

ENTERPRISE **AI World**

From AI Readiness to AI in Action

*Your Guide to the
Key Technologies,
Tools, and Strategies*



SOURCEBOOK

2026



AI Doesn't Need More Data. It Needs Active Context

Give AI systems live, connected, governed, and trusted enterprise data at the moment it is needed

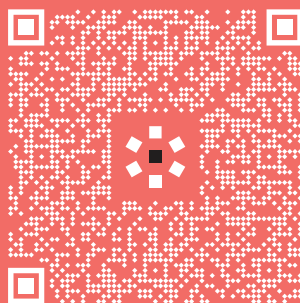


Denodo is the AI data layer

CONNECT • DELIVER • GOVERN • UNDERSTAND

Build the foundation for trusted enterprise AI

denodo.com



denodo

ENTERPRISE AI - EBOOK

The Active Context Layer for Enterprise AI

How to deliver trusted, governed, and operationally aware context for AI agents

© 2024 Denodo Technologies

DOWNLOAD THE EBOOK



EDITOR'S NOTE

AI Takes Hold Within the Enterprise

In this year's **Enterprise AI World Sourcebook**, we look at the elements that contribute to the success of AI within organizations. Simply throwing AI against the enterprise wall is far from sufficient to guarantee acceptance and real-world utility.

CO-EDITORS Marydee Ojala and Stephanie Simone look at the power of retrieval-augmented generation (RAG) from two different perspectives. Ojala recognizes how having access to enterprise knowledge empowers RAG to avoid hallucinations and provide actionable, accurate, and up-to-date information that fuels business-critical outcomes. Simone's attention is caught by GraphRAG and its ability to enhance reasoning by using connected data, semantic relationships, and graph-based retrieval. Ojala also touts the importance of AI-ready data.

RAG is not the only technology affecting the success of AI within the enterprise. Joe Hilger, Lulit Tesfaye, and Zach Wahl, from Enterprise Knowledge and authors of the 2026 book *Bridging Knowledge, Data, and AI: Harnessing the Semantic Layer Framework to Drive Intelligence*, consider the importance of semantic layers. They detail the various components of semantic layers and their business value.

Serious Insights' Daniel W. Rasmus takes on the topic of how to orchestrate agentic AI systems and workflows that humans can trust. He notes that agentic AI is not currently in use by most consumers; it's a work technology that introduces new factors into coding. Creating agents for very specific tasks is not inherently difficult, nor is coordination among agents, but implementing agentic AI at scale is.

Speaking of scale, Seth Earley, founder of Earley Information Science, explains the importance of reusable data products for AI at scale, noting that a move from "project thinking" to "product thinking" helps. AI raises the bar for data quality since it does not always flag ambiguities. He also provides very useful case studies about creating data products.

Judith Lamont focuses on the importance of trust. Without trusted data foundations, all the training of AI agents is wasted. She provides an example of trust from a healthcare perspective, then goes on to detail some aspects of AI devel-

opment, including metadata, evaluation of system performance, design elements, and ownership clarity.

Jelani Harper concentrates on the value that real-time data has for AI systems. The speed at which today's business and technological environments evolve is almost hard to fathom, but this must be a consideration. Look at the data layer, ingestion, data availability, schema mechanisms, memory of models, and streaming data.

On the legal side of enterprise AI, Colin Levy notes that potential risks can differ depending on discrete business processes. While AI should be governed for security, compliance, and trust, there's a real question concerning access. He recommends that policies be explicit and spell out exactly where AI can make decisions and where humans should approve intended activities.

There is little doubt that AI will continue take hold within the enterprise. New models will appear. Advances in data quality, trustworthiness, and real-world applications will materialize. AI-native teams will emerge. The relationship between humans and AI in the workplace will evolve. Staying on top of this rapidly fluctuating, fast-changing world will remain challenging. One thing that remains as important as ever is having a human-in-the-loop when it comes to applying and maintaining these systems.

One excellent way to stay au courant is to attend the Enterprise AI World conference (enterpriseaiworld.com/Conference/2026) in Washington, D.C., Nov. 17–19, 2026. With Driving Real AI Adoption Across the Enterprise as its theme, it offers an immersive look at AI by diving into key technologies and strategies. As AI increasingly becomes part of the operational fabric of modern enterprises, speakers at the conference will share their knowledge, experiences, and practical insights about building AI initiatives, resilience, and teams. We hope the articles in this Sourcebook will spark your own ideas and projects, leading to successful business outcomes.

Marydee Ojala and Stephanie Simone
Co-Editors

AI Does Not Have a Data Problem. It Has a Context Problem.



Most enterprises already have more than enough data to support AI.

What they do not have is a consistent way to tell an AI system which data is relevant, what it means, whether it is current, how it relates to other information, and whether the person or agent requesting it is allowed to use it.

That distinction matters. Giving an AI system access to more data does not necessarily improve its decisions. It can just as easily introduce conflicting definitions, stale information, duplicated records, and content separated from the business rules that make it useful.

The next challenge in enterprise AI is therefore not simply data access. It is the ability to deliver active context: connected, live, governed, and semantically meaningful enterprise information at the moment an AI system needs it.

This is the role of an AI data layer.

From Project-based Integration to Shared Infrastructure

Many AI initiatives still assemble context one application at a time. Data is extracted from selected systems, copied into a new repository, enriched for a specific use case, and governed through controls built around that application.

This may be sufficient for a pilot. But as organizations deploy more copilots, agents, retrieval workflows, and AI-enabled applications, the same approach begins to accumulate technical debt. Each new initiative can add another pipeline to maintain, another copy of the data to reconcile, another set of policies to manage, and another interpretation of business meaning.

Over time, that complexity slows the delivery of new AI use cases, increases governance risk, and makes it harder to maintain consistent context across the enterprise.

An AI data layer offers a different operating model. Rather than rebuilding the connection between AI and enterprise data for every use case, it provides shared infrastructure for delivering trusted context across distributed environments.

Four Requirements for Active Context

Creating active context requires more than connecting an AI model to a collection of sources. An effective AI data layer should provide four capabilities.

Universal connectivity

Enterprise context is distributed across cloud platforms, lakehouses, warehouses, operational systems, SaaS applications, and on-premises environments. AI systems need access across these environments without requiring every source to be migrated into a single platform.

Zero-copy delivery

Context should remain current. Providing live access where appropriate reduces the need to create and maintain a new data copy for every AI use case, helping limit latency, duplication, and inconsistency.

Governed access

Governance must continue to apply when data is consumed by AI. Access should reflect identity, role, purpose, policy, and business context at runtime, rather than being enforced only within the source system.

Semantic trust

AI systems need business meaning, not just fields and records. Shared definitions, relationships, metrics, and terminology help AI distinguish between similar concepts and interpret information consistently.

Together, these capabilities support governed enterprise data products



that can be reused across AI, analytics, applications, and business processes.

Making context an enterprise capability

The goal of an AI data layer is not to replace existing investments. Lakehouses, warehouses, catalogs, governance platforms, and AI tools all continue to play important roles.

The AI data layer connects these environments and makes their data more usable by providing a consistent way to deliver active context across them.

Denodo applies this architectural approach by connecting distributed enterprise data, delivering it without unnecessary replication, enforcing governance, and preserving semantic meaning. The broader principle is more important than any individual product: as AI becomes embedded in everyday operations, context can no longer be assembled separately for every use case.

It must become shared enterprise infrastructure.

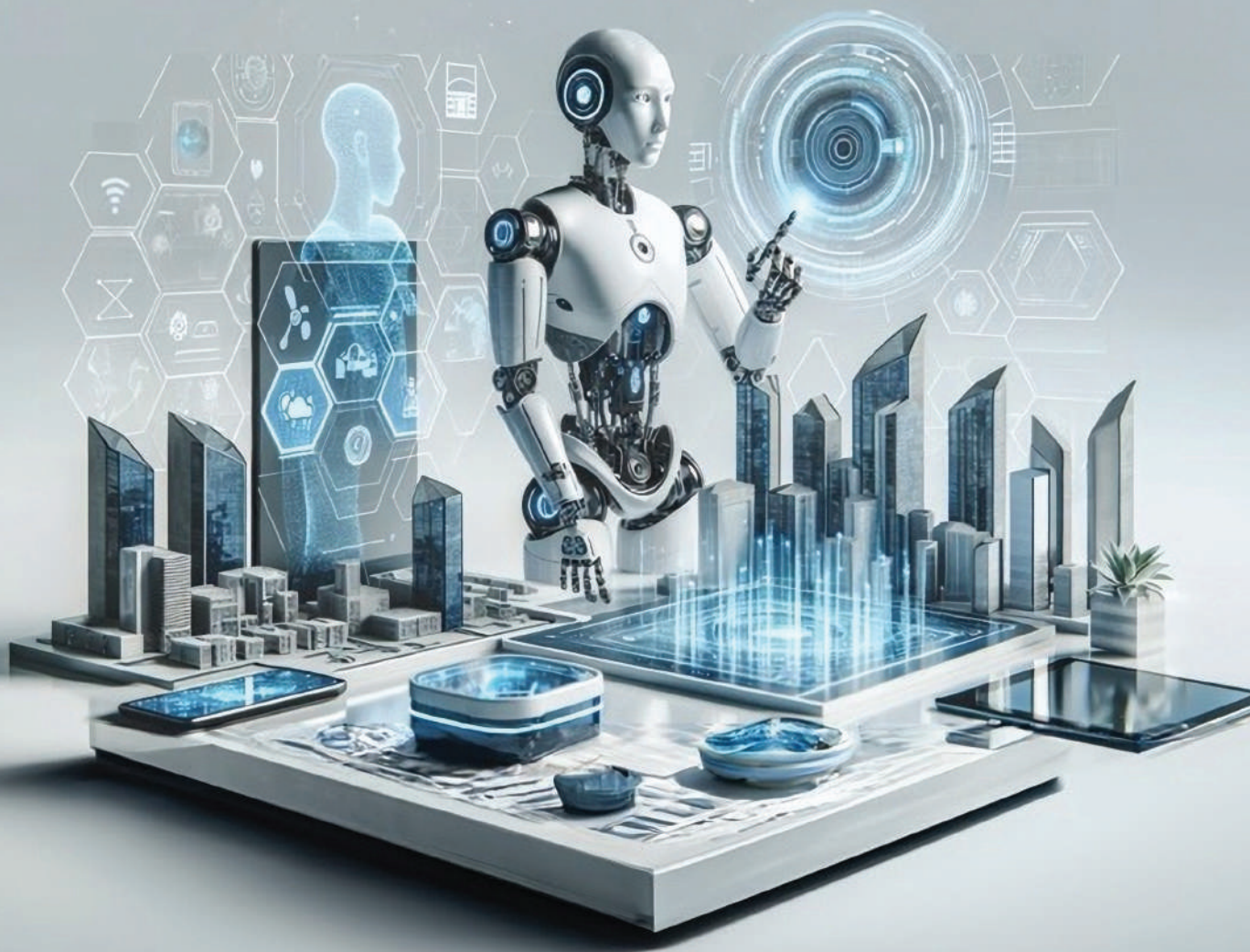
Want to go deeper? [Download the eBook](#) to learn how a context layer can deliver the live, governed, semantically meaningful information enterprise AI needs.

DENODO

www.denodo.com

Improving AI Outcomes Through Context Engineering

By Marydee Ojala



Once upon a time, and that time wasn't all that long ago, we heard that content was what mattered.

Then AI entered the picture, and enterprises shifted from content to context. Content by itself did not lead to relevancy. AI, lacking context, could all too easily misinterpret what was needed. Is a “recent” financial report one that is 1 month, 6 months, or a year old? Is a “current customer” one who has made a purchase in the past quarter or one who maintains a subscription to the company blog? Acronyms pose their own brand of problematic interpretations. Is APR annual percentage rate, annual performance report, or April? What are the business rules that determine context, and how frequently are they maintained, updated, and re-examined?



Context in real life is often intuited by human beings. When the preschooler says, “I want an apple,” you know the request is for a snack and are delighted it’s fruit that is desired rather than candy. Fast-forward a few years. The teenager who whines, “I want an Apple,” is rejecting your choice of a Windows 11 machine and could care less about apple the fruit, preferring to snack on junk food while using his Mac. Intuition works well in human interactions. It doesn’t work that well for enterprise AI.

I’m reminded of William Carlos Williams’ poem about the red wheelbarrow upon which so much depends (poetryfoundation.org/poems/45502/the-red-wheelbarrow). It was published in 1938. Williams never had to deal with enterprise AI, never asked a chatbot to search a large language model (LLM) for information about wheelbarrows, and never had to disambiguate a red wheelbarrow from a blue wagon. What is lacking from the poem is context—which literary scholars have lined up to supply. Analyses of the poem are vastly longer than the 16 words of the poem itself. And Williams doesn’t clarify what it is, exactly, that depends upon the wheelbarrow (or what role the chickens play).

Perhaps it was just an observation with no additional meanings. Sometimes, a cigar is just a cigar, according to Freud. The same could be said of wheelbarrows. Over-analysis may be an occupational hazard for literary critics, but it’s not entirely unfounded for ensuring that enterprise LLMs get the context right when training agents, particularly when you may have hundreds of agents dealing with multiple, specialized activities and responsibilities. Their ability to communicate with each other makes ensuring that all the agents understand the same context that is imperative for smooth functioning. Getting the analysis of context correct for every possibility within an enterprise leads people to trust the technology.

As agentic AI moves into many enterprise operations, determining overall context along with context for individual agents becomes an important element for successful implementations. Plus, the topic of hallucinations that are delivered with utmost confidence even though they convey incorrect information and data has been widely reported and explored. One solution lies in context engineering, which many view as the key to a truly reliable enterprise AI.

CONTEXT ENGINEERING VERSUS PROMPT ENGINEERING

Context engineering is not precisely the same as prompt engineering. And prompt engineering has often been miscast as simply entering a query into a search box. Advice about best practices for prompting LLMs is rampant and often contradictory. The fact is that much prompting is situational: it’s not a “one size fits all” approach. When someone tells you that they excel at prompt engineering, what they probably mean is that they are very good at translating an information research request into a prompt query that will match what a human wants with what resonates with an LLM.

According to Anthropic (“Effective Context Engineering for AI Agents”; anthropic.com/engineering/effective-context-engineering-for-ai-agents), “Context engineering represents a fundamental shift in how we build with LLMs. As models become more capable, the challenge isn’t just crafting the perfect prompt—it’s thoughtfully curating what information enters the model’s limited attention budget at each step.”

Seeing context engineering as a logical extension of prompt engineering, Anthropic distinguishes between context engineering and prompt engineering “Prompt engineering refers to methods for writing and organizing LLM instructions for optimal outcomes. ... Context engineering refers to the set of

Enterprises adopting agentic AI are charged with training not just a few, but often hundreds of, agents, each assigned a specific task and objectives.



strategies for curating and maintaining the optimal set of tokens (information) during LLM inference, including all the other information that may land there outside of the prompts.”

ESSENTIALS OF CONTEXT ENGINEERING

In its “Context Engineering: A Practical Guide for AI Agents” (sourcegraph.com/blog/context-engineering), Sourcegraph views the relationship between prompt engineering and context engineering somewhat differently from Anthropic: “Prompt engineering focuses on how to phrase and structure instructions for the LLM to generate the best results, while effective context engineering is about designing the entire system that feeds the model the right context at the right time, including all the context across multiple interactions.”

With context engineering, the scope is considerably more extensive than with prompt engineering, covering a “full set of tokens at inference time,” says Sourcegraph. While prompt engineering looks at optimizing the phrasing of queries and strives for fewer ambiguities, context engineering looks at lessening the amount of noise that results from a query to heighten the signal, essentially leading to better, more targeted responses. Ideally, there’s a “platform team building the agent pipeline,” rather than individuals entering prompts. Context engineering considers actual instructions, documents previously retrieved, and output schemas. History is thus an important element in the context puzzle.

This is not to say that context engineering makes prompt engineering obsolete. Each has its place. However, it’s important to distinguish between how the phrase “prompt engineering” is used in common parlance and how both it and the “context engineering” phrase appear in more technical environments. In Sourcegraph’s view, “If you’re swapping nouns and adjectives, you’re still doing prompt engineering. If you’re changing what data the agent retrieves, in what order, with what re-ranking, and what gets evicted when the context window fills, you’re doing context engineering. Prompt engineering is essential for one-off tasks, but context engineering is what matters for complex tasks and agent systems that maintain conversation history and pull in external data across many turns.”

In the early days of background prompt engineering, the tweaking, modification, and model training that happened was to ensure that responses to queries were relevant. That is still important, but perhaps more importantly, it is also where guardrails could and should be implemented, so that confidential information is not leaked to those not authorized to view it. It’s necessary to measure context quality against retrieval metrics and to think about the potential security risks, including data leakage and system prompt extraction defense.

Context engineering is responsible for making the language models reliable for agentic AI as well as other AI technologies while retaining the guardrails. Context here is important as well. Categories of employees, such as summer interns, retired, or top management should have different levels of access, or none at all, depending on the context of their category.

This isn’t as simple as intuiting what apple is wanted or knowing that it’s apples not oranges when it comes to choosing fruits; it’s understanding the broader picture. Take coding, for example. Advanced LLMs from companies such as Meta, Anthropic, OpenAI, and Google promise to dramatically shorten the time it takes to code a new project or an add-on feature to an existing product. To do this effectively, however, the code must sync with existing code. If it’s not compatible, the project will fail.

This is another context issue. The LLM must know the coding context to create workable new code. Humans who are coding experts must review the AI-created code to not only ensure compatibility but also to determine accuracy. Coding mistakes can sometimes be difficult to pinpoint, and simply asking AI to troubleshoot itself does not always work.

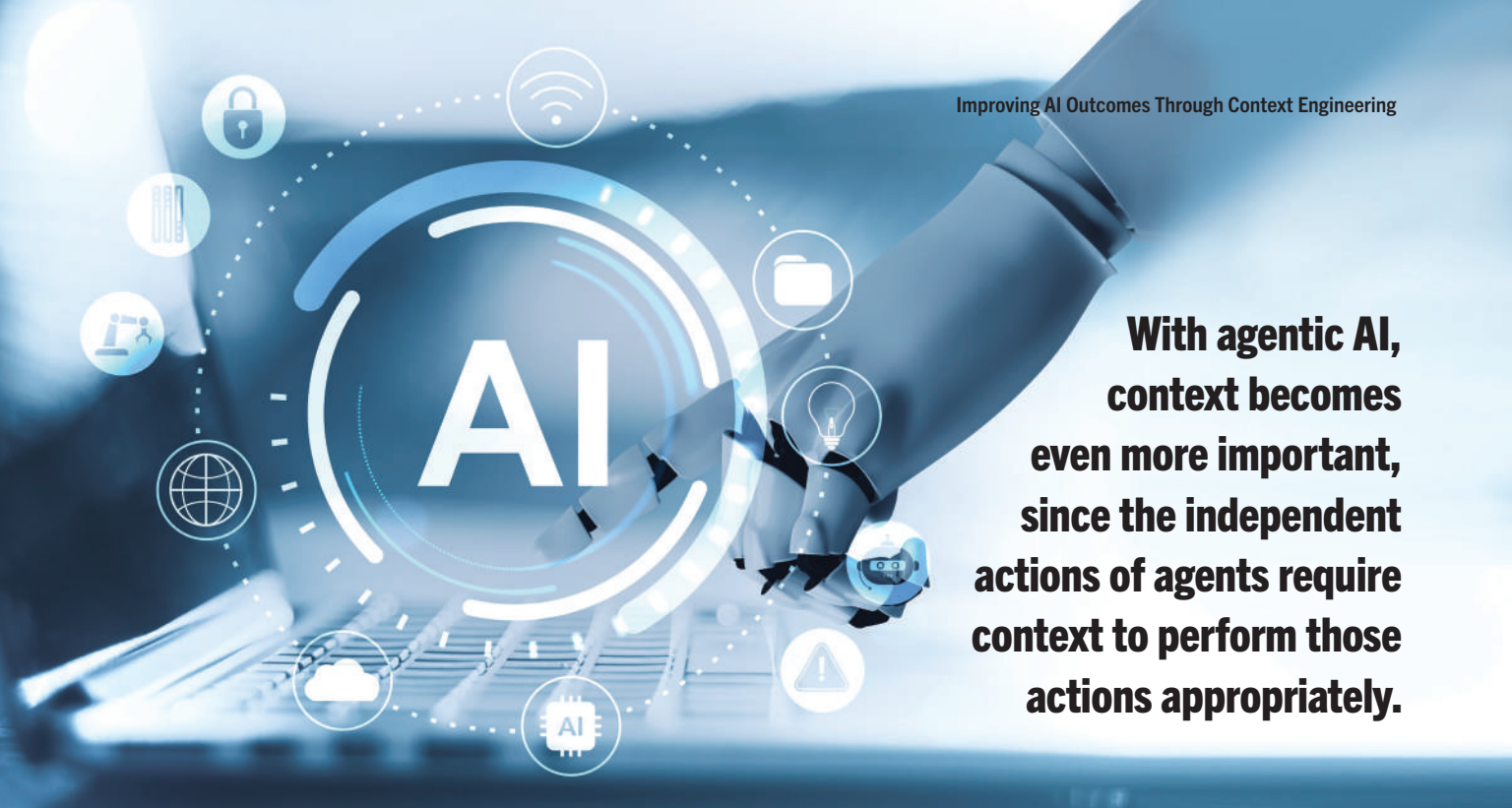
TRAINING MODELS FOR AGENTIC AI

Enterprises adopting agentic AI are charged with training not just a few, but often hundreds of, agents, each assigned a specific task and objectives. It could be staffing, compliance, logistics, or research. Agents are autonomous and proactive, not waiting for guidance from humans to perform their tasks and achieve their objectives. They communicate with each other and learn from changes in the business environment. Suppose a problem arises with the apple crop. Agents can redirect shipping to accommodate the late, or early, harvesting.

On the LangChain blog (lanchain.com/blog/context-engineering-for-agents), it’s phrased this way: “Context engineering is the art and science of filling the context window with just the right information at each step of an agent’s trajectory.” It goes on to list the context types that context engineers need to manage:

- ▶ Instructions – prompts, memories, fewshot examples, tool descriptions, etc
- ▶ Knowledge – facts, memories, etc
- ▶ Tools – feedback from tool calls





With agentic AI, context becomes even more important, since the independent actions of agents require context to perform those actions appropriately.

Many technologies can be included in the context engineer's toolbox, such as classic retrieval software, semantic search, vector databases, retrieval-augmented generation, linked data, knowledge graphs, and metadata, along with other legacy technologies designed to connect AI with trusted enterprise knowledge and real-time business context. It's not a total reinvention of the wheel (or wheelbarrow), but it is looking at how these older technologies interact to enhance the understanding of context.

Although not exactly rule-based in the traditional sense of the term, agent decisions do need to be grounded in the business logic of the individual enterprise. At what point do the apples too ripe for grocery stores become eligible to be sold to a food manufacturer for applesauce? The underlying information architecture and business rules enable the dynamic automation of existing business processes and the ability of agents to notice and solve problems. This can be situational. How much autonomy, for example, do agents have in making pricing decisions will be a decision made differently from one company to another.

CONTEXT HELPS WITH HISTORY

With agentic AI, context becomes even more important, since the independent actions of agents require context to perform those actions appropriately. Context determines not only what they should do today, but also what changes have occurred that would affect the actions. In that sense, context engineering for LLMs relies on history, but must also have human input to identify why that history may no longer be relevant. Agents should not upset the apple cart, but sometimes the apple cart upsets itself due to weather, a pot-holed road, or bad loading decisions. Context is the answer.

In a perfect world, context engineering would solve relevancy problems. That is not always the case. As a vision, a situation to strive for, it's great. Not every enterprise is at the same level of expertise regarding context engineering. They are at different stages of development and have differing resources available for context engineering.

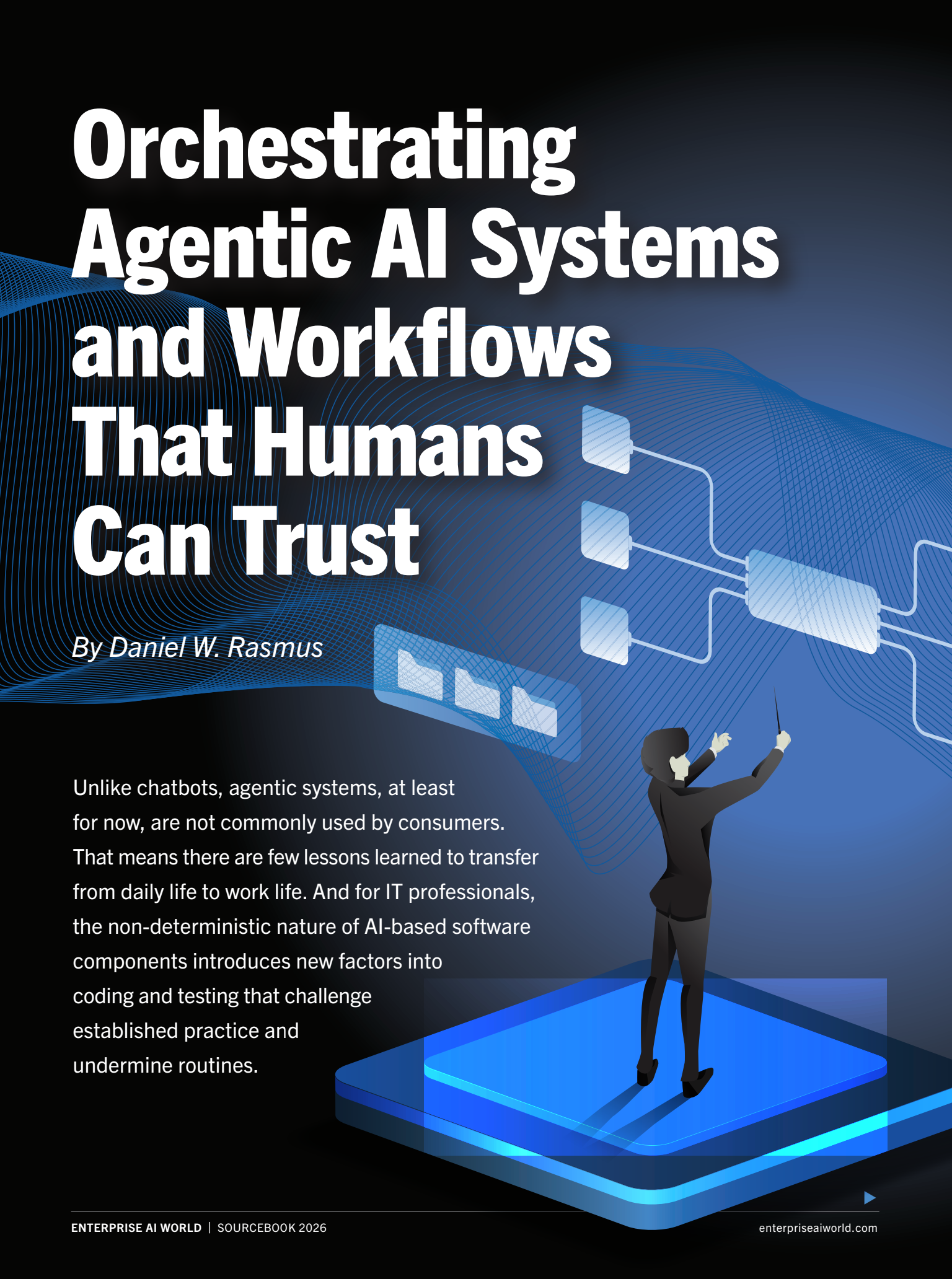
Enterprise AI is evolving fast. Still, too many organizations struggle to deliver accurate and trustworthy results, often because the context has not been implemented at scale. Without context, enterprise AI is essentially working without a firm knowledge of what constitutes accuracy. It's as if they had obscured access to context and blindfolded their engineers. Investing in context engineering will eliminate bottlenecks that threaten to lengthen, rather than shorten, the time it takes to achieve business objectives. Context engineering turns LLMs into effective partners, opening up new opportunities.

As AI moves from experimentation to production, the focus shifts from concentrating on content to determining context. Context engineering allows organizations to improve AI outcomes by delivering richer context, memory, retrieval, and business knowledge to AI systems and agents.



MARYDEE OJALA is an editor at *Enterprise AI World* and editor-in-chief at *KMWorld*. She has programming responsibilities for Information Today, Inc.'s Enterprise Search and Discovery and Data Summit conferences.

Orchestrating Agentic AI Systems and Workflows That Humans Can Trust



By Daniel W. Rasmus

Unlike chatbots, agentic systems, at least for now, are not commonly used by consumers. That means there are few lessons learned to transfer from daily life to work life. And for IT professionals, the non-deterministic nature of AI-based software components introduces new factors into coding and testing that challenge established practice and undermine routines.

While the adoption of agents has been touted as the next level of AI automation, perhaps the truly transformative one, the reality inside organizations is proving difficult to apply at scale. It isn't difficult to create an agent capable of performing a task. Developers, however, are finding it challenging to design multi-agent systems that integrate several models, data sources, business applications, and people without introducing small errors, stale information, missing permissions, or failed integrations that threaten to cascade into larger problems.

Orchestration must be viewed not simply for coordinating AI agents, but as part of the enterprise operating model.

Recent enterprise research points in the same direction. Microsoft's "2026 Work Trend Index Annual Report" (news.microsoft.com/annual-work-trend-index-2026) argues that organizations increasingly organize work across people, agents, and the systems connecting them, increasing the value of judgment and orchestration. The Stanford Digital Economy Lab's "Enterprise AI Playbook: Lessons From 51 Successful Developments" (digitaleconomy.stanford.edu/publication/enterprise-ai-playbook) recommends treating models as interchangeable components within an enterprise-controlled orchestration layer. Together, the studies suggest that successful agent-based systems derive from the workflow, context, integration, governance, measurement, and learning system surrounding them.

START WITH THE WORKFLOW, NOT THE AGENT

The temptation with agentic AI is to start by asking what agents can do. It's better to start with design questions, such as what work needs to be done; how it is currently accomplished; where judgment, data, and accountability enter the process; and most, importantly, what a good work product looks like.

Ryan Ries, chief AI and data scientist at Mission Cloud, argues that organizations need to map "the entire workflow: every step, every gate, every handoff, including the undocumented ones" before automating it. "Teams assume they know their own processes and they usually don't," he said.

That observation echoes one of the Stanford study's broader findings: Process documentation, data access, and change management are not ancillary work. They are often prerequisites for successful deployment. Organizations that treat them as part of the implementation reach production faster than those that assume an AI model will compensate for an ill-defined process.

Once the workflow is understood, decomposition becomes critical. Asawar Ali, founder of Link Building Agency, uses what he calls a "contract-based" approach, in which every agent receives a clearly defined task and expected output. "If this is not done, then the agent will go off on a tangent, which will end up ruining the whole process."



Ries described a similar architecture: “Every production multi-agent system we’ve built ends up needing a supervisor pattern, where one agent breaks down the task, specialist agents execute narrow pieces, and a validator checks the output before it reaches a human or another system.”

Production systems therefore favor narrow specialists grounded in operating procedures and rules. As Shelley Copsey, CEO and co-founder of FYLD, has observed in infrastructure and fieldwork deployments, these agents are often most useful when monitoring bounded work for exceptions rather than attempting to manage an entire process.

That architecture also helps contain failure. An agent with a bounded task, explicit inputs, defined outputs, and limited authority creates a smaller blast radius when something goes wrong. Once work is divided among bounded agents, the reliability of the system increasingly depends on what happens at the handoffs between them.

MAKE FAILURES VISIBLE BEFORE THEY BECOME CONSEQUENCES

Agent orchestration diagrams usually emphasize reasoning: One agent delegates, another retrieves information, a third performs analysis, and perhaps another checks the answer. In production, however, the connections between those boxes often determine whether the workflow works at all.

“The biggest pains are always at the integration points,” said Kris Zyp, SVP of engineering at HarperDB. “The market tends to see amazing demos of individual solutions and assumes they will all fit together nicely. Bringing vector search

together with robust access and governance, plus the essential business logic and auditing, is where the real work and the real innovation is.”

Failures are not always visible. Phillip Hamnett, CEO of TalentAid AG, discovered that his team’s biggest operational problem was not agents producing bad results. Some scheduled jobs simply stopped running. “We were watching for agents doing bad work, and what was really happening was no work at all,” Hamnett explained. His team moved the jobs to a server with a watchdog and required every run to report its status. Silence became an observable failure condition.

Hamnett’s experience broadens the definition of observability. A production system should record whether each workflow started and completed; the data, sources, and tools used; timing and handoffs; validation results; exceptions; and the business outcome. These signals must arrive during execution. This does not require exposing literal internal reasoning, but it does require traceability. Otherwise, a team may know that an action occurred without understanding why, or discover a failure only after flawed data and recommendations have propagated.

The orchestration layer therefore needs the equivalent of instrumentation in distributed computing. AI may introduce probabilistic behavior, but the system around it cannot be allowed to become probabilistically observable.

Observability, however, is only half of the control system. Once an organization can see where failure occurs, it must decide when a human should intervene. Human-in-the-loop orchestration is often implemented as a final approval

Key Takeaways

- Treat orchestration as part of the enterprise operating model, not merely as a way to coordinate agents.
- Instrument every important handoff, make failures traceable, and treat silence or missing work as a failure condition.
- Map the complete workflow, including gates, handoffs, judgment, and accountability, before introducing automation.
- Assign agents bounded tasks, explicit inputs and outputs, and only the authority required to perform their roles.
- Place human intervention according to the consequences of an error, not simply on the complexity of the task.
- Maintain context as shared infrastructure, and apply identity, permissions, approvals, and audit controls to every action.
- Measure accuracy, timeliness, and business outcomes while keeping humans accountable for the system’s performance.

box. That may be the wrong place for the human. If an early error propagates through several agents, final review may come only after the workflow has produced a coherent but unreliable result.

Mia Morin, editor-in-chief at Intimeros, learned this after outdated information entered an agent research workflow and spread into a draft before fact-checking caught it. Her team moved the human checkpoint earlier. Morin's rule is simple: "Do not place the human review at the final handoff, but at the first handoff, [so] small errors are detected when they are small."

Yet, review is not necessary at every handoff. Yury Byalik, founder of CivilCase.com, encountered demand letters containing incorrect filing deadlines in three states—errors that could have resulted in cases being dismissed. His conclusion was that the trigger for human involvement should be consequence rather than complexity. "Complexity is not the determining trigger for a human review. Rather, it is the impact of a wrong output that determines when a human must participate in that part of the process."

Organizations should consider implementing consequence mapping, which evaluates each handoff on four dimensions: the impact of an error, the likelihood that it will be detected downstream, whether it can be reversed, and how far it could propagate. The same analysis should cover work that is late, incomplete, or never produced.

Low-impact, recoverable decisions may operate autonomously if their execution and outcomes are recorded. High-impact decisions require stronger controls, such as automated validation, escalation, human review, or explicit approval. Legal documents, regulated work, and other zero-error-tolerance activities justify heavier involvement.

Effective orchestration requires visibility across the workflow and selective intervention at points of consequence. Organizations should instrument important handoffs, treat silence as failure, and establish escalation paths before deployment. The point is not to place a person everywhere. It is to place human judgment where failure becomes consequential. Even well-instrumented workflows, however, will fail when agents work with incomplete, inconsistent, or outdated information.

CONTEXT IS INFRASTRUCTURE

Agents also fail when the context available at the moment of action, including data, instructions, rules, and institutional knowledge, arrives incomplete, inconsistent, outdated, or poorly governed.

Laura MacGregor, founder of Savvy Marketing Works, approaches AI as if it is an intern accumulating institutional knowledge. Her workflows draw on structured information about her business, services, background, clients, and writing style. "It's as good as its context," she noted. That context cannot simply be dumped into a prompt. It needs to be maintained.

Hamnett initially discovered that individual machines accumulated different operational memories. His solution was to move rules, playbooks, and lessons from previous runs into tracked files that every agent reads at startup. "Local agent memory is fine as a secondary copy; it's just never the source of truth."

Steve Karp, chief innovation officer at Unanet, made the same point about enterprise data. "It must be fresh, accurate, and relevant or else your AI-generated outcomes will be useless."

Research from Hex (*The Data Leader's Playbook for Agentic Analytics*; <https://hex.tech/resources/data-leaders-playbook-for-agentic-analytics>) about agentic analytics frames context engineering as an enterprise capability encompassing trusted data, semantic models, business terminology, data quirks, and workflow preferences. The goal is not simply to give agents more information, but to direct them toward approved information and impose rigid rules where accuracy requirements demand them.

HarperDB's Zyp points to an additional problem: Agents do not always know what they need to retrieve. "Most agent mistakes, when they are not [due to] a lack of human-level judgment, come from falling into the gaps between memory and retrieval."

Context must therefore become operational infrastructure, with clear ownership, versioning, freshness standards, evaluation, and maintenance. While governance determines which information and actions agents are permitted to use, context engineering determines what information agents can employ.

GOVERNANCE HAS TO TRAVEL WITH THE WORK

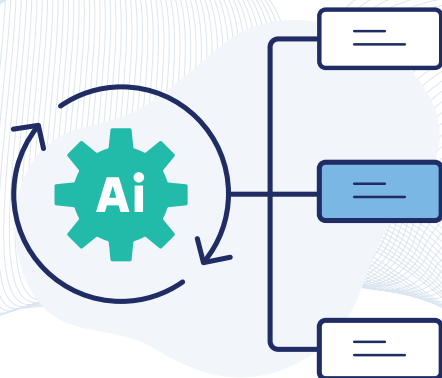
As agents gain access to business systems, governance becomes inseparable from orchestration.

John Moore, founding partner at Tristella Advisors, encountered this while helping deploy a multi-agent business intelligence system. The agents could query databases, a data warehouse, and technical documentation, but the requirements did not define authorization boundaries. During staging, the team discovered that users could retrieve sensitive information they were not permitted to see. "The lesson is that governance is not an afterthought. Governance in multi-agent systems is complex and compulsory," he emphasized.

Ries bumped into the same issue in an analytics implementation and developed a dual-token architecture: "One token establishes the user's identity while another scoped token authorizes each outbound action. The agent never acquires more authority than the person for whom it is acting."

That principle aligns with emerging Model Context Protocol security guidance from Wiz ("Model Context Protocol Security: A Practical Guide for Teams"; wiz.io/academy/ai-security/model-context-protocol-security), which recommends tightly scoped credentials, distinct identities for agents and human administrators, explicit approval mechanisms, and centralized audit logging. As agents gain the ability to





As agents perform more workflow steps, human work will shift from execution toward system design and accountability.

act through connected tools, identity and authorization have to apply to every action rather than simply to the agent as a whole.

The same principle should apply to autonomy. Hamnett says his team stopped treating autonomy as “a single dial per agent.” Instead, permission is granted by action type. Some outputs may be drafted automatically but sent only by a person; others may execute autonomously within strict limits, with thresholds and kill switches.

The governing question is no longer simply “Can this agent act?” It is “Under these circumstances, with this data, on behalf of this user, is this specific action permitted?” While governance can constrain what an agent is permitted to do, it cannot transfer responsibility for the result.

HUMANS REMAIN RESPONSIBLE FOR THE OUTCOME

Orchestration can automate execution, but not accountability. Accountability begins with measuring accuracy of the workflow rather than the volume of AI output.

Karp recommended measuring “time from insight to action, not just insights generated,” along with fewer handoffs, shorter cycle times, earlier risk detection, and tangible business outcomes. Ries added another useful filter: frequency. A workflow performed constantly may justify significant automation; one performed twice a quarter may not.

Accuracy and timeliness also need to be considered together. A perfectly accurate result delivered after the decision has been made has little value. A fast answer that is confidently wrong may have negative value. Workflow designers must set acceptable thresholds for accuracy and latency, then build the appropriate data, validation, escalation, and monitoring around them.

That makes the human role larger than approving whatever an agent produces. Humans establish intent. They map

consequences. They decide what agents may do, which information they may use, and when uncertainty requires escalation. They monitor drift, investigate anomalies, maintain institutional context, and redesign workflows when experience reveals that the original assumptions were wrong.

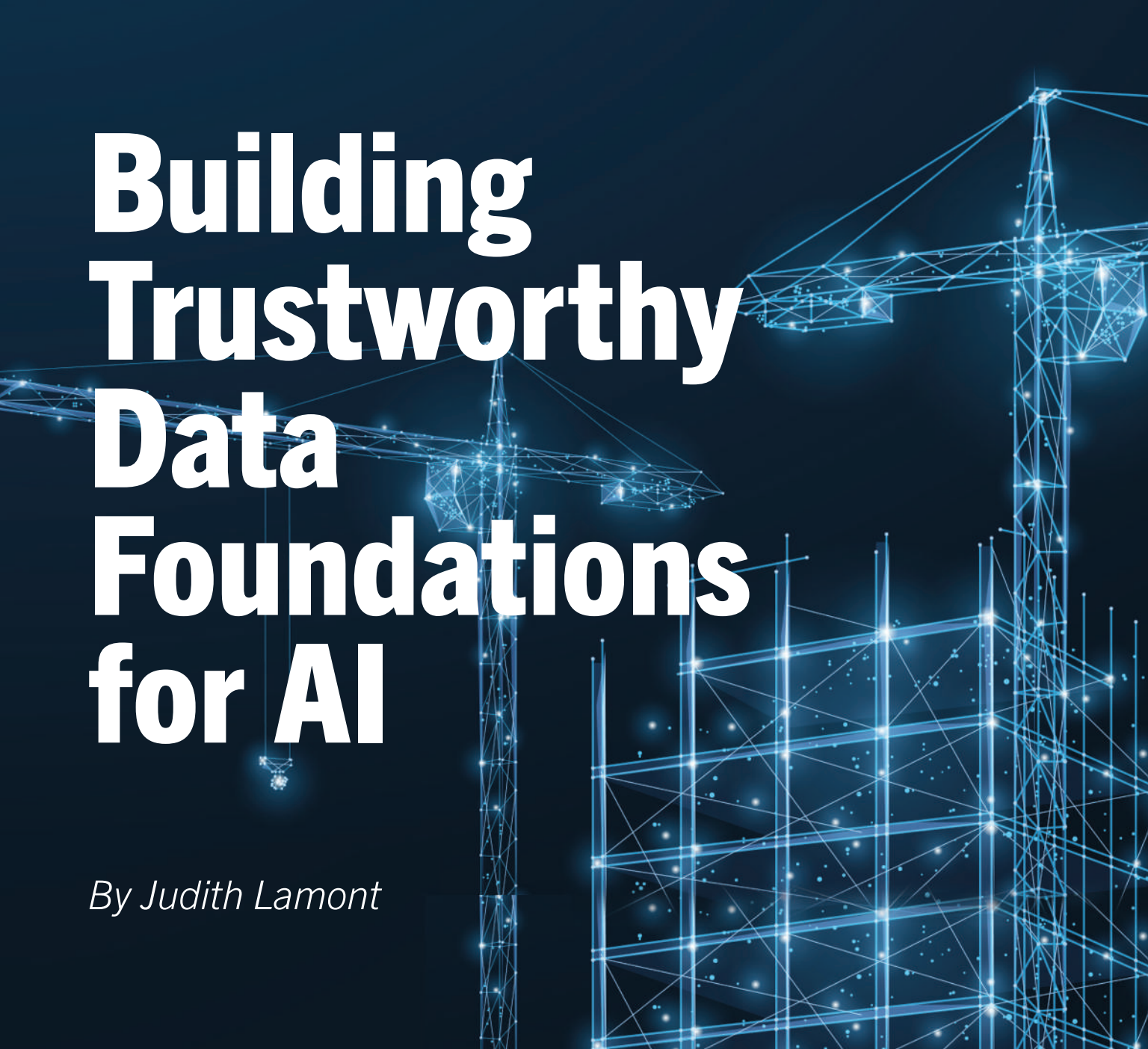
Microsoft’s research found that AI users increasingly identify quality control and critical thinking as crucial human skills, with 86% treating AI output as a starting point rather than a final answer. At organizational scale, the report argues for an evaluation infrastructure that explicitly assigns responsibility for reviewing agent performance and updating the workflows agents execute.

Roman Oliinychenko, CEO of New Wave Devs, described the emerging skill succinctly: “The skill is no longer writing the code; it is orchestrating the tool and knowing the moment it is confidently wrong. That judgment is what still has to be human.”

As agents perform more workflow steps, human work will shift from execution toward system design and accountability. The future of orchestration is not humans outside the loop watching machines work. It is humans designing the loops, choosing where judgment must enter, and remaining accountable for whether the work is accurate, timely, and fit for purpose.



DANIEL W. RASMUS is founder and principal analyst at Serious Insights LLC.



Building Trustworthy Data Foundations for AI

By Judith Lamont

A plethora of reports highlight the prevalence and dangers of AI hallucinations and other forms of inaccuracies resulting from AI applications. These errors can be traced to many different root causes, but the solutions boil down to a few methods: restrict the content to a carefully defined and structured set of validated information, protect the application through governance and monitoring, and evaluate the output to provide continuous improvement. ►



AN AI PARTNER FOR MEDICAL WORKERS

After years working in a variety of healthcare disciplines observing the challenges faced by researchers and practitioners as they searched and digested large quantities of information, Ome Ogbru sought to remove some of the laborious aspects of their work by using AI to assist them in collecting, analyzing, and presenting medical data.

A medical doctor, he founded a company called AI Narrative Generation and Engagement Solutions (AINGENS) and developed a software product called Medical Affairs Content Generator (MACg) that uses PubMed (pubmed.gov) and user-uploaded documents as its preferred sources for developing scientific content. PubMed consists of over 40 million citations for biomedical literature from MEDLINE, life science journals, and online books. The collection is held in the National Library of Medicine and is considered a highly reliable source of data. In existence since 1996, it is extensively used by medical information professionals, including physicians, nurses, and librarians.

MACg is designed to provide an AI-based workspace for medical writers, healthcare professionals, and researchers. “We cover the entire workflow,” said Ogbru, “from literature searches to summarizing, editing, and producing slide presentations.” Having all these functions in one system gives a more seamless user experience, improves efficiency, and reduces cost for individual users and organizations. “The system cites its sources,” he emphasized, “providing traceability and verifiability, which increases trust. The evidence-first approach ensures that the system is using curated sources, reducing the risk of hallucinations.”

“There are a few steps to take in order to maintain accuracy,” Ogbru continued. “One is to understand the capabilities of the AI models, which have context limits. Users should not overwhelm AI models with many large files or very long instructions. Using the AI model’s training knowledge as a source of truth also increases the risk of inaccuracy. MACg reminds users to provide a knowledge source, says when it does not

have the right information to fulfill a request, and acknowledges when it does not know instead of fabricating information.”

BUILDING A SOLID AND PRECISE DATA FOUNDATION

Not every situation affords the opportunity to work with a focused set of information, which is particularly the case for large enterprises whose performance depends on numerous processes and volumes of disparate data. “Many organizations have multiple data lakes, warehouses, and application data,” said Gaurav Pathak, SVP and GM for data governance and privacy and CLAIRE AI for Informatica, a Salesforce company. “Our message is that a strong data foundation is essential to any AI application.”

Informatica has been managing enterprise data for over 3 decades and offers expertise in data integration, quality, governance, cataloging, and master data management through Informatica’s cloud. “Our core differentiator is our unified metadata foundation that allows us to provide productivity and efficiency benefits across data management tasks with CLAIRE.” CLAIRE is the metadata-driven intelligence layer powering the Intelligent Data Management Cloud.

This detailed metadata can be used to drive AI agents toward trusted data. For example, the agent’s goal may be to discover data that is managed by a certain owner, has been used repeatedly in reports, or has been published within the past month. “We can establish certified datasets that have been validated,” commented Pathak, “so that out of a huge body of information, the most trusted and relevant datasets are chosen. By using an index of high-quality data, we are saving time, reducing token costs, and increasing accuracy,” he pointed out.

AI provides the benefit of scaling otherwise manual processes, but that scaling also produces risks. “Even the smallest mistakes will be magnified when at scale,” Pathak noted. “Each of these mistakes compounds. In addition, with agentic AI, there is usually not just a single workflow,



A key element for reliability is to provide governance constructs and ownership clarity.



but multiple ones chained together. Even if each one is 95% accurate, having multiple workflows can quickly reduce the overall accuracy.”

Other than having a solid data foundation, the most important action an organization can take in AI development is to evaluate system performance. “Right now, evaluation building for AI is in the early stages, more art than science,” explained Pathak. “Getting performance metrics such as how many customer calls are deflected versus being resolved automatically is key to continuous improvement. Without exploring the root causes of any issues, you don’t know if the answers are incorrect or the system is just lacking the information to provide an answer.”

A SYSTEMIC APPROACH TO ATTAIN TRUST

Organizations should keep in mind that AI applications are systems, and all the components must work in concert for AI to be successful. “Every piece of the system influences reliability,” said Ben Prescott, head of AI solutions at Trace3, an IT consulting firm. “The biggest misunderstanding organizations have about creating AI applications is that a new model will solve their problems. AI is reliant on data. If there are version control issues, data quality issues, or structural issues in the data, AI will not function properly, regardless of the model.”

Design elements can help mitigate AI fabrications. “Training data existed at a point in time and may have been accurate at that time, but once in production, flaws may become evident—a link from the training data may be dead, for example,” explained Prescott. “In that case, it is not technically wrong, but it is no longer producing reliable results.” By anchoring the

AI to updated information from the organization or by implementing a validation loop to open and verify sources before presenting them to the user, these effects can be reduced.

A key element for reliability is to provide governance constructs and ownership clarity. “Who is responsible for reviewing and updating the data? This needs to be built into the AI lifecycle and measures taken to transfer ownership if the owner leaves that job,” advised Prescott. “In addition, there should be a way to identify data that is ready for AI, and then marry it up with business use cases, rather than trying to get all the data ready for any use case that might come up.”

The systems approach should extend to the development process itself. “It is better to not treat the AI development team as separate from other teams, but to have developers embedded within the business units and working closely with them,” commented Prescott. “Otherwise, the AI team can become its own silo, and that is a recipe for stalled or failed projects.” Consistent interactions with the users and subject matter experts will help all the stakeholders develop the knowledge and skill sets they need for success.

Metrics for success for AI applications are not as straightforward as they are for many other enterprise applications. “Defining what ‘good’ looks like is a challenge,” Prescott acknowledged. “Success for a business user is not the same as success for an engineer.” How accurate does an AI answer have to be for it to be accurate enough? As good as or better than that of a human? “Every step has things we can tweak to improve reliability, and humans should always be in the loop to provide feedback,” he concluded. “It is not a static scenario, but an ongoing process.”



PROACTIVELY MANAGING COMPLIANCE

Another approach to ensuring valid output from AI systems is to deploy a platform specially designed to manage data, risk, compliance, and privacy. Daimler Truck North America (DTNA) wanted to use AI agents to provide early warning of supply chain problems, given the severe impact that such issues can create. “Daimler has a million trucks on the road,” said Satyen Sangani, CEO and co-founder of Alation, “and if a truck breaks down, they guarantee delivery of parts within 24 hours.” The company developed AI agents in two categories: one group to serve as planners and another to carry out tasks.

The agents monitor inventory, detect risks, and alert human planners about impending supply chain problems before small ones become big ones. DTNA uses Alation to manage and govern the underlying data that informs the AI agents about the status of company resources and activities with respect to its manufacturing processes.

Prior to launching this initiative, DTNA had also been using Alation to manage General Data Protection Regulation and other compliance requirements, finance-related matters, and assess data quality throughout the enterprise. As a result of its reliance on Alation, DTNA is able to make evidence-based decisions across numerous enterprise functions, and Daimler has been forthcoming with its expressions of satisfaction about the role Alation is playing.

Alation was founded to connect people with the high-quality data they need in order to do their jobs. The company first introduced a data catalog that combined machine learning with human curators. In the following years, Alation developed AI agents to manage and govern data and AI applications and, most recently, introduced the Alation Intelligence Operating System.

“Ultimately, our platform is designed to help organizations get reliable answers from AI,” explained Sangani. “The best way to launch an AI project is to start with a specific business problem, such as a compliance issue that has to be implemented. In such cases, it is usually clear what data is needed, and the outcomes can be tested.” One of the most difficult obstacles is that although customers may like the idea of AI, they don’t fully trust it and do not want to change the way they work. “People have legitimate reasons to be cautious,” agreed Sangani, “but when they work with a reliable system that proves its value over time, there can be real benefits to efficiency and ROI.”

GOVERNANCE AND ACCURACY FOR KNOWLEDGE MANAGEMENT

Knowledge management (KM) has been one of the major beneficiaries of AI. “Many have woken up to the fact that knowledge management as a discipline is fundamental and a good use of AI,” said Stephen Bedford, product director at ClearPeople. “However, provenance is increasingly important, and the output needs to be correctly sourced.”

ClearPeople’s AtlasFuse is a KM platform designed for Microsoft 365 users. It was developed based on the company’s professional services experience in KM to address the lack of an existing KM platform for Microsoft 365. “All of the security of Microsoft 365 is present, but we add several layers of governance,” continued Bedford. “The user can configure AI settings such as the model’s ‘temperature,’ which controls how deterministic or creative the responses are, as well as how restrictive the search should be.”

Typically, the user would limit the search to content in a particular collection, or only retrieve content within a certain timeframe or just the final approved version of a document. Users can also identify the owners of the information to allow for direct verification. “Copilot alone can come up with an authoritative-sounding answer that is wrong,” he noted. “AtlasFuse can restrict the AI search, instructing it not to provide an answer if the specified content does not contain it.”

At the Royal College of Psychiatrists in England, one of the tests students must take requires an interview with an individual taking the role of the patient to ensure that the students are interacting appropriately. In order to improve the performance of those who did not pass, feedback was provided by two evaluators who analyzed the conversation and actions of the student. AtlasFuse was trained to do these reviews, with issues of concern defined by subject matter experts, and now the time required for the feedback process has been reduced by 80%.

The Royal College of Psychiatrists methodically validated the system before deploying it by comparing its assessments against expert-reviewed, trusted reference answers across a representative set of examinations, refining the prompts and evaluation criteria until the AI consistently aligned with the expected outcomes. In addition, the college can now provide useful feedback to all test takers, not just those who failed, which was not previously possible from a time and resources perspective.

Building reliable, governed, and high-quality data foundations that reduce hallucinations and improve trust in AI-driven actions and outputs is an achievable goal. However, it requires a combination of planning, vision, and technical expertise. We are still on the learning curve, but the potential is significant and worth the effort.



JUDITH LAMONT is a freelance writer specializing in knowledge management and other technology topics, email jlamont@sprintmail.com.

Governing AI for Security, Compliance, and Trust

Who Gets to Grant AI Access?

By Colin Levy

Organizations are moving AI from isolated tasks into everyday operations. A tool that started out drafting emails or summarizing documents may soon be asked to search internal files, retrieve customer information, prepare transactions, route requests, or update records. ▶



Those uses require more than an assessment of whether the tool produces accurate answers. They require decisions about access, authority, review, and responsibility. An organization needs to know what information the system can use, which systems it can connect to, which actions it can take, and when an employee must review its work before anything changes.

These questions become more urgent as AI is increasingly being connected to business processes. A system that summarizes a contract for a lawyer presents one set of risks. A system that retrieves contracts from a document repository, compares terms against internal policies, sends approval requests, and updates a contract-management record presents another. The system has access to company information and the ability to affect a business process.

The same pattern appears in finance. An AI system may help an accounts-payable team identify duplicate invoices, find missing documentation, or organize a payment queue. Those are useful forms of assistance. Allowing the system to release payments, modify vendor banking instructions, or approve an unusual transaction requires additional controls. The potential loss is higher, and an error may be difficult to reverse.

PLANNING ACCESS FOR HUMANS, MACHINES, OR BOTH

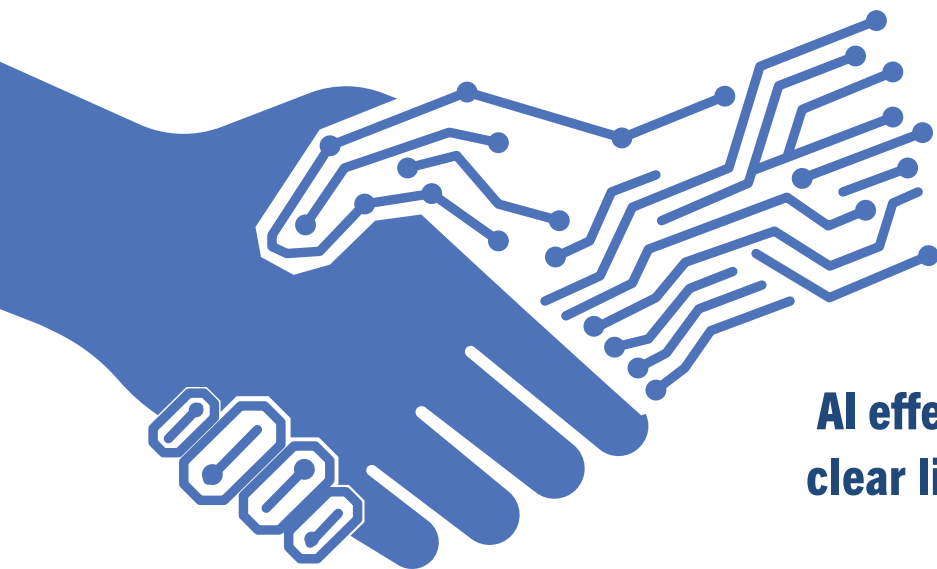
Organizations should decide in advance which activities AI may support, which require human approval before completion, and which remain outside the system's authority. Those decisions should be reflected in system settings, employee guidance, and the design of the underlying workflow. A policy that says people should "use their judgment" is not enough if the technology permits an AI tool to complete an action without review.

Access should be limited to the information and systems needed for a specific task. A customer-service tool may need access to a current order status. It may not need unrestricted access to payment details, personnel records, legal files, or every customer interaction in the company's history. A human employee would not ordinarily receive all those permissions for a narrow assignment, and AI systems should be subject to the same discipline.

This is particularly important because AI can process and distribute information quickly. It can search large sets of documents, combine material from different sources, and repeat a flawed instruction across many cases almost instantaneously. Errors do not need to be dramatic to create meaningful problems. An inaccurate summary, an improperly disclosed record, or an incorrect automatic update can affect customers, employees, and business partners before anyone notices.

Organizations also need a clear record of how AI is being used. They should be able to identify the tools in use, the data each tool can access, the business purpose it serves, the employees who oversee it, and the steps available to pause or disable it. This record should be reviewed constantly as the tool's role changes. A system approved to organize meeting notes may later be used to prepare client communications or assess employee performance. That expansion should not occur without a new review.

Legal and compliance teams should be involved before a system becomes embedded in a process. Their role is not limited to identifying rules that may apply; they can help



Organizations can use AI effectively while maintaining clear limits on what it can see, recommend, and do.

determine whether the proposed use relies on sensitive information, creates a record-retention issue, affects a regulated decision, or changes an existing approval process. Security teams can evaluate access and data handling. Business leaders can assess operational value. The organization needs all those perspectives before granting broader authority to a system.

Vendor arrangements require the same level of attention. A provider may host the tool, process data, or supply the underlying technology. The organization using the tool still needs to understand where its information goes, how long it is retained, whether it may be used to improve the provider's services, and what happens if the provider experiences a security incident. It should also know whether the vendor can explain the system's actions, preserve relevant records, and cooperate with an investigation.

Responsibility inside the organization should be assigned to a named owner. That person does not need to manage every technical detail. But they should be accountable for the system's intended purpose, approved access, required oversight, and performance in practice. When a system produces an inaccurate or harmful result, employees should know who can investigate the issue, correct the process, and determine whether the tool should continue to be used.

Employee training matters as well. People need to understand that AI output may be incomplete, inaccurate, or based on information that does not fit the situation at hand. They should know when to verify an answer, when to avoid entering sensitive information, and how to report a result that appears wrong. They should also understand that responsibility for a decision remains with the organization and the people authorized to make it.

APPLYING CONSISTENCY TO GOVERNING AI

A workable AI program does not require an organization to review every low-risk use in the same way. It does require a consistent method for distinguishing low-risk assistance from systems that affect sensitive information, money, rights, or access to important services. As a tool gains access or begins to influence more consequential decisions, the level of review should increase. It's important to remain one step ahead of the issues AI could create and quickly adapt when problems arise.

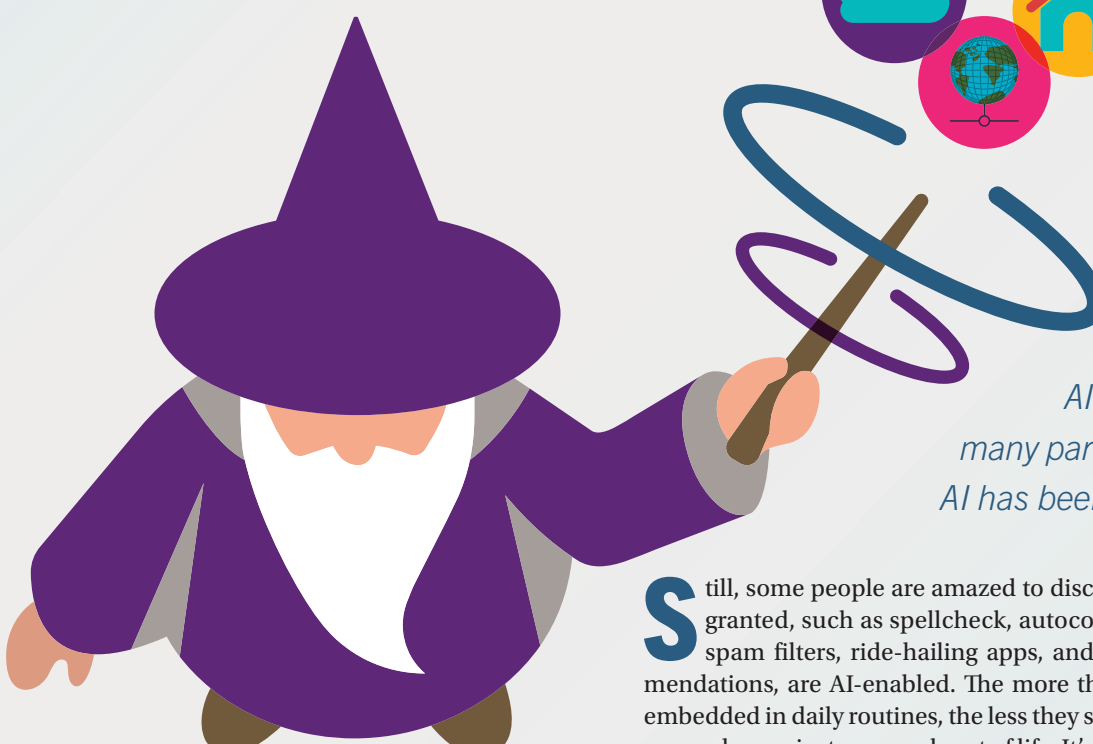
Organizations can use AI effectively while maintaining clear limits on what it can see, recommend, and do. That requires specific permissions, documented controls, active oversight, and accountable leadership. Without those measures, AI use is likely to expand through informal decisions made one connection and one convenience at a time.



[COLIN LEVY](#) is a well-known speaker, writer, and legal tech evangelist. Levy has more than a decade of experience in corporate transactional law and is often found educating lawyers on how to leverage technology to improve their delivery of legal services. His recently released book, *The Legal Tech Ecosystem: Innovation,*

Advancement & the Future of Law Practice, demonstrates how to bridge the gap between the world of high tech and the law.

A collection of approximately 20 circular icons in shades of purple, yellow, pink, and teal. The icons represent various digital and organizational concepts: folders, documents, bar charts, speech bubbles, film strips, padlocks, server racks, shields, houses, globes, clouds, magnifying glasses, and network diagrams. Some icons include small symbols like dollar signs, plus signs, or minus signs. The background is white with faint blue geometric shapes at the top left and bottom left corners.



AI
many par
AI has been

Still, some people are amazed to discover that many of the AI-enabled features we take for granted, such as spellcheck, autocorrect, and spam filters, ride-hailing apps, and music recommendations, are AI-enabled. The more these tools are embedded in daily routines, the less they seem like magic and more just a normal part of life. It's

Still, some people are amazed to discover that things they take for granted, such as spellcheck, autocorrect, real-time traffic maps, spam filters, ride-hailing apps, and personalized music recommendations, are AI-enabled. The more that AI-enabled tools become embedded in daily routines, the less they seem to be advanced technology and more just a normal part of life. It's no longer magic, it's normal.

We're not entirely at the same stage of magic becoming normality when it comes to agentic AI. Many aspects and implementations, however, are becoming standard operating procedures within enterprises. A key driver in powering AI agents is the connection of internal knowledge, wherever it might be stored, with retrieval-augmented generation (RAG). What agentic AI brings to the enterprise, what sets it apart from the more standard and somewhat simplistic uses of AI, such as spellcheck, is the ability to reason.

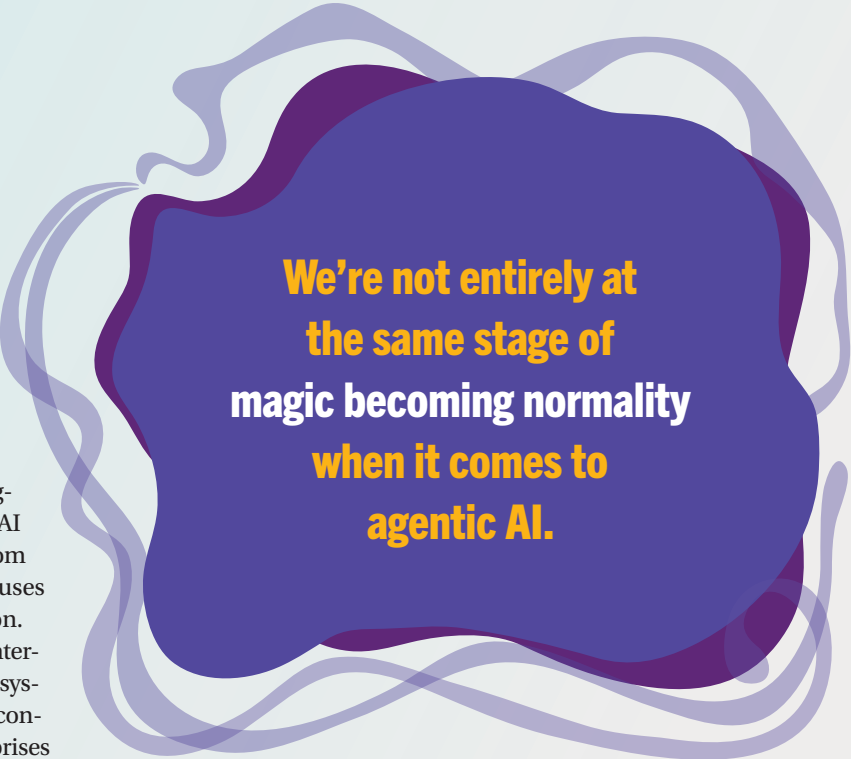
As agentic AI takes center stage in our enterprise AI world, RAG and enterprise knowledge systems help AI agents to deliver more accurate, contextual, and business-aware responses. Enterprises embracing agentic AI is a logical next step beyond the static chatbots and limited large language models (LLMs) that became commonplace for generative AI (GenAI) only a few years ago. They had no reasoning ability and too easily forgot what they had delivered earlier, even to the same inquiry from the same person. In addition to a paucity of memory, they were one-trick ponies, unlike the dynamic, multistep, reasoning workflows of today's agents. The agents employed within enterprises can make decisions and act autonomously.

The early LLMs, available at no cost to pretty much everybody, were, by today's standards, surprisingly limited, although they seemed extremely impressive at the time. Even the more advanced versions currently in existence are generic, not tailored to a particular enterprise or trained on language for a specific discipline. Basically, they are commodities, and their lack of proprietary context makes them less useful within organizations, even though they are perfectly adequate for less challenging, more general tasks. Human oversight should exist, but it is often minimal.

THE VALUE OF RAG

A huge drawback to the acceptance of GenAI into the enterprise was the LLMs' annoying tendency to hallucinate. As enterprises sought to ensure that their models returned accurate, truthful information, not the plausible but false hallucinated information, they quickly turned to RAG.

RAG provides the very useful antidote to false information by consulting external data sources to validate what it's found within an LLM. External, that is, to the LLM. Depending on how RAG is implemented, the validity checking might be to sources totally external to an organization. Within enterprises, it's much more likely to be external to the model, but internal to the organization. For a query about time-off



**We're not entirely at
the same stage of
magic becoming normality
when it comes to
agentic AI.**

policies, for example, an external source could be applicable labor laws, but policy manuals, documents, and databases held within the human resources department would probably be consulted first.

Context is another area where RAG has value. A common methodology for handling vast amounts of data is to chunk it, breaking it up into vectors to make finding nuggets within it easier for the model. This can have the unfortunate effect of combining chunks that do not actually relate to each other, thus creating answers that are inaccurate. Outside of the tech world, this can also happen when a statement is taken out of context. An inaccurate statement is merged with an accurate one, resulting, overall, in misinformation. This can be as relatively innocuous as a statement that John Doe graduated from ABC University when, in fact, he lived for a time in the town where the university is located but did not attend that university. Perhaps he audited a class. Other statements taken out of context can have more serious repercussions; just ask any public figure.

On the positive side, RAG can help identify chunking errors. Scoring and reranking help with this. RAG also pulls from recent data so that newer, more accurate, information overwrites older, now inaccurate, data. In addition to promoting better accuracy and more targeted business-aware responses, this can reduce costs from both financial and computational standpoints.

There's more to RAG than information retrieval. Agentic AI is a step well beyond GenAI, which is where RAG initially proved its value. Agents act; they adapt to changes in the environment in which they are to perform their tasks; they reason; and they scale. The agentic AI capabilities enabled by RAG include breaking down complex, multilayered, business-centric

queries into a series of steps. Agents then follow those steps sequentially. Having done that, they can circle back, going through several iterations until they've adequately satisfied the request. According to an email from enterprise search company elastic, "The shift from retrieval to reasoning requires a different approach to building AI systems."

AGENTIC AI ARCHITECTURE

The architecture behind agentic AI has a number of moving parts, given its autonomous, multi-agent nature. It reflects a complex system with many interdependent components. It's not a question, however, of simply taking a turnkey approach. It's not an easy "find and replace" exercise. It's not, "Oh, let's take a chunk of technology, throw it at an agentic AI product, and hope it works." To be successful, agents should understand what their tasks are, have the relevant training to act autonomously, and carry on with optimal performance.

Training agents requires an understanding of all the legacy databases, unstructured data, and cloud resources that are part of the organization's information infrastructure. Previously created enterprise knowledge can critically impact the agent training process. This can seem overwhelming. Most companies produce massive amounts of documents in numerous formats. They store manuals, policy papers, press releases, customer correspondence, Slack and Teams messages, emails, spreadsheets, video clips, audio files, and a host of other types of information. Where and how that storage occurs vary greatly from one enterprise to another. Commonly, however, it's not nearly as systematically organized as one might wish.

To ensure that agents work as intended, some foundational tools are necessary. Moving away from the generic LLMs is an obvious first step. Understanding how to train models for differing formats is also essential. What works for text files doesn't necessarily work for images, audio, or numeric data. Which model works best will depend heavily on what the use case is. In most instances, there will be multiple use cases, and one model will not suffice. Instead, enterprises often invest in several models. How many is the right number? That varies considerably from one company to another.

CONVERTING DATA TO AI-USABLE FORMATS

To enable agents to access legacy enterprise knowledge in all its multitudinous incarnations, incantations don't work. Instead, a mechanism to convert unstructured data to structured, usable-for AI-data is needed. Several possibilities exist for this activity; many of them are tried-and-true older

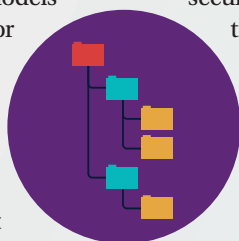
approaches that can be adapted to the newer realm of agentic AI. Vector embedding is one.

An IBM explainer (ibm.com/think/topics/vector-embedding) puts it this way: "Vector embeddings are numerical representations of data points that express different types of data, including nonmathematical data such as words or images, as an array of numbers that machine learning (ML) models can process." The similarity of the numbers reflects close matches back to the original material. The explainer adds, "Armed with such logical assumptions, vector embeddings can be used as inputs to models that perform useful real-world tasks through mathematical operations that compare, transform, combine, sort or otherwise manipulate those numerical representations."

Vector embeddings have long been popular, even before the ascendance of agentic AI, to make data semantically searchable, increase relevancy, and categorize input such as images or customer communications automatically. Vector embeddings are important to RAG by allowing LLMs access to the information stored in vector databases. They've acquired more potency as organizations struggle with implementing agentic AI as an operational part of their AI strategy, not as a pilot.

Enterprise service bus is another approach. This is "bus" in the computer science sense of the word, not the yellow school bus that goes by my front door at 7:30 a.m. Monday through Friday. According to AWS (aws.amazon.com/what-is/enterprise-service-bus), "The enterprise service bus (ESB) is a software architectural pattern that supports synchronous and asynchronous data exchange between separate applications. Large organizations have multiple applications that perform various functions using diverse data models, protocols, and security restrictions. An ESB makes application integration easier by performing operations such as data transformation, protocol conversion, message routing, and orchestration. Applications publish relevant data to the ESB, and it converts and forwards the data to other applications that need it."

AWS goes on to say that ESBs are not used as much as they were, as the preference has switched to API gateways. Microsoft explains that an API gateway "provides a centralized entry point for managing interactions between clients and application services" (learn.microsoft.com/en-us/azure/architecture/microservices/design/gateway). "An API gateway simplifies communication, enhances client interactions, and centralizes the management of common service-level responsibilities. It acts as an intermediary, and it prevents direct exposure of application services to clients."





There's no magic spell that can be invoked, no 'presto chango' that instantaneously causes all that legacy data to become easily findable.

RAG, LEGACY DATA, AND THE FUTURE

Whatever technology is used to make older information AI-ready—and it may well be more than one, based upon individual use cases—it can be an arduous process. RAG is powerful, but not omniscient, nor will it solve every content issue within enterprises. For one thing, RAG was designed with text in mind. Applying it to images and audio is problematic. Depending on the format of the data, going back to the source code could be necessary.

Strategic decisions about what legacy data is important for agents and what can be ignored need to be made. There's no magic spell that can be invoked, no "presto chango" that instantaneously causes all that legacy data to become easily findable. Moreover, it's possible that legacy systems have unacknowledged intrinsic value. It's best to run some tests to ensure that AI can access older data. And not just access it, but also deliver—accurately, quickly, and without additional steps—needed information that is routinely requested by users.

Going forward, the key is having standard approaches to newly created content so that no backward compatible issues will need to be addressed. Consistent and up-to-date metadata, agreed-upon schemas, Model Context Protocol (MCP) connectors, and curated catalogs contribute to agentic AI-ready data. Agentic RAG will play a major part, as queries become evermore nuanced and require a more advanced series of steps and decision points. If all human workflows were as simple as "if this, then that," we wouldn't need AI agents. But they're not. Decisions about

credit lines, supply chains, regulatory approvals, and exceptions to policies are far from being amenable to formulaic solutions.

There is real science behind transforming legacy content into information usable for AI agents and to forestall problems going forward. Varieties of RAG technology hold great promise for reducing, if not totally eliminating, hallucinated data and providing the magical single source of truth. Of course, it's not really magic, but as Arthur C. Clarke propounded as one of his three futuristic laws, "Any sufficiently advanced technology is indistinguishable from magic." The corollary is that, when those advanced technologies, in this case agentic AI, become just part of people's everyday lives, they no longer seem magical. Meanwhile, we'll look for how agentic and other AI technologies can bridge the gap between what is needed for real business outcomes and what is still in the realm of fantasy.



Marydee Ojala is an editor at *Enterprise AI World* and editor-in-chief at *KMWorld*. She has programming responsibilities for Information Today, Inc.'s Enterprise Search and Discovery and Data Summit conferences.

Enhancing AI Reasoning With **Knowledge Graphs and GraphRAG**



By Stephanie Simone

There is a disconnect happening with AI that cannot be ignored. A model can retrieve information, summarize it, or generate an answer, but it cannot determine whether the answer reflects how the business works or if it's truly correct.



The limitation is not model intelligence but missing context. Using connected data, semantic relationships, and graph-based retrieval improves AI reasoning, contextual understanding, and decision making.

According to Paige Leidig, CMO at TigerGraph, “AI without context behaves as an isolated predictor, operating without a mechanism to confirm whether its output aligns with enterprise reality. And this is where a knowledge graph becomes essential.”

DEFINING KNOWLEDGE GRAPHS

A knowledge graph (KG) stores a knowledgebase using graph-based data structures to represent and store the underlying information. The nodes of the graph represent entities, such as objects, places, people, etc. The edges represent the relationships (such as “is a customer,” “was born in,” etc.) between different entities.

KGs are also called semantic networks because they store information about how different entities are related to each other. The graph provides structure, logic, and relationships,





giving the AI system a grounding mechanism for deductive reasoning, rather than relying solely on inductive prediction.

A context-aware AI system can justify its answers, trace the logic behind them, and operate with greater reliability. A KG provides the connective layer required to make this possible.

According to the article, “Opportunities for Knowledge Graphs in the AI Landscape—An Application-Centric Perspective,” which appears in the special issue of *Web Semantics* titled *Science, Services and Agents on the World Wide Web*, “The potential of KGs extends across several domains, including knowledge discovery, intelligent automation, and cross-domain data integration. They enable AI systems to process vast amounts of data and generate deeper, context-rich insights. These capabilities make KGs a powerful asset in a wide range of AI-driven applications, from healthcare and finance to smart cities and autonomous systems.”

[MarketsandMarkets](#) reports that the growth of KGs is expected to expand exponentially, along with its potential. “As digital transformation initiatives accelerate and the demand for AI-driven applications increases, KGs are becoming an

essential component of modern data architectures, supporting scalability, interoperability, and continuous insight generation across industries.”

DataM Intelligence agrees, [reporting](#), “The investment case is especially strong because knowledge graphs now sit at the intersection of enterprise search, data governance, compliance, AI assistants, GraphRAG, semantic layers, fraud detection, customer intelligence, and data lake modernization.”

GRAPHRAG AS A CONNECTIVE TISSUE

GraphRAG is an extension of retrieval-augmented generation (RAG) that incorporates graph structure into the retrieval process. Traditional RAG retrieves documents based on vector similarity, whereas GraphRAG adds a layer of structural reasoning, so the system retrieves not only linguistically relevant information, but information that is contextually and relationally accurate. In a GraphRAG workflow, the KG becomes the source of truth for context assembly.

A recent report from [Gartner](#) has found that GraphRAG is poised to surge in popularity based on its dependability to



deliver contextual meaning and accurate results for complex use cases. Gartner predicts that “40% of enterprises will have leveraged GraphRAG techniques by 2029 to improve factual accuracy of responses and reasoning capabilities of LLMs [large language models].”

GraphRAG was introduced by Microsoft in 2024 to address the limitations of LLMs.

A [post](#) from the IBM website about GraphRAG says it includes the following components:

Query processor: The user’s query is preprocessed to identify key entities and relationships relevant to the graph structure. Techniques such as named-entity recognition (NER) and relational extraction from machine learning are used to map the query to nodes and edges within the graph.

Retriever: The retriever locates and extracts relevant content from external graph data sources based on the processed query. ... GraphRAG retrievers handle graph-structured data by leveraging both semantic and structural signals. They use techniques such as graph traversal algorithms—methods such as breadth-first search (BFS) or depth-first search (DFS) that explore the graph to locate relevant nodes and edges. Additional techniques include graph neural networks—advanced AI models that learn the structure of graphs to retrieve data effectively, adaptive retrieval—dynamically adjusts how much of the graph to search, reducing irrelevant information or noise—and embedding models.

Organizer: The retrieved graph data is refined to remove irrelevant or noisy information through techniques like graph pruning, reranking, and augmentation. The organizer helps ensure the retrieved graph is

clean, compact and ready for processing while preserving critical contextual information.

Generator: The cleaned graph data is then used to produce the final output. This can involve generating text-based answers using LLMs or creating new graph structures for scientific tasks, such as molecule design or KG expansion.

IBM noted that implementing GraphRAG systems is often a seamless process, as various tools and frameworks can be utilized, such as open source options, to support document processing, KG creation, semantic search, LLM integration, and more.

Popular tools include LangChain, LlamaIndex, Langflow, AstraDB from IBM watsonx.data, OpenAI, among others.

These solutions can work in tandem to enable semantic search by using vector embeddings, metadata handling for transparency, and context-aware response generation. Additionally, LLMs, including OpenAI GPT models, integrated through APIs can help produce precise and pertinent answers based on retrieved graph data, IBM concluded.

COMMON KG APPLICATIONS

As stated by the *Web Semantics* article, “KGs provide a fundamental framework for managing structured knowledge and are highlighted as a key component in the DNA of modern AI systems, offering significant potential for enhancing AI transparency, robustness, and the ability to integrate diverse data sources.”

There are many applications of KGs working to enhance AI capabilities, including these suggested in the *Web Semantics* article:

- ▶ Semantic reasoning and validation in healthcare
- ▶ Creating RAG based on KGs for education



- ▶ Using KGs to mitigate LLM hallucinations
- ▶ Using KGs for improving LLM-based question-answering
- ▶ Using KGs for addressing legal implications of LLM-based decision making
- ▶ Using KGs for AI system engineering, auditing, and data governance
- ▶ Combining KGs and AI for procedural knowledge management

Once established, a KG becomes a reusable source of truth that supports search, analytics, AI reasoning, and governance across many use cases. As the graph evolves with new data, it continuously improves the accuracy, explainability, and trustworthiness of the AI systems built on top of it.

Here are other practical use cases mentioned by Vishal Mysore, AI and data science expert:

Enterprise knowledge management: Organizations that need to connect relationships and insights, yet lack technical expertise to do so, can build dynamic knowledgebases that employees can query using natural language to discover essential and hidden information.

Research and academics: Researchers can use a knowledge graph to map complex relationships between concepts, papers, and authors to further explore connections through conversational interfaces.

Customer relationship management: Businesses can visualize customer journeys, product relationships, and service dependencies, thus making data-driven decisions more accessible.

Content recommendation: Media platforms can leverage relationship connections to deliver personalized recommendations based on complex user preference graphs.

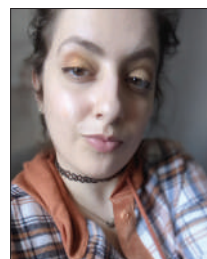
KGs also enrich AI agents, which fail when they act without enough enterprise context. In this instance, a KG gives agents a structured source of business meaning. It helps them ground outputs, retrieve relevant information, and trace answers back to governed entities. For high-stakes enterprise RAG applications, this structured context is often what separates a useful answer from an untrusted one.

THE FUTURE AND BEYOND

To operationalize KGs at scale, organizations should focus on building clear ontologies and semantic models, integrating data sources with graph-based pipelines, maintaining governance and access control, combining LLMs with KG-driven reasoning, and deploying graph-aware AI agents across workflows, as recommended in the *Web Semantics* article.

The future of KGs will likely focus on their integration with broader data ecosystems, improvements in scalability, and enhanced tools for real-time applications.

KGs excel at modeling relationships between entities, making them valuable for tasks such as search, recommendations, and data unification. As organizations handle increasingly complex and interconnected data, it will become pivotal for structuring context-aware systems.



STEPHANIE SIMONE is an editor at Enterprise AI World and [KMWorld](#). She is also the editor-in-chief of [Database Trends and Applications](#), a division of Information Today, Inc., and [Big Data Quarterly](#). She can be reached at ssimone@infotoday.com.

Enabling Real-Time Data for AI Systems

By Jelani Harper

The most immediate consequence of the widespread availability of modern statistical AI applications is perhaps their inordinate emphasis on enterprise data. Regardless of how capable they are, deployments of language models, intelligent agents, retrieval-augmented generation, and digital twins hinge on the data fueling them.

Moreover, these implementations introduce other data requisites which, if not necessarily novel, greatly transcend the scale at which they were previously necessary. Latency, even that involving a matter of seconds, has become all but intolerable. There is a renewed prioritization of data currency, for which traditional transactional or analytical engines simply aren't designed.

According to Aerospike CEO Srinu Srinivasan, the pervasive use of contemporary advanced machine learning (ML) systems has placed an onus on enterprise data infrastructure for "storing more data in real time, accessing more data in real time, and ensuring the data you access is accurate. This means you can't use a cache as much, because then you get a wrong answer, maybe an answer based on data that was 2 hours ago, and [it] isn't the right answer the user expects."

Numerous technologies have emerged as viable options for accounting for the real-time data interactions driving today's AI. Operational databases—including those specializing in time-series calculations—are gaining significant traction. The same can be said of streaming data tooling.

Regardless of which approach is invoked, the worth of real-time data access is predicated on considerations such as memory for AI resources, schema flexibility, high cardinality, data quality, and timely access controls. These concerns effectively ground lofty aspirations for statistical AI in the practical realities of the data driving them.

Or, as Sean Falconer, head of AI at Confluent, put it, "The moment AI has to do something useful, it needs data that changed seconds ago, sitting in operational systems the model has never seen."

THE DATA LAYER

The immediacy of the data required to capitalize on current investments in advanced ML may be its most prominent attribute. What time series databases, other types of operational databases, and streaming data stacks actually

“We’re the sensors: the eyes, the ears, the nose, the skin, the touch, the taste, and the smell. We’re also the arms and legs that take action. But, we’re not necessarily the brain, which is built elsewhere.”

*—Evan Kaplan,
InfluxData CEO*



furnish is the foundational data layer on which intelligence is applied. There’s a bifurcation to their utility for doing so, which Evan Kaplan, InfluxData CEO, described with a simple analogy. “We’re the sensors: the eyes, the ears, the nose, the skin, the touch, the taste, and the smell. We’re also the arms and legs that take action. But, we’re not necessarily the brain, which is built elsewhere.”

The sensors to which Kaplan referred are the respective data sources that real-time data infrastructure processes. They might include audio or textual inputs for model training data, network telemetry, or the latest signals from satellites in space. Moreover, once that data is processed and sent to applications of language models, it is often transmitted back for a defined business purpose.

What real-time data infrastructure actually does is purvey the “ingest, storage, and taking of action on the thing in the real world,” Kaplan summarized. It’s vital to realize the delineation between what the data layer and the intelligence layer achieve. Although several data platforms have mechanisms for analytics, “The LLM [large language model] has the intelligence to take data, which is real time, and then come up with the answer,” Srinivasan pointed out. “It has all the knowledge built into its models.” Consequently, there

may be less of a need for analytical databases within these low-latency constraints.

INGESTION AND DATA AVAILABILITY

Real-time data ingestion is pivotal for making data available for AI deployments at the current pace of business. Time-series databases that serve operational purposes specialize in high ingestion rates. Kaplan described an Internet of Things use case at scale in which there’s a sensor network of batteries—each generating approximately 40 metrics—in a million homes; organizations need to determine the state of the network in the past 10 seconds.

“Normal databases can’t ingest like this because you have to index a bunch of stuff, and it tends to take off,” Kaplan revealed. “A time-series database ingests this stuff and makes it easily available in almost zero time to be read, so it’s available to act.” Interestingly, these and other facets of real-time data processing run on CPUs, as opposed to GPUs, which are more specialized and commonly found for running the actual AI systems. There are numerous ways to expedite and scale ingestion operations, including employing what Kaplan characterized as a “Write Ahead Log, which makes it so that before the data is resonant in the database, you can act on it.”



SCHEMA AND DATA QUALITY

It's not uncommon for the data infrastructure layer's value to be based in part on adaptable schema mechanisms. Granted, options such as JSON, Avro, and Parquet have endowed organizations with schema flexibility prior to today's advanced ML. Nonetheless, real-time data infrastructure is replete with numerous constructs to reinforce the schema flexibility helpful for AI deployments. With schema-on-write capabilities, "You don't have to have everything mapped out perfectly beforehand and then manage the database that way," Kaplan explained. "You don't have to have this many columns and rows. You can say these are the things I want to collect, and I want to write it this way."

Data quality is imperative for accurate model responses and tenable actions from agents. Numerous options, particularly those from streaming data platforms, include mechanisms so that "duplicates, late data, and malformed records get validated and filtered in flight, before they reach the model," Falconer mentioned. Rectifying discrepancies in late data gives AI systems the proper sequence of time-sensitive data. High cardinality, which is said of columns in a database with many unique values, is another important notion that top time-sensitive data infrastructure natively supports. Kaplan described an optimal situation in which databases provide an "unlimited number of combinations of data, tags, and values that let you get a good picture of a system's current state."

LONG- AND SHORT-TERM MEMORY

The tenet of memory is ascending in importance across applications of language models, particularly when involving dynamic agents endowed with them. Real-time data infrastructure's rapid ingestion rates are complementary to a storage architecture that facilitates data access for low latency interactions with models. Top platforms that avail themselves of SSD storage and DRAM, in addition to forms of parallelism to optimize transactional interactions, can quickly make data

available "for short-term memory," Srinivasan remarked. "You can easily create new data items on-the-fly and also share it across multiple agents as you are creating these items."

The capacity to combine these techniques for short term storage—or memory for agents—with long-term storage within the same engine also assists with low-latency AI deployments. Support for ACID transactions ensures data isn't lost once it's stored. By pairing long-term and short-term storage in the same repository, real-time data infrastructure underpins numerous agent-based use cases. "Agents can audit the workflows of other agents," Srinivasan observed. "You can do audits after the fact or during the fact."

The enlarging number of agent applications is emblematic of the need for real-time data for AI systems, while heightening other concerns such as controls for them. "We're going through a shift from a world that's hyper-static, from a security and control perspective, to one that has to be incredibly dynamic, because we have this new actor type that is probability distribution," reflected Keycard CEO Ian Livingstone. "It has a different security profile, and what we want is dynamic access controls based on what the agent is assigned at that point in time."

STREAMING DATA

The capacity for language model-powered agents to perform actions in real time typifies the evolutionary leap forward of modern AI applications. Although other forms of data infrastructure, including operational databases, are applicable to these model-infused bots, streaming data platforms have several characteristics that are ideal for such applications. In fact, the pendulum may be shifting so that "agents are the primary adopters of streaming systems," Livingstone said.

The uninterrupted processing of streaming data platforms is utilitarian for low-latency agent deployments. Even popular open source tooling contains many time-sensitive calculations devised to transform data into a state consumable for downstream applications involving agents or foundation models. According to Falconer, "Tumbling and sliding windows give you rolling metrics such as the last hour's error rate or today's completed tasks. Session windows group a string of events, a user interaction, or a trip into one unit the agent can reason about."

REAL-TIME AGENTS

Streaming data systems are particularly helpful when there are multiple data sources channeling continuous data flows. Such applications might involve reading sensors from a factory or supporting facets of robotics. The queuing constructs of popular streaming data resources such as open source Apache Kafka are ideal for these workloads. "When you're having multiple streams, meaning multiple writes from





Contemporary solutions provide such functionality in addition to event-based controls that can catalyze actions with negligible, if any, latency.

different systems at different times, you can put it in one queue and pull it from that one queue,” Kaplan commented.

Such queue-based tooling typically adheres to a publisher subscriber paradigm, in which some sources write data to a queue while consuming applications, typically called sinks, receive it. Coupling this functionality with some of the temporal capabilities of streaming data platforms accelerates deployments of agents. According to Falconer, “Pattern detection recognizes a sequence of events that should inform or trigger the agent. And latest-state materialization collapses a changelog to the current value per key, so the agent reads the current status of an order and not every update it ever had.”

The complexity of the workloads supported by real-time agents is nothing short of admirable. They enable AI deployments at tremendous scale. Subsequently, “One person can now deploy tens or even hundreds of agents to do a bunch of tasks, and they’re autonomous,” Srinivasan revealed. “They go around generating more data and sharing more data and looking at things and making decisions, and all of this is machine-driven.”

AUTONOMY IN PRACTICE

More importantly, agents are directly responsible for a shift in the very nature of what a “task” is, from one that is predefined and static to one that naturally evolves based on mutable business circumstances, if not user requirements. Such sophistication hints at the very potential of real-time AI—when underpinned by low-latency data reflecting the latest state to an objective. “With an agent, a task is an emergent concept,” Livingstone explained. “You go to ChatGPT; you start typing in the box. If the task starts off with ‘Research what the best pair of jeans are,’ then the task evolves from ‘I found them,’ to a purchase decision. It’s not a concrete thing that’s preregistered. It’s sort of emergent properties of interaction patterns.”

The facility at which agents can take the next step in a workflow based on real-time developments strikes at the epicenter of the autonomy for which many laud them. It also

raises pertinent questions about governance, traceability, and the limits to such autonomy—as well as how to impose them. Conventional methods are predicated on auditing, which requires analyzing events after they take place to see if they adhered to established protocols. Contemporary solutions provide such functionality in addition to event-based controls that can catalyze actions with negligible, if any, latency.

They’re based in large part on “collecting telemetry of what the agent is doing in real time,” Livingston divulged. “So, as that agent becomes unaligned and starts doing things someone didn’t ask it to do, you can immediately detect that and kick off an autonomous response.” Responses and remediation measures can include revoking an agent’s access privileges, modifying governance policies, and instituting human approvals for agent actions.

DATA DICTATES

With streaming data platforms, time-series databases, other operational databases, and modern governance measures, organizations can avail themselves of a reality in which, in real time, their agents are “invoked by a change in the data itself,” Falconer said.

One can readily extrapolate those capabilities to include everything AI systems can do via real-time data. Enabling real-time data for AI systems is an ongoing effort showing real benefits to organizations.



JELANI HARPER (ddeux@comcast.net) has worked as an analyst, research lead, information technology editorial consultant, and journalist in the data management space for over 10 years.

Creating Business Context With Semantic Layers

By Joseph Hilger, Lulit Tesfaye, and Zach Wahl

As organizations lean into the “Age of Artificial Intelligence,” setting strong priorities with accompanying investments in AI, many are waking up to what we, in the knowledge management (KM) space, have always known: that no technical solution alone will deliver true business value without clearly and consistently defined context and meaning to bridge the enterprise. ►





Large language models (LLMs), although incredibly powerful, will inevitably fail to deliver highly accurate and consistent results without this specific meaning and context unique to every organization. This is in large part why, according to McKinsey in its “State of AI in 2025” survey, less than half of organizations report realizing enterprise-level financial impact from their AI programs. The gap is not in the technology, nor in organizational will; rather, it is in the semantic foundations that deliver consistent meaning and context.

Enter the semantic layer. A semantic layer is a standardized framework that organizes and abstracts organizational knowledge assets through semantic components to connect both business users and machines to an organization’s information. Rather than physically moving data, it acts as a logical connector that translates, defines, and contextualizes information to ensure it is understood consistently across disparate systems. By bridging the gap between how content, data, and other knowledge assets are stored and consumed, the semantic layer serves as the core differentiator between a failed AI initiative and one that sticks within the organization, delivering the value of enterprise AI with accuracy, consistency, and organizationally specific results.

COMPONENTS OF A SEMANTIC LAYER

Notably, none of the components that comprise a modern semantic layer are new. Taxonomies, ontologies, and metadata all date back thousands of years; business glossaries, knowledge graphs, and knowledge assets, although somewhat more recent, still far predate the latest crazes around AI and context. What’s new is the specific combination of each of these semantic components to form the semantic layer and, moreover, what it can actually do for an organization and its AI aspirations. We define each of these semantic components to explain their unique value and role in the semantic layer, then discuss what they deliver for an organization that successfully brings them together.

Taxonomies are structured, typically hierarchical or categorical, lists that use controlled vocabularies to classify and organize information. Within a semantic layer, they serve as a fundamental backbone that supplies the structure, consistency, and contextual framework necessary for users and machines to effectively access and interpret enterprise information.

An ontology is a standard model that organizes structured and unstructured information through entities, their properties, and their relationships. For the semantic layer, it serves as a translator that defines the meaning and connectivity of organizational knowledge assets, enabling machines and humans to understand and reason across disparate data sources and fundamentally define relationships between assets.

A business glossary expands on taxonomy, delivering not just vocabulary, but definitions that serve to standardize key language, ensuring all business users within an enterprise interpret and understand concepts consistently. Mature business glossaries go beyond simple definitions and standardize formulas and processes, delivering greater alignment and consistency. Within the semantic layer, the business glossary acts as

a foundational translation bridge by providing authoritative, standardized definitions and relationships that inform taxonomies, ontologies, and knowledge graphs, effectively linking technical data assets to the business vocabulary.

Collectively, taxonomies, ontologies, and business glossaries are the three core semantic components that deliver consistent meaning and context both for humans and machines.

The classic definition of metadata as “data about the data” still holds true. Metadata is the application of taxonomy terms and other semantic enrichments to describe each knowledge asset. This codifies an asset’s characteristics, such as its title, topic, or lifecycle status, to identify what it is and how it should be managed and used. Metadata also enhances its other descriptors that help AI understand each asset’s applicability and relationships. Within a semantic layer, it acts as a foundational component by ensuring AI-powered tools and users can effectively find, understand, and leverage information.

A knowledge graph is a machine-readable representation of an organization’s knowledge assets that leverage semantic components (ontologies in particular) to describe and relate data. These graphs allow organizations to model concepts and processes as entities, with connections between them illuminating their dependencies and relationships. Within the semantic layer, knowledge graphs function by using foundational building blocks such as taxonomies, ontologies, and metadata to provide structure, context, and meaning to underlying data. By mapping these assets, the knowledge graph transforms raw information into meaningful, interconnected data that powers enterprise search, AI applications, and complex reasoning.

Finally, “knowledge assets” encompass the breadth of an organization’s knowledge and information, including traditional concepts such as content, data, and documents, as well as people, processes, products, and physical equipment. Indeed, anything that can be tagged with metadata and connected to other knowledge assets is itself a knowledge asset. These diverse materials serve as the essential, unified fiber of an organization, capable of being linked, made traversable, and understood when integrated via a semantic layer framework. Within this framework, the primary role of knowledge assets is to be connected and abstracted to provide consistent organizational context and meaning for both machines and business users. By organizing these assets through a standardized semantic layer, organizations can ensure their information is not only findable and accessible but also effectively leveraged for intelligent applications and reliable, AI-driven reasoning.

While we mention them last, knowledge assets are where most organizations will see their AI goals rise or fall. After all these years, possessing accurate, enhanced, and complete knowledge assets remains the difference between an organization equipped to deliver the right information to the right people at the right time and the organization that is serving up old, obsolete, duplicate, and near-duplicate assets that will cause both humans and AI to “hallucinate.” Moreover, AI

is amazing, but it isn’t magic. It can’t fill organizational gaps in knowledge on its own (although it may try, to ill effect, without the proper guardrails).

Organizations that proactively invest in filling gaps in knowledge, leveraging traditional concepts in KM, along with the latest that AI note-taking tools and business intelligence have to offer, will be those that deliver the most complete solutions.

Putting it all together, the semantic layer functions as a unified ecosystem that brings together the individual components to transform fragmented content, data, and other knowledge assets into cohesive organizational knowledge. Taxonomies and business glossaries establish a foundational common language and structure, which ontologies then utilize to model complex relationships and entities across the enterprise. These words and their meaning are applied as metadata to enhance knowledge assets, giving them greater context and connectability. Knowledge graphs then leverage these building blocks to create a machine-readable representation of the entire landscape that can be employed for any number of use cases within an organization. Collectively, this semantic layer framework ensures that both business users and AI applications operate with a shared, authoritative context, delivering the accuracy and consistency required to realize true enterprise value.

BUSINESS VALUE FOR THE ENTERPRISE

At this point, we’ve answered the “what” and a bit of the “how” regarding semantic layers, but the most important question to answer is the “why.” Semantic layers are not easy to implement for the enterprise: They can be costly to implement and maintain, and the related concepts can certainly be esoteric to folks outside of the field. The most important part of this article, especially related to your ability to justify your organization’s needs to your executives, is what follows. What does your organization get from a semantic layer?

As we already stated, one of the major drivers for interest in semantic layers right now is what it can do for your organization’s AI initiatives. The implementation of a semantic layer is the primary driver for delivering accurate, reliable, and organizationally specific AI results. By acting as a standardized framework that abstracts and organizes knowledge assets, the semantic layer translates disparate information into a consistent, machine-readable language that both humans and AI can interpret. It “grounds” AI models by providing a governed knowledge layer, which significantly reduces the risk of hallucinations to ensure that agents are consulting the correct context and authoritative definitions rather than relying on obsolete or duplicate assets.

By leveraging ontologies and knowledge graphs, the semantic layer allows AI to understand complex relationships and reasoning across the enterprise, traversing different repositories and sources and building complete and actionable answers.



The value of semantic layers doesn't end with AI, however. In fact, it didn't begin there either. Semantic layers deliver a huge amount of value independent of the current AI boom. Any of these use cases can deliver massive value for an organization, regardless of whether the organization is prioritizing an AI transformation, while also serving as the first steps toward future AI initiatives:

- ▶ **360-Degree Views** – Call it a knowledge portal, a single pane of glass, or dashboard 360; at its core, a semantic layer can power the integration of your organization's knowledge assets, bringing together disparate types of sources into a single integrated view or set of answers. This is the fundamental dream of so many organizations: to simply deliver a complete view that is immediately actionable for their stakeholders, finally burning down the silos.
- ▶ **Findability and Discoverability** (moving from pull to push) – When you consider people as a type of knowledge asset, a semantic layer can be the digital matchmaker between a person and the information they need to do their job. Rather than going to look for it, it can be delivered to them at the point of need, reducing friction and making work more efficient and consistent.
- ▶ **Proactively Filling Knowledge Gaps** – Once an organization has a complete map of what they should know through taxonomies and ontologies, a semantic layer can help to spot where there are statistically significant gaps in an organization's explicit knowledge assets. Said differently, organizations have long lamented that they "don't know what they don't know." A semantic layer can address that issue and enable both people and AI to proactively fill in those gaps.

All of these potential cases, those with and without AI, lead to many of the same results for an organization. First come better, more complete, and more trusted results for their end users. This leads to greater consistency and quality of delivery, in turn improving efficiency.

When you factor employee learning and performance into this, organizations will also wind up with happier employees who upskill faster and more naturally, reducing training and recruiting costs while building up the capabilities of the organization and its people. There are other cost benefits to be had as well. A complete semantic layer will identify redundancies, both in systems and in knowledge assets, so organizations will, at a minimum, be reducing their server or cloud needs, while potentially reducing their license costs by eliminating redundant systems.

Semantic layers, put simply, deliver context for both humans and AI, resulting in more complete, reliable, and trustworthy delivery, consumption, and generation of information. The case for their present value is clearer than it's ever been. Organizations that want to get started should set the technology conversation aside for a moment and look at operating models, the readiness of their knowledge assets, and the foundational value of their semantic components. Organizations that have already made investments in those areas are the same ones that have realized benefits from AI. Semantic layers are the X factor that can deliver that value today.



JOE HILGER is the COO and co-founder of Enterprise Knowledge, a KM consultancy that has been listed on the *Inc.* 5000 List of Fastest Growing Companies in the U.S. for 5 straight years. He also co-authored a book on KM systems called *Making Knowledge Management Clickable: Knowledge Management*

Systems, Strategy, Design, and Implementation. Hilger has more than 30 years of experience designing, leading, and implementing cutting-edge, enterprise-scale KM solutions.



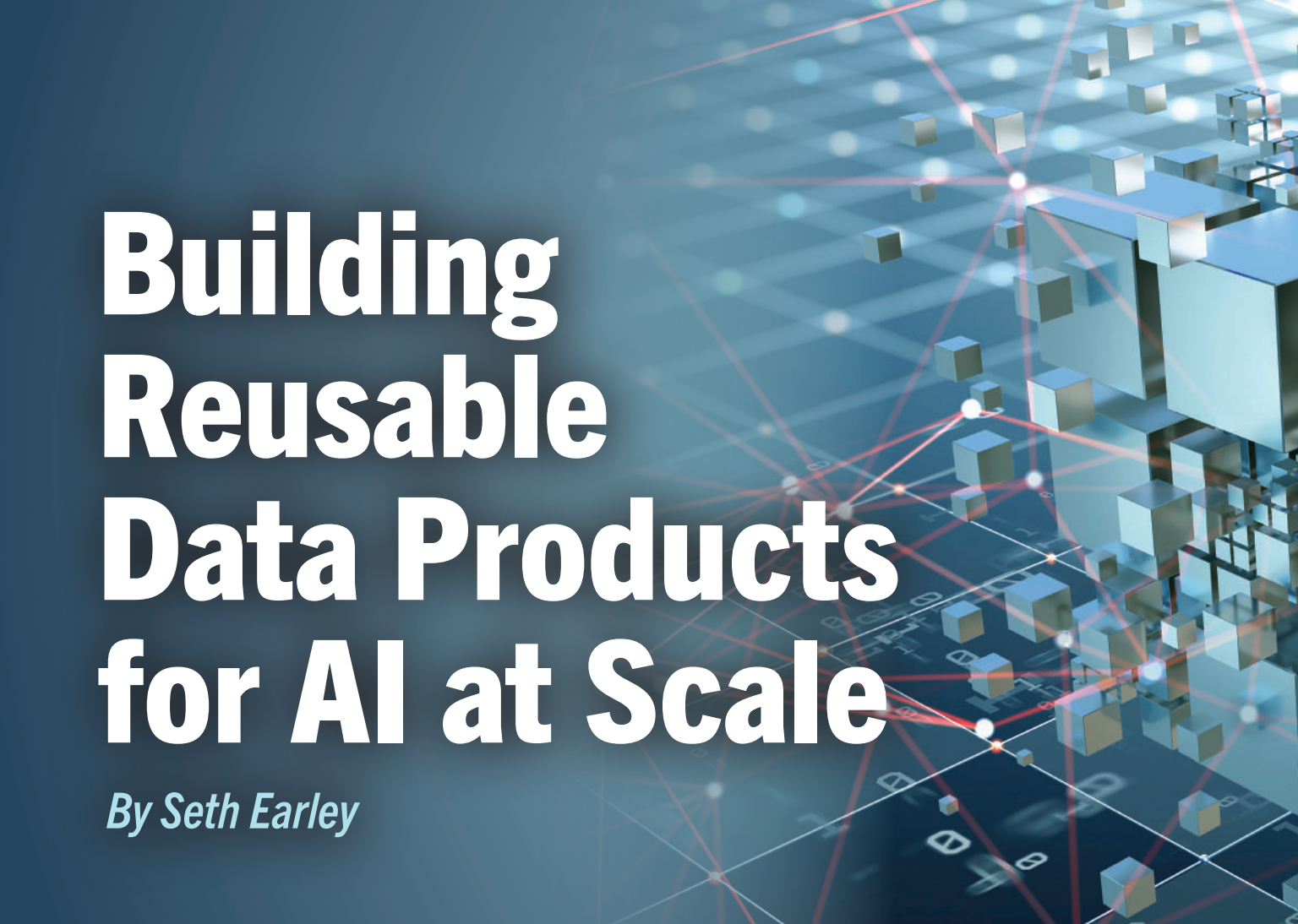
LULIT TESFAYE is a partner and the VP for knowledge and data services at Enterprise Knowledge. She has more than 17 years of experience leading diverse knowledge, information, and data management initiatives, specializing in technologies and integrations. Tesfaye is most recently focused on employing

advanced enterprise AI and semantic capabilities for optimizing enterprise data and information assets.



ZACH WAHL is the founder and CEO of Enterprise Knowledge, the world's largest dedicated KM consultancy. He possesses 25 years of experience leading KM initiatives, ranging from strategy and design to highly technical, enterprise-level transformations and implementations. He also hosts Knowledge Cast, the number-one rated KM podcast,

and is the co-author of *Making Knowledge Management Clickable: Knowledge Management Systems, Strategy, Design, and Implementation*.



Building Reusable Data Products for AI at Scale

By Seth Earley

An enterprise builds a customer service assistant. It takes several months, and most of that time goes into data work: extracting product information from a legacy system, reconciling conflicting customer records, and cleaning and structuring knowledge articles. The next project, a sales enablement tool, needs much of the same data, and the team rebuilds the pipelines. By the third initiative, an order management agent, the organization is funding the same data preparation a third time, with three different owners, three sets of business rules, and three definitions of “active customer.”

Duplicate pipelines are a cost problem that is solvable. Divergent definitions are a correctness problem that propagates. One pipeline treats a customer as active if they have logged in within the last 90 days. Another treats the same customer as active only if they have been invoiced within the last 12 months. Both counts are right, and neither can be fixed, because there is no bug, only a divergence of definitions.

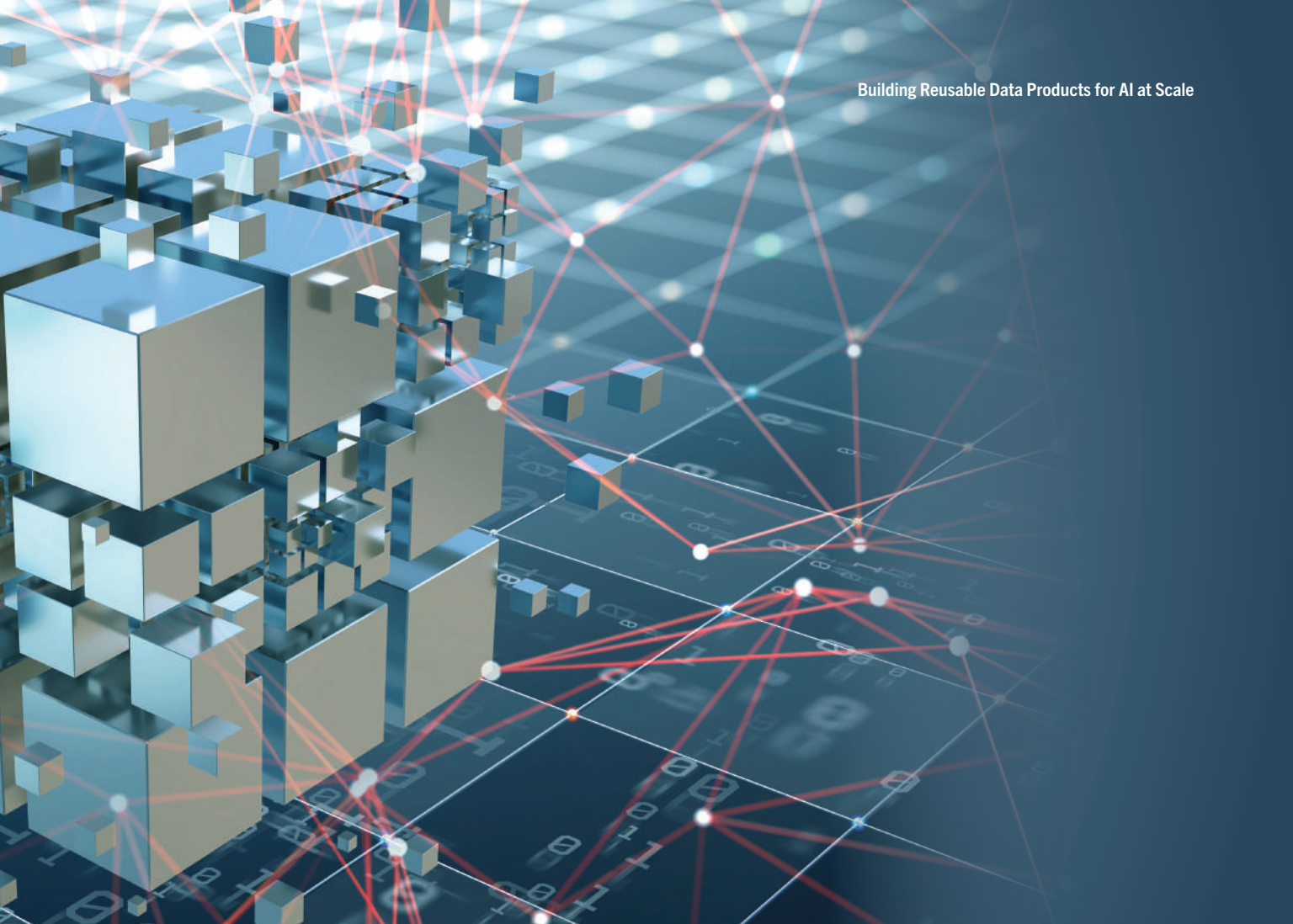
Gartner predicts that through 2026, organizations will abandon 60% of AI projects that are not supported by AI-ready data, and its survey of more than 1,200 data management leaders found that 63% either lack the data management practices AI requires or do not know whether they have

them. MIT’s Project NANDA reached a similar conclusion from another direction: 95% of enterprise generative AI (Gen-AI) pilots produce no measurable profit-and-loss impact, and the failures trace to integration and organizational readiness rather than to the models.

The pilot-to-production gap is a reusability and scalability gap. A pilot works on a bounded dataset that one team can clean by hand: a few thousand documents; one product line; and a single source system, prepared once for a single audience. Production removes every one of those conditions. It requires the same quality, structure, and meaning to be available continuously, across departments, for use cases that have not been fully scoped yet. Reuse depends on consumers agreeing on what the data means, and treating data as a product is how that agreement gets built, funded, and enforced.

FROM PROJECT THINKING TO PRODUCT THINKING

Product thinking makes enterprise data reusable, but only after the organization agrees on what the data means. Most enterprise data work is organized as projects. A project has a start date, an end date, a budget, and a defined scope of delivery. Teams build for the requirements in front of them; embed



business rules in transformation code, where they are hard to find later; leave the documentation in a departing contractor's head; and may not continue ownership at go-live. Repeat that across a portfolio of AI initiatives, and the result is integration debt, inconsistent answers across systems, and multiple un-governed copies of the same source data.

A product works differently in that it has a named owner, defined consumers, a road map, quality guarantees, and a lifecycle that extends beyond any single use case. Applying that discipline to data yields governed, reusable assets designed for consumption, with a contract that specifies structure and meaning, service levels that specify quality and freshness, and interfaces that let people and machines use it without reverse-engineering a schema.

Reuse is the economic engine. When five use cases draw from one governed product, the cost of quality, governance, and semantic reconciliation is paid once and amortized five ways. When five use cases each rebuild their own pipeline, the enterprise pays five times and gets five slightly different answers. McKinsey's research on managing data as a product found that new business use cases can be delivered up to 90% faster while total cost of ownership drops by about 30%. Those numbers match what we see in client engagements.

The idea has roots in the data mesh movement, but an organization does not need a mesh architecture to apply the discipline. Product thinking is an operating model choice, and it works with centralized platforms, federated domains, or anything in between.

WHAT MAKES A DATA PRODUCT AI-READY

Six properties separate a data product from a well-documented table. The first five are management discipline. The sixth determines whether AI can use the product at all.

1. **A named owner with a funded team.** Accountability for accuracy, refresh cycles, and consumer satisfaction persists after any individual project ends. Ownership without funding is a courtesy title.
2. **A published contract.** A schema describes structure. A contract adds quality thresholds, versioning, and a deprecation policy, and when it changes, consumers get a notice and a migration path. This is the difference between an API and a shared drive.
3. **Discoverability for humans and machines.** A catalog entry in business language serves human consumers. Agents need machine-readable descriptions of what the product contains, what questions it can answer,



and how to invoke it, which is what standards such as the Model Context Protocol are converging on. Data products are becoming the tool inventory agents select from.

4. **Trust signals that travel with the data.** Lineage, provenance, freshness, and quality metrics ship as part of the product rather than sitting in a governance tool nobody checks.
5. **Multiple consumption ports.** The same governed asset serves query access for analysts, APIs for applications, chunked content for retrieval, and event streams for agents acting in real time.
6. **Semantic grounding.** Field names and values resolve to enterprise definitions, and those definitions live in structures a machine can read. Does “customer” include former customers? Does “revenue” mean booked or recognized? When “lapsed,” “churned,” and “inactive” arrive from three source systems, which concept do they map to and who decides? Answering those questions is inherent to the process. It takes a reference model that fixes entity scope, controlled vocabularies with synonym control that collapse variant values onto single concepts, an attribute model specifying which properties the enterprise maintains for each entity type and at what granularity, and a relationship layer connecting customer to account to entitlement to product. Published once in machine-readable form, it is the layer that every data consumer inherits instead of reinventing. If this is missed, it will cause two AI assistants to give two different answers to the same question, and neither is wrong given what it retrieved.

The sixth is the most critical, and the first five depend on it. A contract without agreed-upon semantics specifies format, not meaning. A catalog is searchable only to the extent that its entries use vocabulary consumers share. Lineage tells you where a number came from, not whether it means the same thing as the number sitting beside it. Organizations defer this work because it requires business domain owners to collaborate with the data engineering function, and that collaboration rarely has a home in the org chart.

How much metadata to apply is the practical constraint. Too little and AI cannot find or interpret the data. Too much and nobody will maintain it. The workable target is the minimum set of attributes that supports the priority use cases, extended incrementally as new consumers come online.

WHY AI RAISES THE BAR

Traditional analytics kept ambiguity visible. An analyst who found two conflicting revenue numbers often did not know which one was right either, but they knew to flag the conflict, footnote the caveat, or go ask the source system own-

er. The output was a report, produced on a cycle, reviewed by people who understood how the numbers were built. GenAI removes that check. It answers directly to users with less visibility into the sources, at a volume that defies consistent review, and it gives no indication that the underlying data disagrees with itself. Ambiguity that used to surface as a question is now presented as a confident answer.

Retrieval-augmented generation requires quality data. Retrieval accuracy is a function of structure, metadata, and context, which is why the same model can look capable against a governed corpus and useless against an ungoverned file share. In the AI readiness assessments we conduct, knowledge readiness—the state of an organization’s data and content foundations—consistently scores lowest of the four domains we evaluate, behind operational, technical, and governance readiness. Many organizations have spent 2 years tuning prompts and swapping models while the constraint is the data.

Data products include product documentation, policies, procedures, and service histories. Each needs an owner, a curation process, componentized structure, consistent classification against the enterprise taxonomy, and a defined refresh cycle. Content chaos, meaning thousands of redundant, outdated, and trivial documents in ungoverned repositories, is the fastest way to make a model fail.

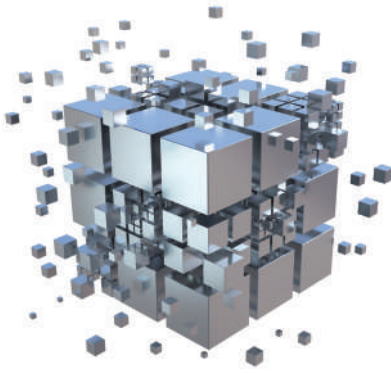
Agents raise the stakes when engineered into workflows. Pricing usually comes from a system of record, so that is not where this breaks. It breaks on the data surrounding the transaction: which contract governs this customer, what they are entitled to under it, whether a discount tier still applies, or whether a part is compatible with the installed configuration. That information is typically spread across CRM, contract management, the product catalog, and the pricing system, with no agreed rule for which source wins when they disagree. An assistant that gets it wrong gives a bad answer to someone who can question it. An agent that gets it wrong issues the credit or commits to the term. When AI systems perform an action rather than answer a question, the data they consume needs quality guarantees, entitlements, and provenance that travels with the record and supports an audit trail. Those are properties of a product, and prompt engineering will not replace them.

WHAT THIS LOOKS LIKE IN PRACTICE

An industrial distributor we worked with made product data its first data product. The distributor started with a rationalized taxonomy and an attribute model covering specifications, hierarchies, and compatibility relationships, governed by a dedicated team rather than mapping variations across channels by channel owners. That one product now feeds commerce search, the dealer portal, and a GenAI assistant for customer service. Conversion improved 40% in the enriched categories due to improved findability and discoverability, and onboarding a new product line dropped from weeks of manual work to days due to reuse and syndication.



Every enterprise running more than a few AI initiatives will end up treating its critical data as products, because paying to rebuild the same pipelines is not sustainable.



A financial services client took the same approach with customer interaction data, packaging records with entitlements and provenance so a service copilot could use them safely. Answer accuracy reached 96% once the underlying content and data were structured, tagged, and governed. The same model running against raw source systems produced results the compliance team would not release.

Both organizations started with use cases, identified the data those use cases shared, and settled the meaning of the shared data before building anything.

WHERE TO START

Progressive enhancement beats a 2-year program that productizes everything before delivering anything. Start where reuse value is clear and let each use case fund the next increment of the foundation. Here are five practices to keep front of mind:

1. **Anchor on a use-case portfolio.** Select two or three AI use cases with measurable business value and executive sponsorship, then map the data each one requires. The overlap is your candidate list, and there will be overlap. Customer, product, transaction, and knowledge domains appear in nearly every portfolio.
2. **Resolve semantics before building pipelines.** For the shared entities, get the business domain owners in a room and settle scope, allowed values, and relationships. Give the exercise a deadline, and require a decision at the end of it. A week of definitional argument now prevents a year of reconciliation later.
3. **Ship a minimum viable data product.** It should have a named owner, a versioned contract, a catalog entry, one consumption port, and a baseline quality measure. Resist modeling every attribute and supporting every access pattern on Day 1.
4. **Embed governance in the product.** Governance applied as a review board at the end of the pipeline blocks delivery. Governance embedded in the product, as validation rules in the pipeline, entitlements in the access layer, and stewardship staffed as a product management function, speeds it up. Policy is enforced by design rather than after the fact.
5. **Fund products, not projects, and measure reuse.** Project funding is why pipelines decay. The money stops at go-live. Give each data product a standing

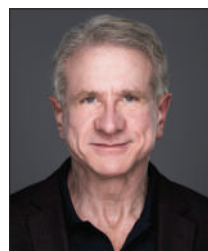
team and a road map driven by consumer demand. Then track consumers per product, time to stand up the next AI use case, and accuracy differences between AI systems running on productized versus raw data.

None of this succeeds inside the data organization alone. It requires collaboration across business and technical functions. Data products require agreed-upon definitions from business domain owners, entitlement rules from security and compliance, infrastructure from platform engineering, and consumption requirements from the AI teams. Start with cross-functional alignment on a small set of shared domains.

Every enterprise running more than a few AI initiatives will end up treating its critical data as products, because paying to rebuild the same pipelines is not sustainable. The open question is whether that happens by design, in advance, or after pilots stall. There is no AI without information architecture, and a data product is where that principle acquires an owner, a contract, and a budget. Pick the two use cases that matter most this quarter, identify the data they share, settle what it means, and put a product team behind it.

Note on Use of AI in This Article

This article was drafted with AI working from a curated corpus of my published work, client deliverables, and firm methodology, plus a voice guide built from that same material. The thinking, frameworks, and evidence are mine. I set the direction, iterate on drafts (many times!), and am accountable for the claims and conclusions in the piece.



SETH EARLEY is the founder and CEO of Earley Information Science and author of the award-winning book *The AI-Powered Enterprise* (LifeTree Media, 2020). For more than 3 decades, he has helped Fortune 1000 organizations structure the knowledge that makes enterprise AI accurate and trustworthy. He hosts the Earley AI Podcast and contributes regularly to *Harvard Business Review*.