Chaps Vision

The Ultimate Guide to Enterprise Agentic AI

A Practical Guide for Enterprise Leaders, Architects, and Practitioners

Table of Contents

INTRODUCTION

From AI Hype to Enterprise Impact

Everyone is talking about agentic artificial intelligence (AI). Vendors promise AI coworkers, self-running businesses, and autonomous decisioning at scale. But if you are responsible for delivering these outcomes in a large enterprise, the real questions are a lot more pragmatic:

- What is agentic AI in concrete, operational terms?
- How do you get from today's chatbots and copilots to real AI agents and multi-agent systems?
- How do you ground those agents in your own data, keep them secure, and make them observable and governable?
- What kind of platform do you need, and what does doing it right look like in practice?

This guide is for enterprise leaders, architects, and practitioners who need answers to these questions and more. It lays out the journey from simple chatbots to full agentic AI, explains the knowledge and trust challenges that derail most initiatives, and shows why Retrieval-Augmented Generation (RAG) and enterprise agentic AI search are now widely regarded as foundational to any credible agentic strategy.

CHAPTER 1

What Is Enterprise Agentic AI

Enterprise agentic AI is the use of AI agents and multi-agent systems that are grounded in your own enterprise knowledge and governed like any other critical system. Their job is to understand goals, plan steps, and take actions inside the same engineering, operations, customer, and back-office workflows your people run today.

From Chatbots to Agents

Traditional chatbots and copilots use Large Language Models (LLMs) to answer questions and assist with simple tasks. They are powerful but fundamentally reactive. They respond to prompts and rarely take initiative.

In contrast, agentic AI introduces software entities, known as AI agents, that can:

- Understand high-level goals expressed in natural language
- Plan multi-step workflows to achieve those goals
- Use tools such as APIs, enterprise applications, and services to gather information and take action
- Adapt their plan as they go, based on feedback and intermediate results

An AI agent is not just a more sophisticated chatbot. It is a system that can decide what to do, how to do it, and when that task should be done. It can call other tools and systems, and in more advanced settings, collaborate with other agents to complete complex objectives that would overwhelm a single agent.

What Makes It Enterprise Agentic AI

In enterprise environments, agentic AI must operate under far more demanding conditions than the tools typically available to consumers. It must:

- **Span thousands of systems and formats**, including legacy, highly specialized repositories such as Product Lifecycle Management (PLM) stores and research and development (R&D) data.
- **Respect complex security and regulatory constraints** while using internal knowledge that never leaves controlled enterprise environments.
- **Be observable and governable** across many business units at once, so teams can trace agent behavior, intervene when necessary, and prove value at enterprise scale.

Enterprise agentic AI is therefore not simply about giving agents more autonomy to act without human intervention. It is about building an ecosystem of specialized agents, deeply integrated with enterprise knowledge, that can operate safely and reliably in real business workflows. In large enterprises this means supporting many teams and business units on the same platform, without duplicating integrations or compromising security boundaries.

Delivering this in practice requires an enterprise agentic AI platform: a unified stack that connects to all your enterprise systems and data, powers advanced Retrieval Augmented Generation (RAG), and provides the security, observability, and governance needed to run agents at scale.

See Enterprise Agentic AI in Action

Book a demo →

CHAPTER 2

The Journey to Agentic AI

Agentic AI is best understood as a journey rather than a single technology implementation. Each step adds capabilities to the previous one. Not every use case needs to reach the final step; indeed, many high-value applications stop at the level of AI assistants or general experts.

The Progression of Systems

1) Chatbot (LLM)

A conversational interface built on an LLM. It answers general questions, writes emails, summarizes text, and helps with brainstorming. However, it knows almost nothing about your internal business and has no understanding of tools or workflows.

2) AI Document Expert (AIDE)

A chatbot paired with a single document. An employee uploads a test specification, contract, or policy, then asks targeted questions, such as what load testing is required for a specific part or what the termination clauses are. The system reads the document and responds with answers and references to the relevant sections.

3) AI Subject Expert (AISE)

A chatbot grounded in a predefined set of documents on a specific topic, plus a custom prompt. One example might be a travel policy expert or benefits assistant that always uses approved human resources (HR) documents and guidance. An AISE can answer questions, summarize policies, and help employees navigate complex rules, but its knowledge is limited to what was uploaded to the chatbot.

4) AI General Expert (AIGE)

Here, RAG enters the picture. Instead of being limited to a small document set, an AIGE can search across enterprise knowledge, such as file shares, Drive or SharePoint, PLM systems, wikis, and ticketing systems to dynamically retrieve the right information to answer a question. An AIGE can handle permissions, stay up to date, and provide citations.

5) AI Assistant

An AI Assistant takes RAG-powered understanding and embeds it in a predefined multi-step workflow. For example, a status report assistant might: ask for a time window, collect meeting summaries and tickets, extract key issues, milestones, and metrics, fill in a standard report template, and email the result to a defined distribution list. The workflow is static and predefined, but it can include branches, loops, and tool calls.

6) AI Agent

AIAgents go beyond static workflows. They are given a goal, such as perform deep internal research on the causes of warranty failures in a given region for a specific product. Then they construct, adapt, and execute the workflow in real time. They determine which tools to use, how often to call them, and when the task is complete.

They can backtrack and iterate when something does not work. Most offerings marketed as agents today stop short of this behavior and are closer to scripted assistants with fancier prompts than they are to true goal-driven systems.

7) Agentic AI (Multi-Agent Ecosystem)

Immature deployments, multiple specialized agents work together. One might handle customer authentication, another pricing, another compliance checks, and another reservations. They pass information and control between each other to achieve complex objectives. The system includes mechanisms for agents to discover one another, orchestrate their collaboration, and provide end-to-end visibility across all agent activities.

For many organizations, this journey does not start with agents at all, but with becoming truly information-driven by unifying access to research, operational data, and historical knowledge so employees can reliably find what they need.

> **TIP:** Don't start with a fleet of agents. Start with a strong AIGE and assistants built on enterprise search and RAG and let real results tell you what deserves to become an agent.

In many enterprises, teams spend months building a generic AI employee that can, in theory, answer anything, only to discover that it cannot reliably find the latest specification in PLM or the right contract clause in SharePoint because there is no solid enterprise search and agentic RAG layer underneath. Instead of starting with a fleet of autonomous agents, establish a solid AIGE and a small set of well-designed assistants on top of enterprise AI search and RAG, then promote only proven patterns into agents and, eventually, multi-agent systems.

Choosing the Right Destination

Itmaybe tempting to aim for full agentic AI everywhere, but that is not always necessary or efficient. Many organizations' needs are fully met by an AIGE or a well-designed AI Assistant. The goal is to match the complexity of the solution to the complexity of the problem.

Task Type	Best Fit	Capabilities
User just needs answers faster	AI General Expert	RAG-powered search over enterprise knowledge.
Repeatable, well-understood task	AI Assistant AI	Fixed workflow with clear steps and
Open-ended, variable task	Agent / agents	tools. Goal-driven planning and tool use.

All of this only works if the underlying knowledge layer is strong. When retrieval is weak or incomplete, even well-designed agents will fail in production.

To see this journey in action, watch the webinar AstraZeneca's Journey to Become Information-Driven with Cognitive Search.

From Search to Action: The Real Journey to Enterprise Agentic AI

Ready to map out your own path from enterprise search to full agentic AI? Our whitepaper breaks down every step of the journey.

Download the Whitepaper →

CHAPTER 3

The Knowledge Challenge in Enterprise AI

LLMs Alone Are Not Enough

Generative AI has an obvious limitation. LLMs are models of language, not of your business. They are trained on public data and recognize patterns in text, but they do not store or retrieve information the way a database or search engine does. This limitation creates two fundamental problems.

1. LLMs know nothing about the internal world of your enterprise. They have no knowledge of your internal projects, product variants, historical failures, customer contracts, or proprietary designs. At best, they know what has surfaced in public documentation or on your public website.

2. They will hallucinate with confidence. LLMs generate plausible text, not guaranteed truths about niche topics or enterprise-specific details. They often sound authoritative even when they are wrong, which is dangerous in business contexts where decisions affect revenue, safety, or compliance.

As a result, generic copilots and chatbots quickly hit a wall. Without access to internal knowledge, they cannot answer the questions that matter most to your employees.

The Nature of Enterprise Knowledge

Enterprise knowledge itself presents serious challenges, because it is:

- **Fragmented** across dozens or hundreds of systems, such as file shares, SharePoint, PLM, ERP, CRM, ELNs, ticketing, email archives, and other repositories.
- **Multimodal.** Information spans text, diagrams, CAD models, tables, spreadsheets, images, videos, and call recordings.
- **Permissioned** with complex security models. Permissions vary by department, role, geography, and project, and sometimes are inherited from legacy systems.
- **Dynamic**, with constant updates, new versions, and evolving structures as products change and organizations reorganize.

That is why data readiness consistently emerges as the number one blocker in moving from proof of concept to production.

The Need for a Shared Knowledge Store

Tosupportassistantsand agents,enterprisesneed a unified enterprise knowledge fabric that can ingest content from all relevant systems, normalize and enrich it, preserve and enforce source-system security, and support multimodal hybrid search so agents can find answers.

Enterprise AI search platforms and advanced agentic RAG pipelines are what turn scattered content into an accessible, governed knowledge store. Increasingly, this fabric is exposed to agents through Model Context Protocol (MCP) servers. MCP servers sit between the agent framework and your enterprise systems, brokering secure, governed access to tools and data. However, they also introduce their own security and governance risks that must be managed explicitly.

CHAPTER 4

Ways to Bring Enterprise Knowledge into AI

Before the industry converged on RAG, three approaches were widely used to bring enterprise knowledge into generative AI.

Custom Models

Custom models sound attractive, simply training an LLM on your own data, but in practice they are a poor fit for most enterprises. Training a model from scratch is very expensive in terms of data, compute, and specialized expertise, and the result still inherits core LLM issues, including hallucinations. Whatever you build into the weights is also frozen, so the model starts going stale as soon as products, policies, or documents change, and retraining the model to keep up is rarely realistic. In addition, any sensitive data used in training is effectively spread throughout the model, which makes it hard to guarantee that confidential information cannot leak in unexpected ways. For most organizations, the real bottleneck is not the model itself but secure, high-quality retrieval over their knowledge.

Fine-tuning Models

Fine-tuning can be useful for style, tone, or specialized reasoning, but it is insufficient on its own as the foundation for enterprise AI. Fine-tuning rarely solves the real enterprise problems of security, freshness, and retrieval; it mostly makes the model sound more like you want it to, while still not knowing very much about your organization.

Grounding via RAG

Grounding takes a different approach. Instead of trying to bake your knowledge into the model, it gives the model the relevant knowledge at the moment of the request. At enterprise scale, RAG operationalizes this pattern by:

- Using enterprise AI search and retrieval to find the most relevant passages across your systems, based on the user's intent
- Feeding those passages into the prompt as grounding context before the model responds
- Having the LLM base its reasoning and responses on that retrieved knowledge
- Providing citations so humans can verify and drill down into the underlying sources

Grounding via RAG is accurate and verifiable, with minimal hallucinations, full traceability to source content, and auditable knowledge paths. It is also economical, adaptable, and safe, because you avoid retraining, can swap and combine models, control usage centrally, and ensure employees only see permitted information as new data is indexed.

RAG has become the standard way to bring enterprise knowledge into generative AI.

More on Enterprise Knowledge: Webinar: Why Enterprise Search Is Foundational to Effective Knowledge Management ▪ Guide: Knowledge Management Guide

CHAPTER 5

RAG for Grounding Enterprise AI

Connection: Access to All the Right Knowledge

A RAG system is only as good as the content it can see. Connectors must:

- Reach all important systems, including specialized ones such as PLM, ELNs, and regulated repositories, without losing permission controls
- Support all relevant formats, from Office files and PDFs to CAD drawings, scientific formats, and compressed archives
- Handle multimodal content, including text, tables, diagrams, photos, videos, and audio, so that agents can reason using all evidence, not just text
- Preserve security by capturing access control lists (ACLs) and permissions from source systems, then enforcing them during retrieval and agent execution
- Avoid shortcuts, such as indexing only metadata or the first few pages of a document, which can hide critical information from agents

Retrieval: Using the Right Techniques for the Job

Successful enterprise agentic RAG uses a hybrid of at least five retrieval methods:

1. **Vector retrieval:** uses embeddings to match by meaning, finding semantically similar content even when different words are used.
2. **Keyword retrieval:** finds exact terms and phrases, essential for technical language, product codes, and regulatory clauses.
3. **Graph retrieval:** leverages a knowledge graph to explore relationships between entities (parts, suppliers, defects).
4. **Structured retrieval:** queries relational databases for precise numeric or categorical data (dosage ranges, warranty durations).
5. **Multimodal retrieval:** searches across images, diagrams, tables, audio, and video, especially important in engineering, life sciences, and field service.

Different business questions are best served by different retrieval combinations. A naive vector-only approach may work for small, narrow document sets, but it quickly fails in large, heterogeneous environments.

Naive and Sophisticated RAG

Most early RAG implementations were naive. In practice, naive RAG tends to fail in exactly the places enterprises care most about. For example, in a manufacturing context a safety engineer might get an answer that ignores a crucial test result because it was in a table the pipeline skipped, while in a regulated industry a lawyer might see a confident summary of a contract that omits a key amendment stored in a separate system.

These are not hallucinations in the model; they are retrieval failures. Sophisticated agentic RAG fixes these issues not by tweaking prompts, but by changing how content is chunked, which retrieval methods are used for which content, and how results are reranked and filtered before they reach the model.

> In many large organizations, early architectures consist of little more than an LLM plus a vector database. They can be helpful for demos but rarely form an adequate retrieval strategy for production use.

For truly effective enterprise AI search, organizations need agentic RAG that:

- Combines multiple retrieval methods in parallel, often through a neural and hybrid search pipeline
- Uses intelligent chunking and context windows that respect headings, sections, and semantic boundaries
- Applies query expansion and intent detection to capture what users actually mean, not just what they type
- Performs semantic re-ranking on results to push the most relevant content to the top
- Optimizes retrieval differently for different content types (for example, PLM documents versus customer emails)
- Enforces security at every step, including when agents chain multiple retrievals across tools

Done well, sophisticated agentic RAG becomes the primary driver of accuracy and reliability in assistant and agent behavior and is a core capability of enterprise agentic AI platforms.

CHAPTER 6

AI Assistants

Once grounded agentic RAG is in place, turning it into AI assistants is a natural next step. AI assistants take tasks that are manual, repetitive, and knowledge-heavy and transform them into automated workflows.

A few examples of this transformation include:

- Generating status reports from tickets, emails, and meeting notes for program managers ▪Preparing customer briefings from CRM data, contract histories, and historical support cases ▪Assembling technical overviews from past designs, test results, and field data for engineering reviews
- Drafting compliance documentation or regulatory responses by collating current policies and prior submissions

Assistants follow a predefined sequence of steps, calling retrieval, transformation, and external tools as needed. They offer:

- **Predictable behavior**, because their workflows are defined in advance and can be tested before deployment
- **Reduced time on task** for employees, who move from doing the work to reviewing AI-generated outputs
- **A steppingstone toward more autonomous agents**, because the boundaries of what the assistant can do are explicit

CHAPTER 7

AI Agents

As organizations become comfortable with assistants, they naturally ask why the system cannot also decide how to solve the task. That is where enterprise AI agents come in.

What Makes an AI Agent

An AI agent:

- Uses natural language and other inputs, often powered by generative AI, to understand goals and instructions from humans or systems
- Has a clear objective or set of objectives that go beyond answering a single question
- Plans a sequence of steps to achieve that objective, often using planning and reflection loops
- Uses tools, including retrieval, business systems, messaging tools, and external APIs, to carry out those steps
- Monitors progress and adapts the plan if needed (for example, by exploring new information or alternative tools)
- Decides when it is done or when it must stop and escalate to a human reviewer

Agents may or may not have long-term memory, may or may not be fully autonomous, and may run continuously or be triggered by events or users. The key difference is that the agent determines the workflow rather than following a prewritten one. This flexibility is powerful, but it also introduces risk, because generative models are nondeterministic and occasionally wrong; in long multi-step workflows even a small per-step error rate compounds into an unacceptably high chance that something goes wrong. You cannot make agents trustworthy by tuning the model alone, so reliability must come from the surrounding architecture, especially retrieval quality,
guardrails, observability, and governance.

CHAPTER 8

Agentic AI

Incomplex enterprises, there is no single agent that does everything. Instead, you have multi-agent systems: ecosystems of narrow, specialized agents that collaborate.

Examples of different types of agents include:

- A **customer verification agent** that handles identity checks and eligibility
- A **customer care agent** that orchestrates the main conversation and coordinates other conversations
- A **booking agent** that handles reservations or orders
- A **pricing agent** that checks offers, discounts, and approvals
- A **compliance agent** that ensures rules and regulations are satisfied before actions are taken

Agents may call one another directly or through an orchestration layer that coordinates roles, responsibilities, and handoffs. This brings tremendous capabilities, but also introduces new challenges, such as:

- Interactions can become complex and unpredictable as agents invoke one another
- Costs can spiral if agents call each other excessively or fall into loops
- Debugging errors requires fine-grained traces across agents, tools, and retrieval steps
- Security models become complex when data flows across multiple agents and systems

Enterprise-grade agentic AI therefore requires not only agents, but agentic orchestration. In other words, the ability to design, schedule, monitor, and control an ecosystem of agents from a central vantage point, with clear visibility into every step and call.

CHAPTER 9

Trustworthy Agentic AI

To deploy agents on critical business processes, CIOs, CISOs, and risk leaders increasingly insist that agentic AI be treated like any other critical system: observable, auditable, and governed. In practice, that means being able to answer hard questions, such as: Which agents can see customer PII and export it? Who approved those permissions? Can we reconstruct, step by step, what this agent did before it changed a record in SAP?

Trustworthiness in AI can be broken into four essential pillars: reliability, security, observability, and governance.

Reliable

Reliability means agents rely on comprehensive, high-quality enterprise knowledge and accurate retrieval, not guesses or outdated content. They operate within clear scopes and guardrails, handling errors or out-of-scope situations gracefully, including refusing tasks they cannot perform reliably; and they are regularly monitored, tested, and improved using metrics and traces.

This reliability is achieved through: content comprehensiveness and quality management (including enrichment and deduplication); sophisticated RAG and hybrid retrieval tuned to the enterprise domain; thoughtful prompt design and testing for both tasks and tools; reflection mechanisms (evaluating outputs before they are returned or actions are taken); and strong error and failure management, with clear categories and responses.

Secure

Security is non-negotiable in every enterprise organization. In a multi-agent world, agents may access information not permitted to the initiating user or business unit, data may be accidentally propagated from one agent to another across security boundaries, and tools may be invoked with unsafe or excessive permissions. This risk is amplified when tools are exposed through MCP servers.

A secure agentic AI system must: enforce end-to-end permissions from connectors to RAG to agents to tools and back to users; propagate access rights across all calls and handoffs; default to a closed posture, only exposing information when explicitly allowed; offer on-premises and private LLM options for highly sensitive data; and provide clear mechanisms to grant, restrict, and audit special permissions.

Observable

Observability in agentic AI includes the ability to: see how often agents are used and how successful they are across the organization; trace their workflows step by step, including all tool and agent calls and retrieval steps; track resource consumption (tokens, compute, memory, network) by agent, model, and department; monitor sensitive data usage and security enforcement; and receive alerts when failures, anomalies, or unusual behaviors occur, so teams can intervene quickly.

Without proper guardrails, it is easy for an autonomous agent to loop between systems for hours, increasing token costs and generating conflicting updates before anyone notices. A better pattern is an agent that hits a usage or error threshold, automatically pauses, and sends a notification to a human owner for review before continuing. Observability shows whether AI agents are adding value, where they may be failing, and how to optimize them. Without observability, you are effectively flying blind. And at enterprise scale, flying blind never ends well.

Governed

Governanceensures agents are used responsibly and cost-effectively. It typically covers:

- **Resource controls**, including token quotas, throughput limits, and priority settings so one project cannot starve others of capacity
- **Safety controls**, such as limits on tool call frequency and the ability to toggle agents and tools on or off in response to issues or policy decisions
- **Financial and risk controls**, including budgeting mechanisms tied to business value and human-in-the-loop checkpoints for high-risk actions

Without these controls, a single pilot can consume a disproportionate share of tokens and API calls, leaving other teams throttled or forcing emergency budget cuts. Governance is where observability becomes control; it is how enterprises keep agentic AI aligned with strategy, policy, and economics rather than letting it evolve in an uncontrolled way.

CHAPTER 10

Enterprise Agentic AI Platforms

Architecture Requirements

Enterprise-grade agentic AI cannot be stitched together from a few APIs and a vector database. These architectures frequently fail as soon as teams try to enforce real permissions, support more than one or two business units, or explain agent behavior to a CISO, because none of those concerns live in the LLM or the vector database.

Organizations need platforms that bring multiple capabilities together, including:

- **Connectors and ingestion** for all critical systems and formats, including cloud and on-premises repositories
- **Enrichment** to normalize, annotate, and structure content for downstream AI, including entity extraction, knowledge graphs, and taxonomy alignment
- **Sophisticated RAG** with hybrid retrieval that treats enterprise search as a shared service for many assistants and agents at once, rather than spinning up a separate, naive RAG stack for every new use case
- An **agent framework** with visual orchestration, prompts, LLM management, and tool libraries, so that teams can design and evolve assistants and agents without requiring low-level coding capabilities
- **Observability and governance** across agents, tools, models, and knowledge flows, supported by dashboards, traces, and policy controls
- **Security** that mirrors and enforces source-system permissions everywhere, including across multi-agent interactions. That security model must extend to MCP servers and other tool-exposure layers, so that every agent call, tool invocation, and cross-system hop is subject to the same access controls and audit trails

Architectures that ignore these concerns often look elegant on a slide and quickly run into issues when a CISO, a regulator, or a second business unit asks basic questions about security and traceability. Trying to scatter retrieval into every application, or to centralize all agents in a single monolith, almost always leads to brittle integrations and blind spots.

The most robust architectures centralize retrieval, security, and observability, but keep agents and tools close to the systems where work actually happens.

In short, an enterprise agentic AI platform is the unified layer that connects enterprise systems and data, grounds models with sophisticated RAG, and provides the secure, observable orchestration needed to run assistants and agents at scale. This platform becomes the enterprise brain on which assistants and agents are built, powering a broad range of workflows, from conversational search to fully autonomous agentic processes.
The organizations that succeed with agentic AI treat this platform as shared infrastructure, on par with their data warehouse or Enterprise Resource Planning (ERP) system, not as a sidecar to a single chatbot project.

CHAPTER 11

How Leading Enterprises Are Deploying at Scale

Experience shows that only a small fraction of enterprises moves beyond pilots to deploying agentic AI at scale. Those that succeed tend to follow this blueprint:

- **Start with focused, high-value use cases** where success can be measured, such as engineering research, warranty analysis, clinical trial document review, or regulatory responses
- **Invest in data readiness and retrieval before building flashy demos.** Unify access to critical knowledge sources and stand up a strong RAG layer
- **Build assistants and narrow agents first**, where scope is tight and risk manageable, then expand to multi-agent ecosystems
- **Layer in observability and governance early**, not as an afterthought, so that agent behavior is visible and controllable from day one
- **Partner with experienced platforms and vendors** rather than building everything in house, to benefit from mature connectors, retrieval pipelines, and agent frameworks

In practical terms, the first six to twelve months of an enterprise agentic AI rollout usually mean three things: identifying a handful of high-value, document- and data-heavy workflows, such as engineering research, warranty analysis, or regulatory response; standing up enterprise AI search and RAG on the repositories that matter most; and piloting one or two assistants with clear success metrics before introducing more autonomous agents. Typical success metrics in this phase include time to answer key questions, percentage of workflows successfully handled by assistants on first pass, and weekly active usage in target teams.
Rather than wiring LLMs directly into every workflow, organizations that succeed focus on getting these foundations right, treating enterprise AI search as a shared knowledge fabric, running robust RAG and grounding, and building a secure, observable agent platform with a portfolio of well-scoped agents that solve specific, high-value problems.
With these pieces in place, assistants and agents become durable infrastructure for how the organization searches, reasons, and takes action.

When you evaluate platforms, look beyond models and vector stores. Check whether connectors cover your real systems, whether security and permissions are enforced end-to-end (including tools and MCP servers), and whether you can actually trace and explain agent behavior to a CIO or CISO.

A practical next step is to assess where you are on this journey today, identify gaps in data readiness, retrieval, and platform capabilities, and then evaluate platforms that can close those gaps. Sinequa's enterprise agentic AI platform combines enterprise AI search, advanced RAG, and secure agent orchestration along these lines, so you can move from pilots to production with confidence.

Explore customer stories at sinequa.com/customers/customer-stories.

Ready to See Enterprise Agentic AI in Action?
Talk to our team to see how Sinequa by ChapsVision can help your organization move from AI experiments to trusted, scalable enterprise AI agents.

Book a demo →

Sinequa
by ChapsVision

For more information, please visit

Enterprise Agentic AI platform

About Sinequa by ChapsVison
Sinequa transforms how work gets done by providing employees with knowledgeable, accurate, and secure AI Agents that streamline workflows, navigate complex enterprise data, and deliver reliable, traceable insights. By combining enterprise search with generative AI in a configurable, easily managed framework, Sinequa enables organizations to deploy out-of-the-box Agents or tailor specialized workflows, ensuring every interaction is trusted, governed, and compliant while empowering employees to focus on high-value work. For more information, visit **www.chapsvision.com/platform/sinequa/** or **www.sinequa.com**