ISBN: 978-3-95450-262-2 ISSN: 2673-5350 doi: 10.18429/JACoW-IBIC2025-MOPMO13

INTELLIGENT RETRIEVAL SYSTEMS FOR ACCELERATOR PHYSICS: MATCHING RETRIEVAL APPROACHES TO USE CASES

R. Ischebeck*, M. Sapinski, L. Stuhlmann,
PSI Center for Accelerator Science and Engineering, Villigen, Switzerland
Q. Dai, University of Zurich, Zurich, Switzerland

Abstract

Accelerator facilities generate diverse documentation, from technical reports to structured wikis and semi-structured logbooks, which complicates efficient knowledge access. While Retrieval-Augmented Generation (RAG) offers a path toward intelligent operator assistants, no single method is universally optimal. We present three use cases from PSI: for technical documentation, naive dense retrieval with summarization provides fast and interpretable access; for the AcceleratorWiki, a graph-augmented approach improves reasoning over hierarchies and cross-references; and for ELOG, an agentic pipeline with specialized agents supports multimodal interpretation, temporal reasoning, and iterative refinement. Together, these case studies illustrate how matching retrieval paradigms to data types enables reliable, context-aware assistance in accelerator operations.

INTRODUCTION

Large Language Models (LLMs) have demonstrated impressive performance across a broad array of tasks, including question answering, code generation and debugging, multilingual translation or summarization. Yet, they often suffer from factual errors, limited domain coverage, and poor verifiability. A key issue is hallucination [1], where models generate plausible but incorrect statements due to their probabilistic nature. In scientific and technical contexts, such errors are particularly problematic, as they can result in misleading or harmful conclusions. To address these challenges, Retrieval-Augmented Generation (RAG), proposed by [2], enhances LLM outputs by incorporating external knowledge at inference time, retrieving relevant documents from trusted sources to improve accuracy, reliability, and transparency.

In a standard RAG system, as shown in Fig. 1, a retriever first selects relevant passages from a document collection, which are then provided to the LLM to ground its response. This mechanism reduces hallucinations, allows answers to be verified against the source material, and enables integration of continuously updated information.

Recent surveys highlight the rapid evolution of RAG, open challenges such as retrieval efficiency, hybrid retrievers, and multimodal integration, and the importance of tailoring the pipeline to the target domain [3]. In this work, we extend these ideas to accelerator physics, showing that different documentation sources—technical reports, internal wikis, and logbooks—require distinct retrieval paradigms. Through three case studies, we illustrate when naive dense retrieval suffices, when lightweight knowledge graphs add value, and when agentic RAG pipelines become necessary for complex scenarios. Furthermore, all systems described in this work are powered by locally deployed LLMs, ensuring data privacy, reducing latency, and avoiding reliance on external APIs.

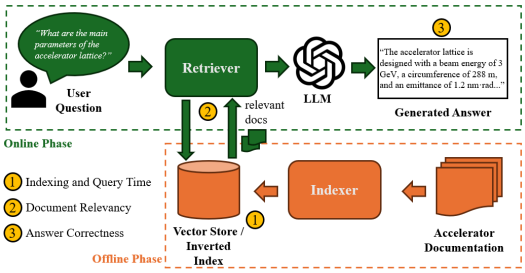

Figure 1: Example naive RAG pipeline [4].

RELATED WORK

Efficient access to accelerator documentation and electronic logbooks (eLogs) has long been a challenge due to their heterogeneous and often unstructured content. Recent work has explored AI-driven solutions to enhance information retrieval in these contexts. Sulc et al. proposed AI pipelines for accelerator logbooks, highlighting opportunities for hybrid retrieval and multimodal analysis across facilities such as Fermilab, Jefferson Lab, and SLAC [5, 6]. Complementary approaches have introduced general AI assistants for accelerator operations, combining RAG with reasoning and action frameworks to interface directly with control systems [7].

Together, these studies underline the importance of tailoring retrieval pipelines to accelerator-specific contexts. Building on these insights, our work investigates when naive dense retrieval suffices, when knowledge-graph-enhanced pipelines are advantageous, and when agentic multimodal RAG becomes necessary.

USE CASES AT PSI

Technical Documentations

The High Intensity Proton Accelerator (HIPA) at PSI was built over 50 years ago, at a time when digital documentation was largely absent [8]. Over decades of experiments and continuous development, the facility has grown significantly, but without a consistent long-term record. As a result, any new specialist taking over a system typically requires several years to reach expert-level performance. To address this

* rasmus.ischebeck@psi.ch

ISBN: 978-3-95450-262-2 ISSN: 2673-5350 doi: 10.18429/JACoW-IBIC2025-MOPMO13

challenge, we collected the existing documentation—HIPA and Proscan technical reports in PDF form, two master's theses, relevant conference proceedings, and books on beam instrumentation—amounting to 1,032 pages in English and 239 pages in German. This corpus served as the starting point for our experiments with a Retrieval-Augmented Generation (RAG)–based chatbot designed to support technical problem solving.

AcceleratorWiki

The AcceleratorWiki contains structured internal documentation for HIPA, PROSCAN, SwissFEL, and SLS. Its underlying data consists of hierarchically organized articles enriched with figures, tables, and cross-references, covering both theoretical background and detailed procedures. In daily operations, it is used to consult safety protocols, calibration instructions, and subsystem explanations, making it a critical resource for troubleshooting and training. However, the layered structure and extensive cross-linking create difficulties: operators must often move from general concepts to specific procedures or combine information spread across accelerators. The presence of mixed modalities—text, tables, and diagrams—further complicates retrieval. A naive RAG approach based solely on dense similarity search cannot address these challenges, as it neglects hierarchy, provenance, and multimodal context. Instead, a graph-augmented retrieval strategy is required to preserve structure and enable reliable, context-aware access.

ELOG System

The Electronic Logbook (ELOG) system records operational data for the four accelerators: SwissFEL, HIPA, SLS, and PROSCAN. Each entry contains a unique ID, title, author, timestamp, free-text body, and optional attachments such as plots or measurement data. Metadata fields further classify entries by category (e.g. Problem, Safety, Shift Handover), domain (e.g. Injector, Athos), system (e.g. RF, Diagnostics), and device-specific section identifiers (e.g. SATUN18).

Information retrieval is complicated by several factors: uneven temporal density of entries, large variability in content length and detail, inconsistent terminology for devices and subsystems, and multilingual usage (German and English). Moreover, related incidents often span multiple entries and subsystems, while critical evidence is frequently contained in multimodal attachments such as plots or screenshots. Effective retrieval also requires sensitivity to time relevance, since the usefulness of an entry depends strongly on its temporal proximity to ongoing operations. These challenges necessitate intelligent methods that can incorporate operational context, correlate related events, and interpret multimodal information to support facility operations.

SYSTEM IMPLEMENTATION

Orbit RAG

The system architecture follows the workflow as in Fig. 1. In pre-processing, we used MinerU [9] as a PDF parser to extract text, tables, and equations from the 1,271 pages of documentation. Each document was split into chunks, and we stored the resulting (chunk → embedding, file-id) tuples in a vector database, with embeddings pre-computed using multilingual BGE-M3 model [10] for efficient retrieval. At runtime, user queries are embedded using the same embedding model and matched against the vector store to retrieve the top-k semantically similar chunks. The query and retrieved chunks are then passed to the instruction-tuned gemma3:27B-it-fp16 model [11] to generate an answer. All data were processed offline, and all models were run locally with Ollama [12] on a Mac Studio M2 Ultra (192 GB unified memory). Figure 2 illustrates the frontend. The chatbot interface returns not only the answer but also a ranked list of the five most relevant files, including similarity scores and snippet previews, with clickable links for further reading. This reduces the manual effort of navigating large volumes of technical documentation.

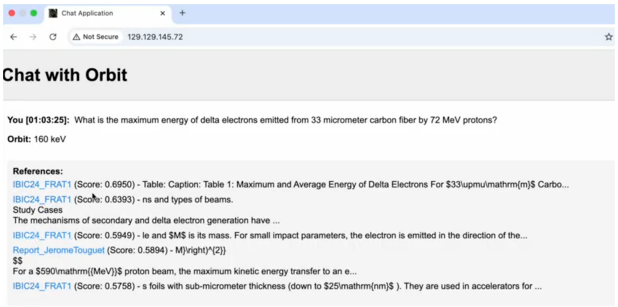

Figure 2: Query and LLM answer.

AccWiki GraphRAG

The GraphRAG system combines dense vector search, keyword matching, and hierarchical graph traversal to address the structural complexity of the AcceleratorWiki. The wiki hierarchy is preserved in a Neo4j graph [13] by modeling articles, tabs, categories, and sections as nodes with explicit parent–child relationships and cross-references. Content is chunked along section boundaries and embedded with the multilingual BGE-M3 bi-encoder model (1024 dimensions) [14] due to the German content of the AcceleratorWiki, while figures are linked directly to their sections to maintain contextual relevance. At query time, semantic search retrieves conceptually related passages, keyword search captures procedural terms (e.g. "checkliste", "prozedur"), and graph-based filtering constrains results to the appropriate accelerator or section. Retrieved passages are then contextualized with their hierarchical provenance (e.g. *HIPA* > *P-Kanal* > *Setup*) and enriched with associated figures. For multimodal reasoning, the system employs the

ISBN: 978-3-95450-262-2 ISSN: 2673-5350 doi: 10.18429/JACoW-IBIC2025-MOPMO13

Qwen2.5-VL-32B model (4-bit quantization), selected after benchmarking on an ArxivQA [15] subset focused on accelerator physics, where it demonstrated superior accuracy in multimodal question answering compared to alternative vision–language models. Through a Model Context Protocol (MCP) interface, the model can directly query and traverse the knowledge graph, enabling structured reasoning over hierarchical relationships in combination with semantic and multimodal retrieval. This hybrid setup preserves semantic relevance, organizational structure, and provenance, providing reliable, context-aware access to the wiki content.

Agentic Elog RAG

We are implementing an agentic RAG workflow in LangGraph, where specialized agents collaborate to transform natural language queries into structured analyses. As shown in Fig. 3, the workflow follows a Planner–Searcher/Analyzer–Evaluator–Reporter pattern with iterative refinement, enabling both simple lookups and complex operational assessments.

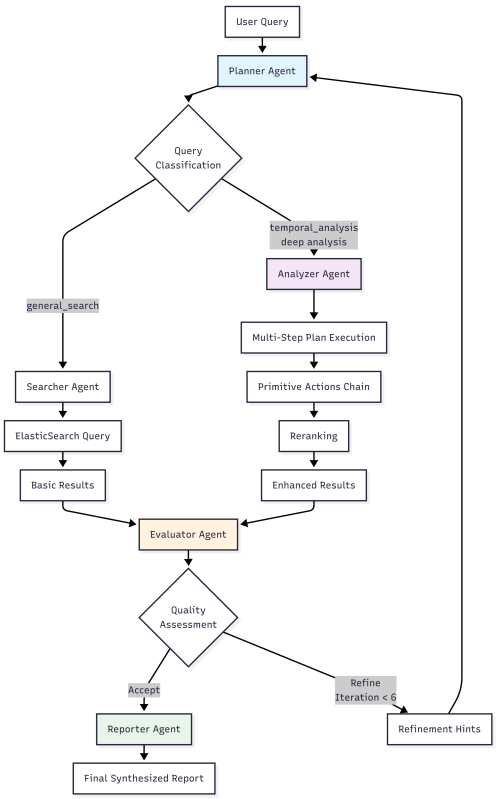

Figure 3: Workflow of the agentic RAG system for ELOG

The planner agent (Qwen3 14B [16]) interprets operator queries and generates structured search plans. A query classification step then routes requests either to the searcher agent (Qwen2.5-VL-32B) for direct ElasticSearch/Qdrant retrieval, producing basic results (left branch), or to the analyzer agent (Qwen2.5-VL-32B) for multi-step workflows such as temporal analyses or device checks (right branch). The analyzer executes chained actions, including incident extraction and reranking, to generate enhanced results. All results are passed to the evaluator agent, which assesses relevance, recency, and device specificity. If necessary, it issues refinement hints that loop back to the planner; otherwise, the reporter agent synthesizes structured outputs with contextual explanations, provenance, and multimodal evidence.

The system currently supports three query types: *general search*, *temporal analysis*, and *analyze* queries for deeper multi-step assessments. By combining BM25 retrieval, neural reranking (ms-marco-MiniLM-L2-v2 [17]), and LLM-based evaluation with heuristics for recency and diversity, the agentic pipeline (Fig. 3) delivers contextualized, synthesized insights that go beyond traditional search.

DISCUSSION

Our case studies highlight that no single retrieval paradigm is universally optimal. For unstructured technical documentation, naive RAG with dense retrieval and lightweight summarization is sufficient. For hierarchically organized internal wikis, a graph based approach is superior to preserve structure and provenance. Finally, for the complex and semi-structured ELOG data, only an agentic RAG pipeline can support temporal reasoning, multimodal evidence, and iterative refinement. Matching retrieval strategy to data type is therefore essential for reliable operator assistance.

LIMITATIONS

The systems presented here remain prototypes: not all components are fully implemented, and systematic quality evaluation is still ongoing. For Orbit RAG, we have already built a domain-specific benchmark dataset from technical documentation and carried out initial retrieval and generation assessments [8], providing a first step toward measurable progress. By contrast, comparable benchmarks for the AcceleratorWiki and ELOG systems are still missing. Developing such datasets will be essential for rigorous comparison across retrieval paradigms and for advancing reliable operator assistance in accelerator facilities.

OUTLOOK

Future work will focus on integrating multiple data sources through centralized large language models accessed via MCP servers. Such an architecture would allow unified access to technical documents, internal wikis, and logbooks, while maintaining provenance, multimodal reasoning, and domain-aware refinement in a single operator-facing assistant.

ACKNOWLEDGEMENTS

We would like to thank Adam Grycner from Google DeepMind and Lucas Fernandez Vilanova for their valuable support with setting up large language models. This project has received funding from the European Union's Horizon 2020 Research and Innovation program under Grant Agreement No 101004730 (I.FAST).

ISBN: 978-3-95450-262-2 ISSN: 2673-5350 doi: 10.18429/JACoW-IBIC2025-MOPMO13

REFERENCES

[1] Z. Ji *et al.*, "Survey of hallucination in natural language generation", *ACM Comput. Surv.*, vol. 55, no. 12, pp. 1-38, 2023. doi:10.1145/3571730

[2] P. Lewis *et al.*, "Retrieval-augmented generation for knowledge-intensive NLP tasks", *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 9459–9474, 2020, arXiv:2005.11401 [cs.CL]. doi:10.48550/arXiv.2005.11401

[3] P. Gupta *et al.*, "Retrieval-augmented generation: A comprehensive review on knowledge-enhanced large language models", Oct. 2024, arXiv:2410.12837 [cs.CL]. doi:10.48550/arXiv.2410.12837

[4] L. Stuhlmann, M. A. Saxer, and J. Fürst, "Efficient and Reproducible Biomedical Question Answering Using Retrieval Augmented Generation", in *Proc. SDS'25*, Zurich, Switzerland, Jun. 2025, pp. 154–157. doi:10.1109/sds66131.2025.00029

[5] A. Sulc *et al.*, "Towards Unlocking Insights from Logbooks Using AI", May 2024, arXiv:2406.12881 [physics.acc-ph]. doi:10.48550/arXiv.2406.12881

[6] T. Hellert *et al.*, "eLog analysis for accelerators: Status and future outlook", presented at the IPAC'25, Taipei, Taiwan, Jun. 2025, paper THPS048, unpublished.

[7] F. Mayet, "Gaia: A general ai assistant for intelligent accelerator operations", May 2024, arXiv:2405.01359 [cs.CL]. doi:10.48550/arXiv.2405.01359

[8] Q. Dai, R. Ischebeck, M. Sapinski, and A. Grycner, "Application Of Large Language Models For The Extraction Of Information From Particle Accelerator Technical Documentation", Sep. 2025, arXiv:2509.02227 [cs.IR]. doi:10.48550/arXiv.2509.02227

[9] B. Wang *et al.*, "MinerU: An Open-Source Solution for Precise Document Content Extraction", Sep. 2024, arXiv:2409.18839 [cs.CV]. doi:10.48550/arXiv.2409.18839

[10] J. Chen, S. Xiao, P. Zhang, K. Luo, D. Lian, and Z. Liu, "BGE M3-Embedding: Multi-Lingual, Multi-Functionality, Multi-Granularity Text Embeddings Through Self-Knowledge Distillation", Feb. 2024, arXiv:2402.03216 [cs.CL]. doi:10.48550/arXiv.2402.03216

[11] A. Kamath *et al.* (Gemma Team), "Gemma 3 Technical Report", Mar. 2025, arXiv:2503.19786 [cs.CL]. doi:10.48550/arXiv.2503.19786

[12] Ollama, https://ollama.com/

[13] J. Webber, "A programmatic introduction to Neo4j", in *Proc. SPLASH '12*, Tuscon, AR, USA, Oct. 2012, pp. 217–218. doi:10.1145/2384716.2384777

[14] J. Chen, S. Xiao, P. Zhang, K. Luo, D. Lian, and Z. Liu, "M3-Embedding: Multi-Linguality, Multi-Functionality, Multi-Granularity Text Embeddings Through Self-Knowledge Distillation,"in *Proc. ACL'24*, Bangkok, Thailand, 2024, pp. 2318–2335. doi:10.18653/v1/2024.findings-acl.137

[15] L. Li *et al.*, "Multimodal ArXiv: A Dataset for Improving Scientific Comprehension of Large Vision-Language Models", Mar. 2024, arXiv:2403.00231 [cs.CV]. doi:10.48550/arXiv.2403.00231

[16] A. Yang *et al.*, "Qwen3 Technical Report", May 2025, arXiv:2505.09388 [cs.CL]. doi:10.48550/arXiv.2505.09388

[17] W. Wang, F. Wei, L. Dong, H. Bao, N. Yang, and M. Zhou, "MiniLM: Deep Self-Attention Distillation for Task-Agnostic Compression of Pre-Trained Transformers", Feb. 2020, arXiv:2002.10957 [cs.CL]. doi:10.48550/arXiv.2002.10957