

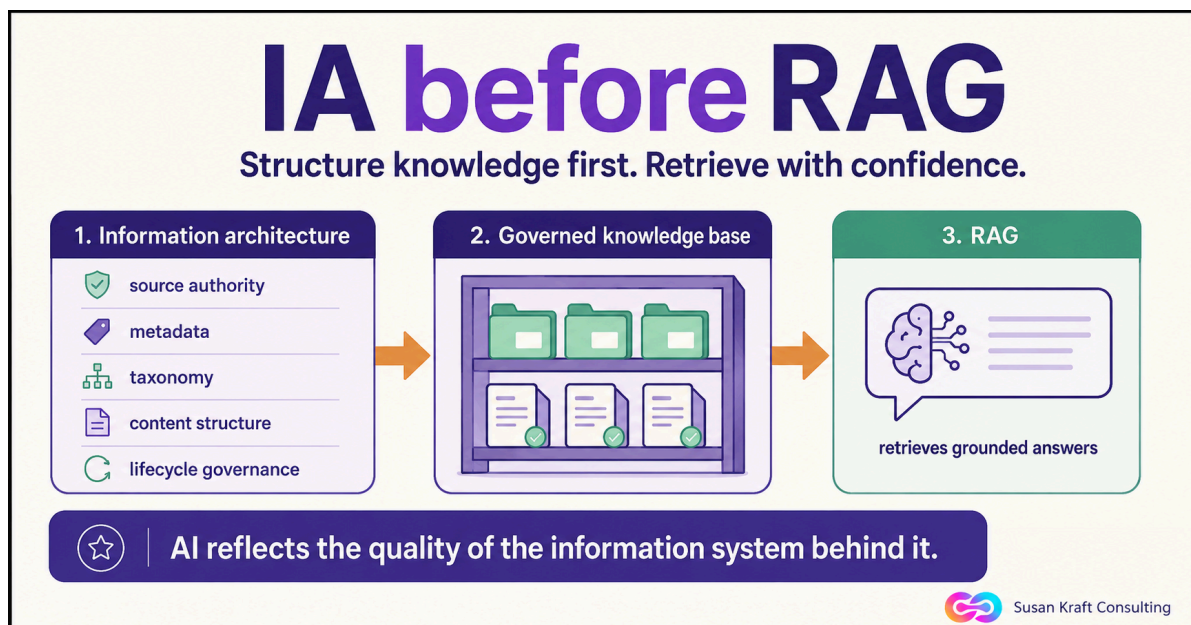
Abstract

Enterprise retrieval-augmented generation, or RAG, is commonly framed as a technical architecture for connecting large language models to internal knowledge repositories. This framing emphasizes retrieval pipelines, embeddings, chunking, and response generation. It gives less attention to the information environment that determines whether retrieved material is authoritative, current, structured, and usable.

This paper argues that information architecture is a prerequisite control layer for enterprise RAG. RAG can retrieve from approved repositories, but it cannot independently resolve duplicate content, stale documentation, unclear ownership, inconsistent metadata, weak taxonomy, or conflicting source authority. These conditions shape retrieval quality and directly affect the trustworthiness of AI-generated answers.

An IA-led approach reframes RAG implementation as a knowledge governance problem as well as an AI engineering problem. Before enterprise content is connected to AI retrieval systems, organizations need to define authoritative sources, normalize metadata, structure documents for retrieval, assign ownership, govern review cycles, and remove obsolete content. These practices improve both human findability and machine retrieval.

The central contribution of this discussion is a strategic operating model: information architecture establishes the governed knowledge system, while RAG provides an AI-enabled access layer to that system. This model shifts enterprise AI readiness away from tool deployment alone and toward the design, maintenance, and governance of trusted information assets.



Summary

Retrieval augmented generation, or RAG, is often described as the main enterprise AI solution: connect an AI system to company knowledge, retrieve relevant sources, and generate grounded answers. That framing is technically correct and strategically incomplete.

The stronger business argument starts with information architecture.

Information architecture, or IA, is the system that determines whether enterprise knowledge is findable, current, governed, structured, and trustworthy. RAG depends on that system being in place and trustworthy.

It is my contention that RAG quality depends on metadata, taxonomy, document structure, lifecycle governance, version control, and content ownership. It's given that poor IA produces poor AI answers.

The real AI-related fear we're talking about is how AI amplifies non-existent or poorly designed information systems.

That is the central argument. RAG is valuable because it helps AI retrieve and use enterprise content. IA is more fundamental because it determines whether the retrieved content deserves trust.

Plan to take time to plan and set up for clean success so you do not have to pay countless hours for imperfect cleanup after your systems built with poorly planned designs.

An IA led strategy turns RAG from an AI feature into a governed knowledge access layer.

Problem

Enterprise AI programs often begin at the retrieval layer.

Many organizations start with a technical goal: connect a language model to internal information sources. Common enterprise sources include Confluence, SharePoint, Google Drive, GitHub, Jira, PDFs, SOPs, API documentation, and support tickets. These systems are typical RAG inputs. Consider common business weaknesses: poor discoverability, duplicate content, stale documentation, unclear ownership, and inconsistent structure.

Those weaknesses shape the quality of every AI answer.

RAG retrieves from the information environment it is given. A source environment with conflicting implementation guides, outdated release notes, abandoned SOPs, duplicate API



“Information architecture as the control layer for enterprise RAG”, by Susan Kraft-Yorke

references, and unlabeled drafts gives the retrieval system weak evidence. The AI system may generate a fluent answer from material that carries limited authority or expired operational value.

The risk sits in the source chain.

The original RAG research by Lewis et al. describes retrieval augmented generation as a way to combine a pretrained language model with an external knowledge index. That architecture improves knowledge-intensive tasks because the model retrieves outside source material before generating an answer. This makes the external information system central to answer quality.

Enterprise AI succeeds when the knowledge system underneath the model supports accurate retrieval, traceability, ownership, and reuse.

Solution

The solution is to make IA the control layer and RAG the access layer.

Information architecture defines the structure of shared information environments. It organizes content, labels meaning, supports search, strengthens navigation, and creates usable retrieval paths across complex systems. Those functions map directly to enterprise RAG requirements.

An IA-led RAG program should use the following operating model.

Establish source authority

RAG needs a trusted source map before ingestion begins.

The organization must define which repositories own each knowledge domain. API references, release notes, runbooks, customer implementation guides, support playbooks, architecture decisions, and compliance procedures need assigned homes.

The AI system should know which source has authority for each type of answer. A published API reference, a draft migration note, and a two-year-old workaround may contain similar terms. Each source carries a different business role.

IA supplies that distinction.

Source authority should define:

- Approved repositories
- Document owners
- Intended audience



- Current version
- Superseded content
- Excluded content
- Review status
- Retirement rules

This step gives RAG a hierarchy of trust.

Design metadata and taxonomy before retrieval

RAG systems need metadata to improve search, filtering, and answer grounding.

Metadata fields such as title, summary, keywords, entities, product, version, owner, document type, audience, status, and expiration date give retrieval systems stronger ranking signals.

That is IA work before it becomes AI infrastructure.

Useful enterprise taxonomy should distinguish setup guides from API references, internal runbooks, release notes, support articles, known issues, and deprecated procedures.

This matters because retrieval is a ranking problem. Ranking improves when the system receives business meaningful signals.

Structure content for chunking and context

RAG systems frequently break documents into chunks. Document structure controls the quality of those chunks.

A long Confluence page with mixed audiences, vague headings, stale screenshots, and unrelated exceptions creates retrieval noise. A well structured document with specific headings, scoped sections, consistent terminology, and clear version context creates stronger chunks.

IA improves RAG by making each retrievable unit more self-contained and more meaningful.

Strong document structure should include:

- Specific headings
- One topic per section
- Clear audience cues
- Product and version context
- Stable terminology
- Explicit prerequisites

- Short procedural units
- Clear ownership and review data

A good IA structure helps humans and machines read the same material.

Govern lifecycle and ownership

RAG retrieves current information when the enterprise governs current information.

A governed knowledge system needs owners, review cycles, publication status, version rules, archive rules, and removal rules. Lifecycle governance gives the retrieval system a cleaner source base and gives humans a way to maintain trust over time.

For enterprise RAG, risk management includes the information layer. The system needs to know which sources are valid, which sources have expired, and which sources require human review before use.

Lifecycle governance should answer:

- Who owns this source?
- When was it last reviewed?
- Which product version does it support?
- Which document replaces it?
- Is it approved for AI retrieval?
- Is it internal, customer-facing, regulated, or restricted?
- When should it be archived?

RAG should retrieve from governed knowledge.

Use RAG as the governed access layer

After IA controls are in place, RAG becomes useful.

RAG can help employees ask natural language questions and receive answers grounded in approved company sources. It can reduce search time, expose relevant source material, and provide citations for verification.

The basic RAG workflow: the user asks a question, the system retrieves relevant enterprise content, the AI generates an answer, and the response may include citations or source references.

That workflow becomes operationally useful when IA supplies structure, authority, and lifecycle controls.



RAG should be measured as part of the knowledge system. Useful measures include retrieval precision, source freshness, citation usefulness, answer traceability, duplicate-source reduction, search time reduction, support escalation reduction, content owner response time, and failed-answer root causes.

These measures keep attention on the whole system. The goal is faster access to trusted operational knowledge.

Strategic discussion

The executive message is direct:

AI accelerates access to the knowledge system already in place.

A weak knowledge system produces weak AI retrieval. A governed knowledge system produces answers that can be traced, checked, corrected, and improved.

This reframes the value of IA in the AI era.

IA is operational infrastructure for AI readiness. It defines how knowledge is classified, owned, maintained, retrieved, and trusted.

RAG is one delivery mechanism inside that infrastructure.

This matters for engineering organizations, fintech teams, cloud platforms, regulated enterprises, and high velocity product groups. These teams already produce large volumes of technical content: APIs, implementation notes, runbooks, migration guides, release notes, support playbooks, architecture records, and compliance materials. AI increases the speed at which people can query that material. IA determines whether the answers are safe to use.

The practical sequence is:

1. Audit the information environment.
2. Identify authoritative sources.
3. Remove obsolete and duplicate content.
4. Define metadata and taxonomy.
5. Restructure documents for human and machine retrieval.
6. Assign ownership and review cycles.
7. Connect RAG to governed repositories.
8. Test retrieval against real user questions.
9. Review failed answers to find IA defects.
10. Improve the knowledge system continuously.

This sequence places IA before, during, and after RAG implementation.



Conclusion

RAG is an important AI architecture. IA is the discipline that makes RAG reliable in enterprise use.

The core insight: RAG quality depends on metadata, taxonomy, document structure, governance, version control, and ownership. Those are IA responsibilities.

The revised narrative is:

Information architecture leads. RAG aids.

IA creates the governed knowledge environment. RAG retrieves from that environment and helps AI generate source-grounded answers.

For business leaders, this changes the investment question. The enterprise should ask: “Is our knowledge system structured well enough for AI retrieval?”

That is where IA becomes strategic infrastructure.

References

Lewis, Patrick, Ethan Perez, Aleksandra Piktus, et al. “Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks.” 2020.

NIST. “AI Risk Management Framework.” National Institute of Standards and Technology.

NIST. “Artificial Intelligence Risk Management Framework.” NIST AI 100-1.

Microsoft Learn. “Develop a RAG Solution: Chunking Phase.”

Microsoft Learn. “Develop a RAG Solution: Chunk Enrichment Phase.”

Microsoft Learn. “Develop a RAG Solution: Preparation Phase.”

Rosenfeld, Louis, Peter Morville, and Jorge Arango. *Information Architecture: For the Web and Beyond*. Fourth edition.

Susan Kraft Consulting. “Why information architecture saves money and time.”

