Vespa.ai

Building Scalable RAG for Market Intelligence & Data Providers

How Vespa Delivers Accurate, High-Performance Retrieval for GenAI Agents at Web Scale

We Make AI Work

Vespa.ai

Vespa is the AI Search Platform for real-time, AI-driven applications—search, recommendation, personalization, and Retrieval-Augmented Generation (RAG). It powers large-scale AI by unifying data, inference, and logic, serving massive datasets and 100K+ queries per second. Trusted by leaders like Perplexity, Spotify, and Yahoo.

Document
Building Scalable RAG for Market Intelligence & Data Providers

08.01.2025

Contents

About this eBook

Market Intelligence & Data Providers face a unique challenge: delivering accurate GenAI-powered insights over massive, domain-specific datasets while keeping response times and infrastructure costs low. Deep research and multi-step AI agents create more value for subscribers, but they place an enormous strain on computational resources.

For years, data service providers have used AI to collect, clean, and enrich vast, noisy datasets. Now, GenAI—delivered as domain-focused AI agents—has the potential to transform how subscribers search, summarize, and act on this intelligence. Unlike consumer-grade tools trained on public web data, these services must blend curated business, financial, and product intelligence with subscriber-specific context to deliver precise, high-value answers. That deeper integration unlocks richer insights but also pushes infrastructure to its limits.

Traditional infrastructures weren't built for this. Vector databases alone solve only part of the problem—they handle nearest-neighbor search but cannot meet the full demands of Retrieval-Augmented Generation (RAG). Large-scale RAG requires much more: combining semantic, keyword, and metadata retrieval; applying machine-learned ranking; and continuously managing dynamic, structured, and unstructured data. Doing this reliably across billions of documents, at sub-100 ms latency, for thousands of concurrent queries typically requires stitching together multiple systems—adding complexity, risk, and cost.

This eBook introduces a scalable RAG approach powered by the Vespa AI Search Platform. Proven at web scale by Perplexity, Vespa unifies accurate retrieval, real-time ranking, and cost-efficient performance into one system. With best practices like The RAG Blueprint, Market Intelligence & Data Providers can deliver GenAI to their customers faster—without sacrificing scale, reliability, or control.

Introduction

The GenAI Opportunity

Generative AI opens new possibilities for Market Intelligence & Data Providers to deliver richer, more intuitive subscriber experiences. Users now expect to ask complex, multi-step questions in natural language and get precise answers instantly—just like they do with Perplexity or ChatGPT. Meeting that expectation is hard. Providers must deliver reliable intelligence at scale across massive, fragmented datasets, where accuracy directly impacts reputation.
RAG—Retrieval-Augmented Generation—does the heavy lifting. It searches, filters, and ranks relevant data, then passes it to a Large Language Model (LLM) to explain in clear, conversational language. At scale, though, this is a massive technical challenge. For example, platforms like Perplexity must search billions of documents and serve hundreds of thousands of users simultaneously, while still maintaining fast, accurate responses.

Why Now?

User expectations have moved beyond static reports and limited chatbot interactions. They now expect systems that understand complex questions and return clear, contextual answers—instantly. At the same time, the volume and complexity of data have made traditional, human-speed search unsustainable. Recent advances in large language models, vector retrieval, and real-time inference have created a tipping point: providers can now deliver intelligent, machine-speed search at scale. The **Vespa AI Search Platform** brings these capabilities together—combining advanced retrieval techniques with a high-performance runtime engine that keeps operating costs low. This enables service providers to support deep research workflows and deploy AI agents that truly deliver.

Why Vector Databases and Lucene Alone Fall Short

Vector databases are a key enabling technology for GenAI, but they offer only a partial solution. Their strength lies in semantic similarity—finding items that are “close” in embedding space. This is useful for matching meaning, but on its own, it falls short of what’s needed for large-scale RAG, where precision, relevance, and performance at scale are critical:

- **Lack of exact matching**: Domain-specific accuracy often depends on precise terminology that vectors alone can miss.
- **Embedding limitations**: Retrieval quality depends on what the embedding model understands; niche vocabulary can be lost.
- **No signal blending**: Vector DBs rely on distance metrics but don’t combine other relevance signals like recency, authority, structured metadata, or business rules—critical for precise results.

Many “vector databases” are Lucene-based (e.g., Elasticsearch, OpenSearch), bolting vector capabilities onto an inverted index. This introduces new trade-offs:

- **Hybrid queries are inefficient:** Lucene’s inverted index was built for keyword search, not high-dimensional vector similarity.
- **Cross-network bottlenecks:** Vector, keyword, and ranking layers are often separate, forcing data to move between nodes and increasing latency.
- **Limited ML ranking integration:** Lucene isn’t designed for real-time ranking models, forcing external services that slow responses.

Lucene-based systems also struggle with fine-grained retrieval. Traditional pipelines only rank whole documents. For RAG, you often need the most relevant chunks within those documents to keep LLM context tight. Pure vector databases force you to either store small chunks as “documents” (losing broader context) or retrieve full documents and let the LLM handle noisy, irrelevant text—wasting tokens and hurting accuracy.

Finally, scaling Lucene-based systems is operationally fragile. Adding or removing nodes causes heavy rebalancing and downtime, real-time updates can be slow and costly, and ANN indexes often require full rebuilds, breaking continuous RAG workflows.

The Vespa AI Search Platform

Vespa GenAI (RAG): Key Capabilities

Vespa is the first AI Search Platform designed for generative AI applications, supporting the demands of deep research and AI agents. It enables users to extract insights from hundreds of millions of documents in seconds. Vespa supports decision-making by securely powering AI-driven search, summarization, and analysis on sensitive data—without sacrificing performance or accuracy. Proven at web scale, Vespa can handle vast amounts of structured data, billions of documents, thousands of concurrent queries, and sub-100 ms latency, all while keeping infrastructure costs in check.

Vespa addresses the fundamental shortcomings of vector-only and Lucene-based solutions. It integrates hybrid retrieval (vectors + keywords + metadata), real-time ML ranking, and distributed chunk-level selection into one platform:

- Retrieval, ranking, and filtering happen locally where the data resides, avoiding Lucene's network bottlenecks.
- It scales elastically with billions of documents while maintaining sub-100 ms latency.
- It supports layered ranking (document + chunk) to feed only the most relevant context to the LLM.
- Nodes can be added, removed, or upgraded without downtime, handling continuous updates natively.

Vespa is a full AI Search Platform delivering accurate retrieval, high-performance ranking, and massive scale for RAG without stitching together multiple components. Two key capabilities matter most for data service providers:

- **Relevance**: delivering the most accurate results.
- **Scalability**: handling massive data sizes, query volumes, and performance needs without breaking cost models.

Accurate Retrieval with Hybrid Search

To deliver precise, highly relevant results, Vespa maintains a multi-layered model of your data, indexing title and body text, dense embeddings for both, and additional fields like site information or link text. It also leverages hundreds of metadata signals, from behavioral metrics and PageRank-style scores to ML classifier outputs, to refine relevance.

Vespa combines dense vector search, keyword matching, and structured-data filtering in a single query, balancing semantic understanding with exact term matching. All data modalities—vectors, text, and structured fields—are co-located on the same nodes, enabling fast hybrid retrieval without network hops.

For example, a vector-only search for "*emerging fintech companies with strong growth in APAC*" might return semantically similar results like "*fast-growing payment startups*" or "*digital banking providers in Asia*." Vespa's hybrid approach refines this further by incorporating keyword constraints (e.g., "APAC"), metadata filters (e.g., funding round, valuation, region), and domain-specific signals such as industry classifications—yielding more precise, actionable results.

Vespa also supports visual retrieval models such as ColPali, which embed entire rendered documents (including visual context) for querying visually rich documents like PDFs. By treating documents as visual entities rather than plain text, Vespa preserves layout and context while simplifying preprocessing—ideal for billion-scale PDF and complex document applications.

Ranking to Deliver the Best Results

After retrieving candidates, Vespa applies GBDT machine-learned ranking models directly on the content nodes, where the data resides. Ranking combines multiple signals—vector similarity, keyword matches, recency, metadata, and custom quality scores—into a single relevance score. Vespa first runs a lightweight function on all matches, then applies heavier ML inference only on the top candidates, optimizing both cost and latency.

Because ranking is executed locally on distributed nodes, Vespa avoids network bottlenecks and scales inference naturally with cluster size. This ensures only the most relevant documents—or specific chunks—are selected for the LLM context.

Traditionally, RAG systems rank only at the document level, retrieving the top N documents wholesale. Vespa introduces layered ranking, which ranks and selects the top N documents and also the top M chunks within each. This second-layer ranking runs in parallel on the nodes storing the data, maintaining constant latency regardless of query rate or document size.

This lets you extract only the most relevant portions of content—avoiding the trade-off between overly fine-grained chunks that lose context and bloated prompts full of irrelevant text.

Massive Scale Without Massive Costs

Market Intelligence & Data Providers need to search millions—often billions—of documents updated daily, serving thousands of users and delivering 99% of queries in under 100 ms with near-perfect availability. But retrieval demands are evolving. AI agents now perform multi-step reasoning, issuing many sequential queries to explore, verify, and synthesize information. Systems optimized for “human-speed” search simply can’t keep up.

Vespa is built for this reality. It scales elastically in both data volume and query load by partitioning and replicating data across nodes. It can serve billions of documents, thousands of concurrent queries, and sub-100 ms latency without saturating network bandwidth. Unlike systems that shuffle data between disconnected components, Vespa co-locates indexes, vectors, and structured data on the same nodes, eliminating single-node bottlenecks.

Scaling up or down is seamless. Nodes can be added or removed while the system stays online, handling queries and writes without disruption. Workloads can be tuned by balancing higher-capacity nodes against larger clusters, and Vespa supports migrating between machine types with no downtime.

Crucially, Vespa handles continuous updates—hourly application and model changes, platform upgrades, and even node failures or migrations—without service interruptions.

Integrating Client Data

Subscribers gain even more value when GenAI can analyze both the provider’s proprietary intelligence and the subscriber’s own private data. Vespa supports this seamlessly. At Perplexity, for example, Vespa powers retrieval over both its indexed knowledge base and any files uploaded by Pro users, enabling unified answers across private and public data.

Indexing

Accurate retrieval starts with strong indexing. Vespa provides a flexible, scalable indexing framework for both structured and unstructured data. Fields can be optimized for full-text search, vector embeddings, or real-time filtering. Key features include:

- **Hybrid indexing**: Combine keyword-based (BM25) and vector-based retrieval for superior accuracy.
- **Real-time & batch ingestion**: Support low-latency updates and large-scale indexing.
- **Tensor-based vector indexing**: Store and search embeddings for AI-driven retrieval.
- **Distributed scalability**: Automatically distribute data across nodes for high performance and fault tolerance.

The RAG Blueprint

Even with Vespa's automation, building scalable RAG systems can be complex. The RAG Blueprint is a modular application template for designing, deploying, and testing production-grade RAG systems. Built on the same core architecture that powers Perplexity, it codifies best practices for accurate and scalable retrieval pipelines using Vespa's native support for hybrid search, layered ranking, and real-time inference. Designed for developers and architects, the Blueprint accelerates production-ready implementations—helping teams move faster without sacrificing quality or control.

Summary

Delivering GenAI-powered insights isn't just about connecting an LLM to a dataset—it's about retrieving the right information with the speed and accuracy that AI agents demand. Vector databases and Lucene-based systems provide a partial solution, focusing on semantic similarity but missing the precision, hybrid retrieval, and layered ranking needed for enterprise-grade RAG. They also struggle to scale to billions of documents, multi-step reasoning, and thousands of concurrent queries without introducing latency, complexity, and operational fragility.

Vespa solves these challenges by integrating all critical retrieval functions into a single AI Search Platform. It combines dense vector search with keyword and metadata filtering, applies machine-learned ranking directly on distributed content nodes, and introduces layered ranking to select not just the right documents but the right chunks within them. This approach ensures only the most relevant context is passed to LLMs—improving accuracy while keeping token usage and costs in check.

Designed for massive scale, Vespa handles billions of documents, sub-100 ms query latency, and real-time updates—all without downtime. It eliminates the need for brittle multi-system pipelines, reducing operational risk and infrastructure sprawl. Proven at web scale by Perplexity, Vespa lets Market Intelligence & Data Providers deliver AI agents that expose proprietary data securely and reliably, creating richer subscriber experiences while keeping performance and costs under control.

With best practices like The RAG Blueprint, Vespa helps providers move from proof-of-concept to production-ready RAG systems quickly and confidently.

About Vespa.ai

Vespa.ai is an AI Search Platform for building and running real-time AI-driven applications for search, recommendation, personalization, and RAG. It enables enterprise-wide AI deployment by efficiently managing data, inference, and logic, handling large data volumes and over 100K queries per second. Vespa supports precise hybrid search across vectors, text, and structured metadata. Available as both a managed service and open source, it's trusted by organizations like Perplexity, Spotify, Wix, and Yahoo. The platform offers robust APIs, SDKs for integration, comprehensive monitoring metrics, and customizable features for optimized performance.

Interested to learn more? We have many different resources and information available through our social platforms

GitHub Twitter LinkedIn YouTube

Vespa.ai