OPEN
MAINFRAME
PROJECT

Modernization Working Group

RAG to Riches

Understanding Actionable Insights from Mainframe Data & AI

Authors:
Advith Krishnan
Dr. Vinu Russell N. Viswasadhas

v1 (Oct 2025)

Executive Summary

Mainframes remain unmatched in reliability, transactional throughput, and zero-downtime performance, making them the backbone of enterprise-scale operations even in cloud-native environments. They hold vast amounts of both business and platform operational data that can power root-cause analysis and trends summarization. However, this data often remains locked away, fragmented across SMF records, IMS transaction logs, DB2 datasets, and vendor-specific tools, making it difficult to integrate into modern analytics workflows or interpret without deep platform expertise [8][9].

This whitepaper introduces RAG to Riches, an explainable infrastructure framework that unifies business and operational mainframe data into a single, accessible intelligence layer. By applying Retrieval-Augmented Generation (RAG) and Large Language Models (LLMs), the framework enables data to be queried in natural language, interpreted in context, and transformed into actionable insights. RAG to Riches bridges structured logs, time-series records, and modern observability tools using a layered architecture featuring database connectors and a dual-stage retrieval pipeline. By treating mainframes as high-value data engines, the framework enriches intelligent automation, accelerates incident resolution, and drives business value.

To ensure ethical and compliant usage, the RAG to Riches framework is strictly focused on operational and business telemetry, maintaining both privacy and regulatory alignment. **Utilizing this framework in combination with data systems containing PID or other sensitive customer information is strongly discouraged.**

The Hidden Cost of Data Invisibility

For decades, mainframes have powered the world's most critical industries, delivering rock-solid reliability that has allowed businesses to operate at scale without disruption. Their transactional precision and resilience remain unmatched [8]. IBM's report shows that mainframes still handle almost 70% of the world's production IT workloads, living up to their nickname "Big Iron." The report's survey further states that 90% of IT and business executives view their mainframe as a growth platform, with more than half reporting an increase in transaction volumes over the year [14]. Yet, while the systems themselves perform flawlessly, the data they generate goes underutilized.

Accessing this data typically requires specialized tools and deep platform knowledge, making it difficult for teams outside the mainframe domain to retrieve or interpret. For example, operational signals may be buried in SMF (System Management Facility) records, showing workload anomalies or transaction latencies that directly affect customer experience, service-level agreements, or compliance obligations [9].

As enterprises adopt hybrid architectures, the ability to bring business records and operational signals together in context becomes essential. Business data shows what happened; operational data explains why it happened. By connecting these perspectives, organizations can anticipate system disruptions, trace complex issues across environments, and link platform health to business impact [12][13]. To unlock this capability, we introduce the RAG to Riches framework.

Mainframe Data as a Business Enabler

RAG to Riches embeds mainframe data into the same decision-making fabric as modern cloud systems. Instead of replacing mainframe workloads, the framework builds a low-latency, zero-downtime data bridge that allows both platform operational data (logs, SMF records, workload metrics) and business operational data (transactions, user activity, audit trails) to be queried in natural language [8][10][11]. Structured logs, time-series metrics, and transactional records are ingested continuously and indexed in a dual-stage retrieval system: one stage optimized for speed and keyword precision, the other for semantic depth and context-aware search [1][2].

This design enables queries that go beyond raw metrics, offering synthesized and context-rich answers. Engineers investigating system anomalies can trace them across systems without switching tools or relying solely on domain-specific commands. Executives can request summaries, capacity forecasts, or compliance audit readiness checks in plain language, supported by the same high-fidelity data [12].

A common concern is whether an LLM can reliably interpret both business and platform data without conflating them [5][7]. RAG to Riches addresses this by separating retrieval paths: platform telemetry is indexed and retrieved distinctly from business datasets, with schema-aware connectors and role-based access ensuring proper contextualization before reaching the model [5][6].

When queries require correlation across both domains, the framework unifies the results, enabling cross-domain reasoning without losing precision [4][3].

The system's modular connector layer supports IMS, DB2, PostgreSQL, Splunk, and other integration points, providing a consistent foundation for analytics without introducing fragility or downtime [8][13].

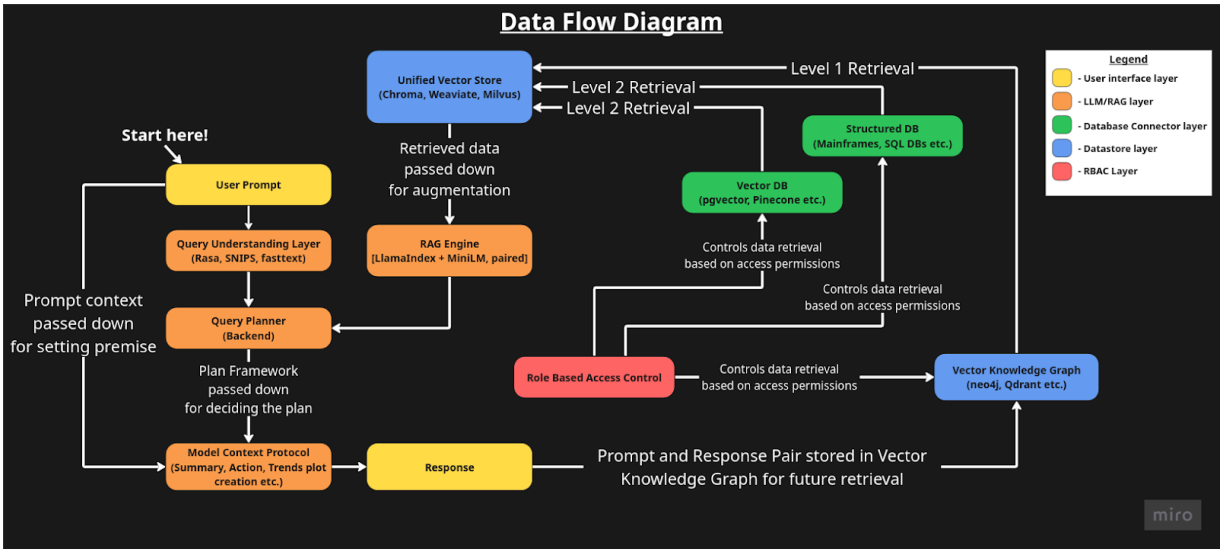

Fig 1) RAG to Riches Framework Data Flow Diagram

The architecture begins with the Query Understanding Layer (QUL). This layer processes the raw prompt to extract the user's intent, identify relevant entities such as system names or timestamps, and classify the prompt type. Once interpreted, the structured representation is sent to the Query Planner, which determines the optimal execution path. For retrieval-based queries, the planner routes requests to the RAG Engine, while command-style or previously answered queries flow directly into the Model Context Protocol (MCP) for execution of relevant tools.

The RAG Engine, built on LlamaIndex and MiniLM, operates on two levels. At the first level, it performs a fast lookup against the Vector Knowledge Graph (VKG) to identify similar past queries or related subgraphs. If this yields insufficient results, the engine escalates to Level 2 Retrieval, which queries structured databases, such as mainframes or PostgreSQL, for transactional

and time-series data, as well as vector databases like Pinecone or pgvector for semantic search over logs and telemetry.

Retrieved data is filtered through access controls and summarized before use. For example, when faced with a query such as *"**Show me what happened during last Friday's login failure,**"* the engine may extract log entries from the vector store and correlate them with records from connected systems. The explanation might be: *"The login failure on Friday was due to an LDAP timeout at 10:43 AM, and failover did not occur because of a misconfiguration in the fallback server."* This highlights a core principle: while mainframe data is essential, meaningful insights often require correlating it with surrounding enterprise systems, such as identity providers, middleware, or observability platforms.

RAG to Riches enables this integration through modular connectors, ensuring that natural language queries return end-to-end insights rather than isolated fragments. Once the necessary context is assembled, the MCP integrates it with the original prompt, applies higher-order logic, executes actions, and generates the final response. The entire interaction, including prompt, response, and supporting evidence, is recorded in the VKG. This feedback loop ensures repeated queries can be resolved faster, enabling continuous learning and compounding intelligence over time.

Recommendations for Technology Leaders

To capture this opportunity, technology leaders should prioritize making existing business and operational data more accessible and ensuring it can be interpreted quickly. The RAG to Riches framework provides a practical way to achieve this: modular connectors unify mainframe and related datasets with modern observability tools, making them queryable in natural language.

The value lies not just in visibility but in the ability to extract relevant insights faster, whether by business leaders seeking summaries and forecasts or by operations teams diagnosing incidents. With RAG to Riches, queries return actionable answers rather than raw dumps, reducing reliance on specialized expertise. In business terms, the impact is clear: reduced downtime, faster incident resolution, improved customer experience, and accelerated compliance reporting. By aligning execution with business objectives, organizations transform mainframe data into a visible and measurable driver of performance.

Conclusion

With the right architecture, mainframe data can power the next generation of AI-driven business intelligence. RAG to Riches provides a framework that bridges business and operational data, making it accessible and interpretable through natural language queries. By unifying these two domains into a single retrieval and reasoning layer, organizations can trace events end-to-end, correlate causes with outcomes, and build confidence in the evidence behind each answer.

This capability enables faster decisions, stronger resilience, and greater innovation across mainframe and cloud environments. RAG to Riches is not about changing the core strengths of the mainframe, but about unlocking the interpretability of its data and turning it into a driver of actionable insight.

References

[1] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, et al., “Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks,” arXiv preprint arXiv:2005.11401, 2020.

[2] X. Cheng, J. Zhang, and X. Liu, “A Comprehensive Survey of Retrieval-Augmented Generation,” arXiv preprint arXiv:2410.12837, 2025.

[3] Z. Huang, et al., “A Survey on Knowledge-Oriented Retrieval-Augmented Generation,” arXiv preprint arXiv:2503.10677, 2025.

[4] L. Zhang and S. Kim, “Astute RAG: Overcoming Imperfect Retrieval Augmentation and Knowledge Conflicts,” arXiv preprint arXiv:2410.07176, 2024.

[5] Y. Zhao and T. Lin, “OG-RAG: Ontology-Grounded Retrieval-Augmented Generation,” arXiv preprint arXiv:2412.15235, 2024.

[6] M. Patel and R. Singh, “Retrieval-Augmented Generation with Hierarchical Knowledge (HiRAG),” arXiv preprint arXiv:2503.10150, 2025.

[7] W. Chen and F. Müller, “Retrieval-Augmented Generation with Knowledge Graphs,” arXiv preprint arXiv:2404.17723, 2025.

[8] IBM Corporation, IBM Z Integration for Observability, IBM Documentation, 2024. [Online]. Available: https://www.ibm.com/products/z-integration-observability

[9] Precisely Software Inc., Mainframe and IBM i Observability: Why It Matters, 2024. [Online]. Available: https://www.precisely.com/mainframe/mainframe-and-ibm-i-observability

[10] SHARE Association, “OpenTelemetry: Enabling Mainframe Insights for Hybrid Observability,” SHARE Technical Blog, 2024. [Online]. Available: https://blog.share.org/Article/opentelemetry-enabling-mainframe-insights

[11] K. Wähner, “Mainframe Integration with Data Streaming: Architecture, Business Value, and Real-World Success,” 2025. [Online]. Available: https://www.kai-waehner.de/blog/2025/06/13/mainframe-integration-with-data-streaming

[12] The Futurum Group, Achieving Mainframe Resilience by Combining System Monitoring and Observability, Futurum Research Brief, 2025. [Online]. Available: https://futurumgroup.com/research-reports/achieving-mainframe-resilience

[13] Open Mainframe Project, Hybrid Integration Patterns for Mainframe and Cloud Systems, Linux Foundation, 2024. [Online]. Available: https://www.openmainframeproject.org

[14] J. Granger, A. Sharma, A. Marshall, and S. Soman, Application Modernization on the Mainframe: Expanding the Value of Cloud Transformation, IBM Corporation, 2021. [Online]. Available: https://www.ibm.com/downloads/cas/7BJPNGND

Authors

Advith Krishnan (advithkrishnan@gmail.com)

Mentee, Open Mainframe Project

Dr. Vinu Russell N. Viswasadhas (vinu.viswasadhas@kyndryl.com)

Associate Director - Data Consulting, Kyndryl

Mentor, Open Mainframe Project

Acknowledgements

This document was developed under the Open Mainframe Project's Summer 2025 Mentorship program and sponsored by the MMWG (Mainframe Modernization Working Group). We are incredibly grateful for their support in facilitating our work. We would also like to thank Mr. Bruno Azenha, MMWG mentor, and Ms. Amrutha Rajshekar, MMWG mentee, for their helpful feedback, which greatly shaped the content of this white paper. We would also like to thank Mr. Steven Dickens, CEO & Principal Analyst at HyperFRAME Research, for his valuable insights that aided the focus and direction of this paper. We hope this white paper will serve as a useful reference document for developers, project managers, and decision-makers looking to drive business value with AI and mainframes.

About the Modernization Working Group

The Modernization Working Group, part of the Open Mainframe Project, serves as a focal point for thought leadership on what it means to modernize mainframe applications. We are an open group passionate about knowledge sharing, leading constructive discussions, challenging the status quo, exploring creative solutions, and generating content that helps the community on their journey. Learn more about the working group at https://openmainframeproject.org/our-projects/working-groups/modernization-working-group/