

Siren Federate: Bridging Document, Relational, and Graph Models for Exploratory Graph Analysis*

Extended Abstract

Georgeta Bordea¹, Stéphane Campinas², Matteo Catena² and Renaud Delbru²

¹*L3i, La Rochelle University – La Rochelle, France*

²*Siren – Galway, Ireland*

Keywords

Exploratory Graph Analysis, Knowledge Graph, Database and Information System Architecture, Distributed Join Algorithms, Document-oriented Database

Investigative intelligence workflows – spanning domains such as law enforcement, cybersecurity, financial compliance, and investigative journalism – require analysts to iteratively explore and correlate large, heterogeneous datasets, often represented as knowledge graphs (KGs) that integrate structured records, semi-structured logs, and unstructured content like text or multimedia. Analysts typically begin investigations with only partial clues – such as a name, phone number, or suspicious transaction – and must follow complex, multi-hop connections to uncover relevant entities and relationships. This exploratory process involves issuing tens or hundreds of queries, making low-latency responsiveness essential for preserving cognitive flow and enabling rapid hypothesis testing. However, existing graph and relational database systems struggle to support such interactive analysis at scale, especially over massive graphs containing billions of entities and relations. As a result, even modest delays can compound and render real-world investigations slow, shallow, or infeasible.

To address these challenges, we introduce Siren Federate, a system designed to enable interactive, low-latency exploration of multi-modal knowledge graphs by integrating relational and graph processing capabilities directly into document-oriented databases such as Elasticsearch. By bridging document, relational, and graph data models within a unified system, Siren Federate supports search, filtering, and multi-hop path traversal – allowing analysts to execute complex, iterative queries with sub-second to second response times, even at the scale of billions of entities and relations.

To achieve the scalability and low-latency required by investigative intelligence workloads, Siren Federate incorporates several key architectural innovations. First, it implements distributed join algorithms optimized for Elasticsearch’s log-structured, shard-based architecture. These enable efficient execution of relational operations across distributed datasets by minimizing data movement across the network. Second, it supports columnar, off-heap, in-memory data processing with late materialization and morsel-driven parallelism, which improves CPU cache locality and memory management.

Siren Federate further includes a cost-based, adaptive query planner (AQP) that interleaves query planning and execution in stages. This planner selects the most efficient join strategy at runtime, based on statistics collected during previous stages. Query plan folding merges semantically equivalent operators within a query plan to eliminate redundancy, while semantic caching stores compact bitset representations of semi-join outputs for reuse across iterative queries.

IIR2025: 15th Italian Information Retrieval Workshop, 3th - 5th September 2025, Cagliari, Italy

*This work is an extended abstract based on the paper “Siren Federate: Bridging document, relational, and graph models for exploratory graph analysis” [1] which has been accepted for publication by the Computer Science and Information Systems journal (ComSIS). That is an extended version of the paper “Siren Federate: Bridging the Gap between Document and Relational Data Systems for Efficient Exploratory Graph Analysis” [2] which has appeared in the proceedings of the 28th International Symposium on Database Engineered Applications (IDEAS 2024).

✉ georgeta.bordea@univ-lr.fr (G. Bordea); stephane.campinas@siren.io (S. Campinas); matteo.catena@siren.io (M. Catena); renaud.delbru@siren.io (R. Delbru)

0000-0001-9921-8234 (G. Bordea); 0009-0008-3058-5376 (S. Campinas); 0000-0002-5571-8269 (M. Catena); 0009-0007-7759-693X (R. Delbru)



© 2025 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

A central contribution in graph query processing is the Semi-Join Decomposition (SJD) technique [3]. SJD mitigates the combinatorial explosion of intermediate results by decomposing multi-hop path-finding queries into multiple semi-joins. This reduces memory usage and computational overhead, thereby enhancing scalability and efficiency when working with large graphs. While applicable to general multi-hop path queries, SJD is especially effective for all-shortest-paths problems, offering a practical solution to the challenges faced by alternative methods. Its integration with Siren Federate’s adaptive query planner and semantic caching further boosts its efficiency for exploratory graph analysis.

We experimentally evaluated Siren Federate to assess its efficiency across different scenarios. In a first series of experiments, we used a synthetic dataset comprising ~15 billion of cell phone location records – a common data source in investigative contexts – to demonstrate the system’s scalability with large data volumes in a distributed environment. With these experiments, we also validated our system’s capability to process semi-joins with sub-second to second response times. This result is important since semi-joins are fundamental for exploratory graph analysis, as they enable operations like set-to-set navigation, graph expansion, and pathfinding.

In a second series of experiments, we employed the LDBC Financial Benchmark [4], which models financial industry data and workloads. The dataset used contains ~5 million entities (e.g., person, accounts) and ~26 million relations (e.g., money transfer), while the queries used required matching graph patterns of varying complexity and were expressed using the standard Graph Query Language (GQL) [5], which is supported by Federate. The experimental results demonstrate our system capability’s to handle complex graph querying pattern and its ability to process these queries within seconds.

Finally, a real-world deployment at Apollo.io¹ confirms Siren Federate’s capacity to operate at scale, supporting a 350-node cluster managing nearly half a petabyte of data and multiple concurrent users. The system reduced average query response times from 7 seconds to sub-second, while significantly improving cluster stability and reducing query failures. These findings demonstrated the applicability and robustness of our system in a large, highly-concurrent, production environment.

Declaration on Generative AI

During the preparation of this submission, the authors used ChatGPT to improve the writing in parts of the text and to check grammar and spelling. After using this service, the authors reviewed and edited the text as needed and take full responsibility for the publication’s content.

References

- [1] G. Bordea, S. Campinas, M. Catena, R. Delbru, Siren Federate: Bridging document, relational, and graph models for exploratory graph analysis, arXiv preprint arXiv:2504.07815 (2025).
- [2] S. Campinas, M. Catena, R. Delbru, Siren Federate: Bridging the Gap Between Document and Relational Data Systems for Efficient Exploratory Graph Analysis, in: Proc. IDEAS, 2024.
- [3] D. Pini, G. Tummarello, R. Delbru, Optimization of Database Sequence of Joins for Reachability and Shortest Path Determination. U.S. Patent 11720564, 2022.
- [4] S. Qi, H. Lin, Z. Guo, G. Szárnyas, B. Tong, Y. Zhou, B. Yang, J. Zhang, Z. Wang, Y. Shen, et al., The LDBC Financial Benchmark, arXiv preprint arXiv:2306.15975 (2023).
- [5] A. Deutsch, N. Francis, A. Green, K. Hare, B. Li, L. Libkin, T. Lindaaker, V. Marsault, W. Martens, J. Michels, et al., Graph Pattern Matching in GQL and SQL/PGQ, in: Proc. SIGMOD/PODS, 2022.

¹<https://siren.io/case-study-transforming-enterprise-search-at-apollo-with-siren-federate/>