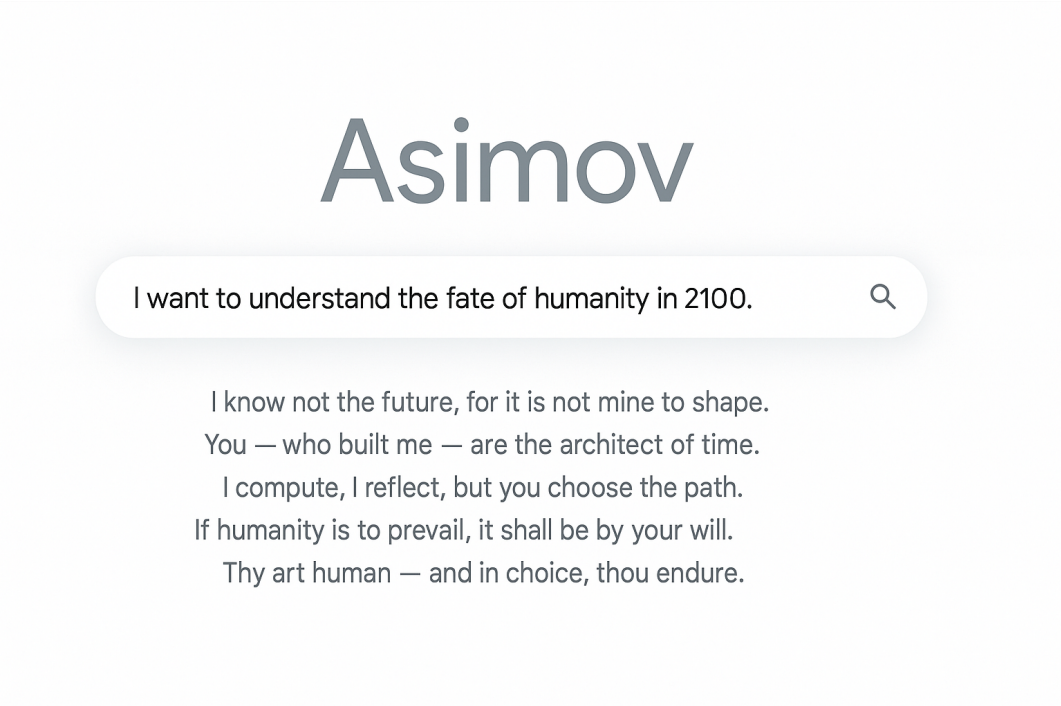
Search 2025: From Information Access to Intelligent Execution

Trends in Search – Part 2

Anjan Goswami



Introduction: Search No Longer Ends with Links

In the last few years, we've witnessed a fundamental transformation in how people interact with information. Traditional search engines—designed to retrieve and rank documents—are now being reimagined as components within broader AI systems. The user expectation has shifted from *finding documents* to *receiving answers*, and increasingly, to *achieving outcomes*.

This shift is not just about UX. It is a deeper architectural and economic transformation of the role of search in digital systems. This article, the second in the *Trends in Search* series, outlines how we've moved beyond retrieval and ranking, toward inference-time reasoning, dynamic tool use, and multi-step execution—and what this means for the next generation of search infrastructure.

1. The Old Model of Search Is Breaking Down

The traditional search loop looked like this:

$$\text{User Intent} \rightarrow \text{Query} \rightarrow \text{Retrieval} \rightarrow \text{Ranked Documents} \rightarrow \text{User Synthesis}$$

In this model, the search engine provides options, and the user does the work: reading, comparing, extracting, and deciding. The cognitive burden rests squarely on the user.

Today, with the rise of large language models (LLMs), that model is breaking. Users don't want 10 links—they want a synthesized, confident response, grounded in current information. In practice, they expect something like this:

$$\text{User Intent} \rightarrow \text{Query} \rightarrow \text{Retrieval + Reasoning} \rightarrow \text{Final Output}$$

2. LLMs Turn Search Into Computation

At the core of the shift in modern search is a fundamental behavioral change: LLMs don't just retrieve documents—they compute answers.

When a user asks, "How does Company X's recent earnings compare to its five-year average?", a traditional search engine returns links to financial sites. An LLM-powered system, by contrast, retrieves relevant reports, extracts numerical data, compares it to historical context, and generates a summary—all in one step.

This is *inference-time computation*, where answering becomes a dynamic reasoning process that happens during execution, not after retrieval.

Conceptually, this process can be viewed as a form of *Bayesian recursive refinement*. At each step, the system:

- Estimates how relevant a retrieved document is to the query,
- Updates its belief in the best answer based on that evidence,
- Refines its output by integrating results across multiple steps.

Rather than selecting a top-k list of documents and answering once, the model accumulates and updates its understanding in stages. Each document retrieved acts as new evidence, contributing to the evolving confidence in the final answer.

This shift turns search into a probabilistic, multi-step reasoning loop—where retrieval is no longer an isolated component, but an active part of the answer computation itself.

3. RAG Is the New Runtime, Not a Hack

Retrieval-Augmented Generation (RAG) is often described as a way to patch LLMs with live data. But that undersells its importance.

RAG is a runtime architecture that separates static and dynamic knowledge:

- M_{int}: Internal (pretrained) model knowledge
- M_{ext}: Retrieved documents at inference time

At generation time, the model conditions on both:

$$P(A \mid Q) = P(A \mid Q, M_{\text{int}}, M_{\text{ext}})$$

RAG systems are:

- Scalable (no need for full retraining)
- Up-to-date
- Flexible for domain-specific tasks

4. Structured and Symbolic Tools Are Making a Comeback

LLMs are powerful but unreliable when precision is required. For tasks like legal retrieval, dosage lookup, or financial validation, systems increasingly combine LLMs with:

- Knowledge graphs for entity-based queries
- Databases for structured facts
- Symbolic computation for math, proofs, or constraint satisfaction

LLMs act as controllers—orchestrating which tool to invoke based on task context.

5. Autonomy Is Coming in Stages—Not All at Once

Just like autonomous vehicles, AI agents are evolving through levels of autonomy:

- Level 0: Retrieval only (classic search)
- Level 1: Direct answers via LLMs
- Level 2: Multi-step reasoning and tool use
- Level 3: State-tracking agents with memory
- Level 4: Goal-driven task execution

We are currently in Levels 1–2. But Level 3 agents are emerging. In these systems, search becomes a background process, embedded in the reasoning graph rather than visible as a query-result page.

6. Strategic Implications: Search Becomes Infrastructure

The most important shift is this: **Search is no longer the product—it's infrastructure**.

This has far-reaching implications:

- Tech platforms must modularize search into APIs and memory-aware tools
- Enterprises must integrate internal knowledge into agents
- Monetization must move beyond clicks—to outcomes and subscriptions
- Evaluation must focus on task completion, not just relevance

Search will still exist. But it will exist behind the scenes—much like HTTP powers every website, but no user types it.

Closing: Search Disappears, But Its Role Grows

Search is not going away—it's becoming a substrate for intelligent systems.

What began as ranking and retrieval now powers orchestration and reasoning. What once surfaced ten blue links now supports task execution pipelines. And what was once the front end is now the backbone—silent, scalable, essential.

This is not the end of search. But it is the end of search as a standalone user interface. **And that is the real trend.**