

从蓝图到现实：一位架构师的RAG系统构建之旅

探索、挑战与架构决策

RAG：超越模型，重塑架构的认知核心

基础LLM的固有缺陷



知识时效性与局限性
事实幻觉与可信度危机
上下文窗口与成本瓶颈

RAG架构的核心价值



事实证据提供者：动态连接实时、私有知识库。
上下文管理器：提供可信、可审计的引用来源。

成功的基石：量化RAG系统的效果与质量

遵循TRIAD框架，将系统分解为**检索器 (Retrieval)** 和**生成器 (Generation)** 两个核心组件进行独立评估，实现指标驱动开发 (MDD)。



上下文精度 (Context Precision)

检索结果的质量，确保排名靠前的文档是相关的。



上下文召回率 (Context Recall)

检索结果的完整性，确保所有关键信息都被召回。



忠实度 (Faithfulness)

衡量生成答案是否完全由检索证据支持，是减少“幻觉”的关键指标。



答案相关性 (Answer Relevancy)

确保答案聚焦于用户意图，而非离题。

从POC到生产：RAG系统面临的效率瓶颈



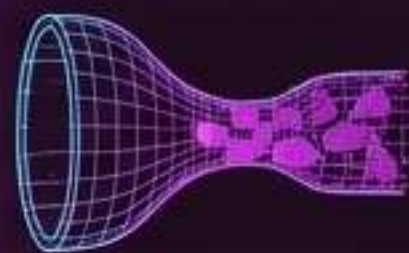
高延迟：平均响应时间过长，尤其在GraphRAG等复杂模型中。



高成本：令牌使用量 (Token Usage) 与长上下文窗口导致API调用成本激增。



算力瓶颈：GraphRAG构建知识图谱时频繁调用大模型，导致算力消耗激增。



吞吐量限制：系统在特定时间内可处理的请求数量受限。

架构师的抉择：为RAG选择核心知识引擎

统一平台 (Unified Platform)

代表：Elasticsearch

理念：整合全文搜索、向量搜索和结构化分析，简化技术栈。

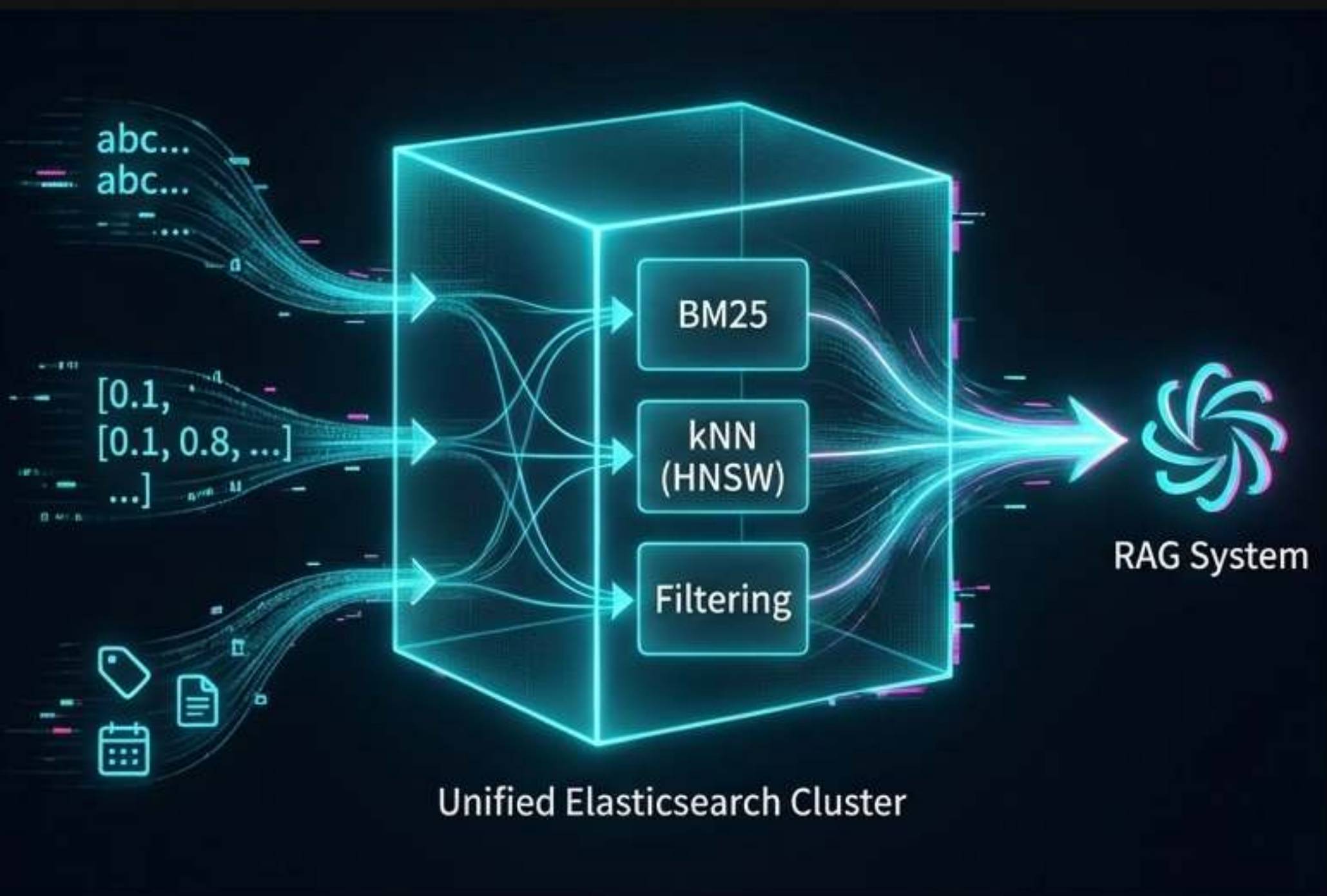
专用引擎 (Specialized Engine)

代表：Milvus

理念：专注于海量向量数据的高性能相似性搜索，为极致规模和性能而生。



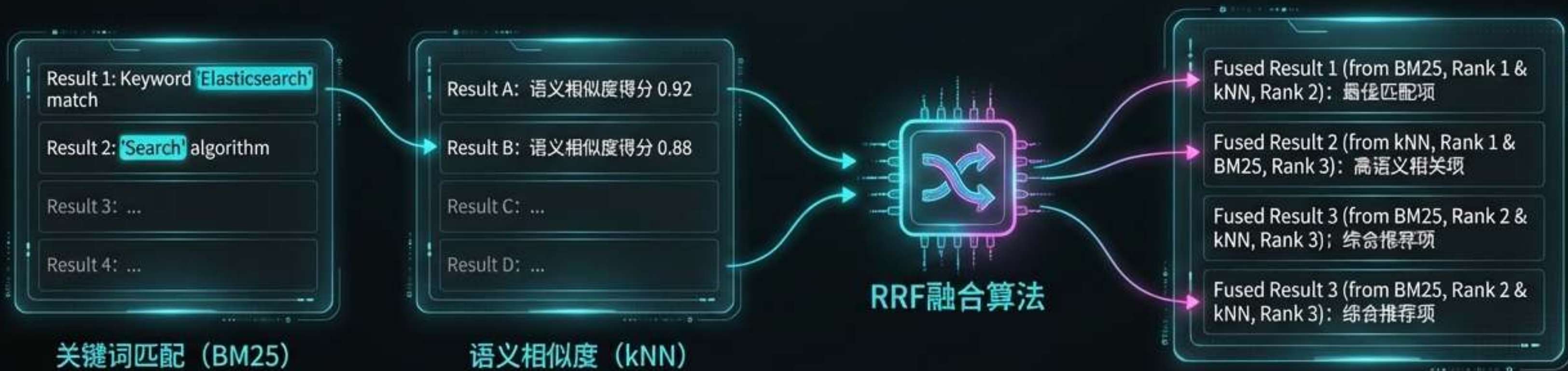
统一的力量：Elasticsearch作为一体化知识检索库



- **一体化平台**：在一个集群中存储原始文本、向量和元数据，避免了引入独立向量数据库的运维复杂性。
- **强大的结构化过滤**：核心架构优势，可利用 Query DSL 在向量搜索期间无缝应用元数据预过滤（Filtered kNN）。
- **成熟的全文检索**：久经考验的BM25算法，擅长精确匹配专有名词和关键词。

Key Technology：使用 `dense\_vector` 字段存储向量，并主要采用 **HNSW 算法** 实现高效的近似kNN搜索。

检索精度的飞跃：Elasticsearch的混合搜索与RRF融合



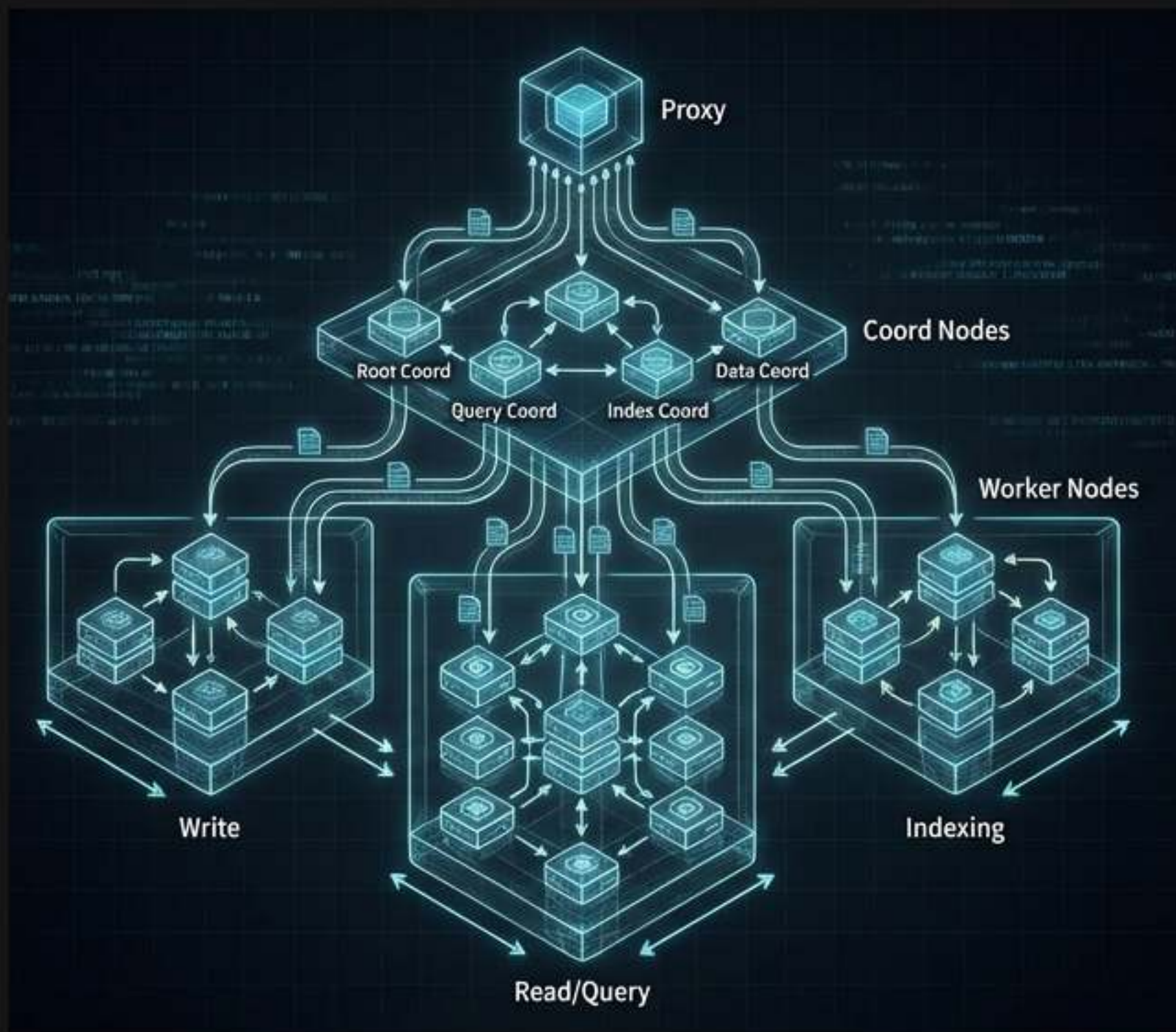
混合搜索并行执行全文搜索 (BM25) 和向量搜索 (kNN)，克服了单一检索模式的局限性。

The Fusion Algorithm

- **倒数排名融合 (Reciprocal Rank Fusion - RRF)**：基于结果的 **排名位置** 而非原始分数进行智能融合。
- 核心优势：**无需调整权重** 即可智能合并结果，显著提高相关性。

通过 **rrf** 检索器实现，将 **standard** (BM25) 和 **knn** (向量) 的输出合并。

极致性能：Milvus为海量向量而生的专用架构



Core Strengths:

- **分布式微服务架构:** 读、写、建索引相互解耦，实现存储和计算分离，具备极高扩展性。
- **丰富的索引算法:** 支持 **HNSW, IVF, ANNOY, FAISS** 等多种索引，允许根据场景进行深度优化。
- **深度硬件加速:** 强调GPU加速，在计算密集的检索和索引任务中性能大幅提升。

Key Design Philosophy: 专为处理 **千亿级向量** 和 **高并发、低延迟场景** 而优化。

架构师的决策清单：Elasticsearch vs. Milvus

维度	Elasticsearch (统一平台)	Milvus (专用引擎)
核心优势	混合搜索 (BM25+kNN+RRF) 强大的元数据过滤	极致性能与扩展性 专为千亿级向量设计
架构设计	一体化平台，简化运维	解耦的分布式微服务
索引算法	HNSW (基于Lucene)	HNSW, IVF, ANNOY, FAISS等
硬件加速	支持GPU索引构建	深度GPU集成，性能提升显著
适用场景	企业级RAG，需高精度混合检索 已在使用Elastic Stack的团队	超大规模数据，对QPS和扩展性有极高要求 计算密集型任务

RAG的能力边界：当简单检索遇上复杂任务



RAG的终极演进：构建Agent智能的知识基石



RAG作为知识源：为Agent workflow (Workflow) 提供稳定、可信、类型化的知识来源，支撑Agent进行决策、规划与反思。

突破能力边界：与Agent架构或 workflow 编排技术结合，突破RAG在过程控制、工具调用和状态管理上的限制。

未来的架构：RAG是构建可靠、可扩展AI Agent的知识基石。掌握RAG架构是通往下一代智能应用的前提。