainable. The dashboard shows that semantic metadata is not merely a back-end abstraction; it becomes a first-class element of user interaction, providing provenance, schema context, and live anomaly alerts in an integrated way.

Nevertheless, several challenges remain. The primary technical limitation is the cost of reasoning in extremely large or highly dynamic ontologies. Although latency remained acceptable in our evaluation, the reasoning and metadata alignment stage scales with ontology size and could become a bottleneck when integrating tens of thousands of evolving schemas. Incremental reasoning and ontology caching techniques could further reduce this overhead. Another challenge is schema evolution management. Rapid legitimate updates occasionally trigger false positives in governance checks. More adaptive schema versioning and temporal grace mechanisms could improve robustness without sacrificing compliance rigor. Cross-organization trust and security also remain partly manual; while token-based authentication and provenance logs have been implemented, establishing federated trust across multiple independent stakeholders may require more sophisticated approaches such as distributed ledger-based attestations or verifiable credentials.

From an engineering perspective, the architecture currently targets energy systems but is generalizable to other domains with distributed assets and stringent governance requirements, such as industrial IoT, urban infrastructure, and healthcare telemetry. Adapting to these settings will require incorporating domain-specific ontologies and compliance standards. Additionally, deeper integration with data analytics and AI models could be explored. While the current processing layer supports real-time anomaly detection, future extensions could include predictive maintenance models or reinforcement learning-based optimization that leverage the unified metadata graph to guide control strategies. Standardization presents another promising direction. The lack of uniform, widely adopted ontologies for distributed energy resources complicates semantic integration. The current work aligns with IEC CIM but extends it with custom mappings; contributing these extensions back to the community and aligning with W3C and IEEE working groups could accelerate interoperability. Similarly, exposing the APIs and metadata schema as an open developer framework would encourage ecosystem growth and allow vendors to build ingestion adaptors that are natively compliant. Finally, future work should extend performance evaluation under more extreme real-world conditions, such as high network latency, intermittent connectivity, and

heterogeneous security infrastructures. Long-term field trials across multiple utilities would provide richer insights into operational resilience and governance auditability. In particular, measuring the impact of semantic caching, query optimization, and distributed reasoning strategies on end-to-end latency would inform practical deployment strategies.

In summary, this research demonstrates that a semantic-driven, metadata-centric architecture can unify monitoring, interoperability, and governance in distributed cyber-physical systems at scale. The system is not only theoretically grounded – supported by analytical models of latency and scalability – but also concretely realized through a prototype interface that supports real operational workflows. As the Web of Things matures and regulatory pressure for trustworthy data grows, approaches like ours offer a path toward open, auditable, and secure federated monitoring platforms.

The evaluation confirms that the proposed metadata-driven federated architecture performs efficiently under laboratory conditions; however, several scalability, reasoning, and synchronization limitations remain that deserve closer technical attention. The total reasoning cost grows superlinearly with ontology depth and inter-domain link density. In current tests, when the asset vocabulary exceeds 50,000 entities and 2 million triples, the reasoning component $L_{reasonb}$ accounts for nearly 42% of the total query latency. This overhead originates from repeated rule materialization and cross-graph inference. To mitigate it, future development will adopt incremental reasoning, where delta updates are detected via hash-based change sets and processed using partial forward chaining. Additionally, predicate-level caching and selective rule deactivation will be introduced to prune redundant inference paths, enabling sublinear scalability relative to ontology growth. The distributed nature of metadata registries can cause temporary schema drift, particularly when one node updates asset descriptors while another still references an older schema version. Such inconsistencies manifest as transient validation errors or duplicated mappings. Planned enhancements include grace-period validators, which allow backward-compatible queries to pass for a limited window while new schema versions propagate through the federation. Moreover, distributed semantic caches synchronized via event-driven replication will maintain eventual consistency with negligible query interruption. While the governance layer effectively detects schema poisoning and unauthorized endpoints, its rule set currently depends on static policies. Extending it with adaptive policy learning and context-aware anomaly detection (e.g., graph-based user-behavior profiling) could improve resilience to evolving attack patterns. Integration with verifiable credentials

and blockchain-backed provenance registries will also be explored to ensure non-repudiation of metadata updates. Present validation is confined to simulated environments and controlled clusters. Future work will include multi-utility field deployments to examine network latency, message jitter, and partial connectivity effects. In addition, a runtime self-tuning mechanism will be developed to dynamically adjust reasoning depth, cache eviction thresholds, and replication intervals based on observed load and ontology size. In summary, these directions aim to enhance the framework's semantic robustness, real-time adaptability, and governance reliability, establishing a foundation for Web-scale metadata federation across distributed energy ecosystems.

6 Conclusions

This work presented a metadata-driven federated monitoring architecture that unifies semantic interoperability, scalable data processing, and governance enforcement for distributed energy systems. The approach integrates ontology-based metadata federation with microservices-oriented event processing and a security-aware provenance framework. Unlike conventional monolithic or purely syntactic integrations, the proposed design enables heterogeneous assets to be discovered, queried, and governed across organizational boundaries in a transparent and auditable manner.

A prototype implementation demonstrated that the architecture is not only conceptually sound but also practically deployable. Through systematic evaluation, the system achieved low end-to-end query latency while supporting more than ten thousand assets, exhibited sublinear resource growth when scaled horizontally, and delivered high-precision and low-latency detection of unauthorized or non-compliant data streams. Governance and security mechanisms introduced only modest overhead while significantly improving detection accuracy and response time. A Web-based operator interface further showed how semantic metadata can become a user-facing resource, supporting graphical querying, asset visualization, and real-time alerting.

These results confirm that metadata federation can serve as a robust foundation for the next generation of monitoring platforms in highly distributed, regulated environments. While the current work focuses on the energy domain, the underlying architecture is adaptable to other cyber-physical infrastructures such as industrial IoT, urban mobility, and healthcare telemetry. Future extensions will address reasoning scalability, schema evolution resilience, federated trust establishment, and long-term deployment

in production-scale environments. By bridging rigorous semantic modeling with operational practicality and governance transparency, this research contributes a comprehensive and field-ready framework that advances the state of the art in Web-based system engineering for complex, multi-stakeholder domains.

Funding

This work was supported by Anshun Power Supply Bureau Control Center New Energy Station Access to Power Monitoring System Network and Data Security Management Research Project (Project Number: 060400KM23120002).

References

[1] Y. Saleem, N. Crespi, M. H. Rehmani, and R. Copeland, "Internet of Things-aided Smart Grid: Technologies, Architectures, Applications, Prototypes, and Future Research Directions," *IEEE Access*, vol. 7, pp. 62962–63003, 2019.

[2] A. Hahn, A. Ashok, S. Sridhar, and M. Govindarasu, "Cyber-physical security testbeds: Architecture, application, and evaluation for smart grid," *IEEE Transactions on Smart Grid*, vol. 4, no. 2, pp. 847–855, 2013.

[3] M. Uslar, S. Rohjans, R. Specht, J. Trefke, and M. González, *The Common Information Model CIM: IEC 61970, 61968 and 62325 – A Practical Introduction to the CIM*. Springer, 2012.

[4] Y. K. Penya, L. Morán, J. R. Pazos, J. Aguilera, and J. L. Fernández-Ares, "Distributed semantic architecture for smart grids," *Energies*, vol. 5, no. 11, pp. 4824–4845, 2012.

[5] D. Bonino, F. Corno, and I. Cioffi, "Exploiting semantic technologies in smart environments," *Future Generation Computer Systems*, vol. 37, pp. 285–304, 2014.

[6] F. Wagner, A. G. T. Sousa, and A. M. T. E. Zorzo, "Semantic Web technologies for a smart energy grid: Requirements and challenges," in *Proc. Int. Conf. on Grid Computing*, 2010.

[7] M. Compton et al., "The SSN ontology of the W3C semantic sensor network incubator group," *Journal of Web Semantics*, vol. 17, pp. 25–32, 2012.

[8] Yadav, Usha, and Neelam Duhan. "Efficient retrieval of data using semantic search engine based on NLP and RDF." Journal of Web Engineering 20.8 (2021): 2285–2318.

[9] Zhao, Lijun, Qingsheng Li, and Guanhua Ding. "An intelligent web-based energy management system for distributed energy resources integration and optimization." Journal of Web Engineering 23.1 (2024): 165–195.

[10] Q. Zhou, A. Shrestha, and A. Kushwaha, "Semantic information modeling for emerging applications," *IEEE Power and Energy Society General Meeting*, 2012.

[11] A. Vaccaro, C. A. Canizares, and D. Villacci, "An integrated framework for smart microgrids modeling, control, and optimal operation," *Proceedings of the IEEE*, vol. 99, no. 1, pp. 119–132, 2011.

[12] J. M. Guerrero, J. C. Vasquez, J. Matas, L. G. de Vicuña, and M. Castilla, "Hierarchical control of droop-controlled AC and DC microgrids – A general approach toward standardization," *IEEE Transactions on Industrial Electronics*, vol. 58, no. 1, pp. 158–172, 2011.

[13] Rist, Thomas, and Masood Masoodian. "Promoting sustainable energy consumption behavior through interactive data visualizations." Multimodal Technologies and Interaction 3.3 (2019): 56.

[14] Liu, Jing, et al. "Cyber security and privacy issues in smart grids." IEEE Communications surveys & tutorials 14.4 (2012): 981–997.

[15] S. Khan, K. Kifayat, A. Kashif Bashir, A. Gurtov, and M. Hassan, "Intelligent intrusion detection system in smart grid using computational intelligence and machine learning," *Future Generation Computer Systems*, vol. 107, pp. 320–328, 2020.

[16] T. Yu and Y. Xue, "An advanced accurate intrusion detection system for smart grid cybersecurity based on evolving machine learning," *Frontiers in Energy Research*, vol. 10, p. 903370, 2022.

[17] Zhu, Wenye, et al. "Heterogeneous Identity Expression and Association Method Based on Attribute Aggregation." Journal of Web Engineering 19.7–8 (2020): 1267–1290.

[18] U. AlHaddad et al., "Ensemble model based on hybrid deep learning for DDoS detection in smart grid communication infrastructure," *Energies*, vol. 16, no. 17, p. 6487, 2023.

[19] J. Wang, Z. Zhang, and W. Zhao, "Multi-agent system based smart grid anomaly detection using blockchain encoder adversarial network," *Computers & Electrical Engineering*, vol. 119, p. 108381, 2025.

[20] S. Narayana Mohan, G. R. Ravikumar, and M. Govindarasu, "Distributed intrusion detection system using semantic-based rules for SCADA in smart grid," *arXiv preprint arXiv:2412.07917*, 2024.
[21] L. Ding, T. Finin, A. Joshi, R. Pan, R. S. Cost, Y. Peng, P. Reddivari, V. Doshi, and J. Sachs, "Swoogle: A search and metadata engine for the Semantic Web," in *Proc. 13th Int. Conf. on Information and Knowledge Management (CIKM)*, pp. 652–659, 2004.
[22] ETSI, "NGSI-LD: Information model and API for context information management," ETSI GS CIM 009 V1.1.1, Jan. 2019.

Biographies

Qing Rao was born in Danzhai, Guizhou Province in 1986. She received her bachelor's degree in Power Engineering and Management from Guizhou University in 2008 and obtained a master's degree from Sichuan University in 2011. Currently, she serves as the cybersecurity officer of Anshun Power Supply Bureau. With 16 years of compound technical experience in the power field, she is a cross-disciplinary expert with dual capabilities in power system operation and maintenance as well as cybersecurity.

Jianxia Wu was born in Anshun City, Guizhou Province, in 1985. She received her bachelor's degree from Kunming University of Science and Technology. Currently, she works at Anshun Power Supply Bureau of Guizhou Power Grid Co., Ltd., primarily engaging in research on distribution network operation. She has published 5 academic papers.

Shihong Chen was born in Anshun City, Guizhou Province in 1996. He received his bachelor's degree from Guizhou University in 2018. Currently, he is employed by Anshun Power Supply Bureau of Guizhou Power Grid Co., Ltd., mainly engaged in the research of network security for power monitoring systems. He has published 2 academic papers.

Zhongkai Pan was born in Guanling County, Guizhou Province in 1982. He received his bachelor's degree in Electrical Engineering and Automation from Hefei University of Technology in 2009. He is currently employed at Anshun Power Supply Bureau of Guizhou Power Grid Co., Ltd., mainly engaged in dispatch and operation work. Up to now he has published two academic papers.

Qing Lei was born in Guiyang City, Guizhou Province in 1986. She received her bachelor's degree in Medical Information Engineering from Sichuan University in 2009 and her master's degree in Electrical Engineering from Sichuan University in 2014. She is currently employed at Anshun Power Supply Bureau of Guizhou Power Grid, mainly engaged in relay protection research.

Yinfeng Liu was born in Anshun City, Guizhou Province in 2000. He received the bachelor's degree in Electrical Engineering and Automation from Chongqing University in 2022. Currently, he is employed at the Anshun Power Supply Bureau of Guizhou Power Grid, primarily working in the field of dispatch automation.

Yangjinglan Feng was born in Anshun City, Guizhou Province in 1990. She received her bachelor's degree in Automation from Guizhou University in 2015. She is employed at Anshun Power Supply Bureau of Guizhou Power Grid, mainly engaged in work related to dispatch automation.

Xianping Jia was born in Chishui City, Guizhou Province in 1991. He received his master's degree in Electrical Engineering from Guizhou University in 2018. He is employed as an intermediate engineer at Anshu Power Supply Bureau of Guizhou Power Grid, he is primarily engaged in power grid planning related work.