O'REILLY®
Report

Structured for Intelligence

Why AI Needs Governed, Discoverable, and Provisioned Data

Tom Taulli, Tom Grabowski & Sai Maddali

Structured for Intelligence

Why AI Needs Governed, Discoverable, and Provisioned Data

Tom Taulli, Tom Grabowski, and Sai Maddali

O'REILLY®

Structured for Intelligence
by Tom Taulli, Tom Grabowski, and Sai Maddali

Published by O'Reilly Media, Inc., 141 Stony Circle, Suite 195, Santa Rosa, CA 95401.

O'Reilly books may be purchased for educational, business, or sales promotional use. Online editions are also available for most titles (*https://oreilly.com*). For more information, contact our corporate/institutional sales department: 800-998-9938 or *corporate@oreilly.com*.

Acquisitions Editor: Aaron Black
Development Editor: Gary O'Brien
Production Editor: Elizabeth Faerm
Copyeditor: Tonya Trybula
Proofreader: O'Reilly Media
Cover Designer: Susan Thompson
Interior Designer: David Futato
Interior Illustrator: Kate Dullea

December 2025: First Edition

Revision History for the First Edition
2025-12-03: First Release

This work is part of a collaboration between O'Reilly and dbt Labs. See our statement of editorial independence.

979-8-341-65297-2

[LSI]

Table of Contents

CHAPTER 1
AI Meets the Data Stack

In many organizations, the process for pulling insights from enterprise data remains a struggle, requiring users to wrestle with a patchwork of tools. These tools include BI dashboards, SQL editors, data warehouses, ETL pipelines, and governance systems, with each piece coming with its own interface, quirks, and learning curve. Most business folks lack the technical chops to use these tools on their own, so they lean on analysts or engineers to get the job done. For example, a marketing manager who needs a simple conversion trend might still wait several days for a data engineer to adjust a pipeline. Even with modern BI systems, it's common for insights to lag just enough that they're less actionable.

The numbers paint a stark picture of just how fragmented today's data landscape remains. Enterprise analytics teams are now working across an average of 400 data sources. At the upper end, nearly one in five enterprises juggles more than 1,000 data sources. And these figures come from organizations with at least 1,000 or more employees, making these numbers even more striking—these aren't small companies struggling with limited resources, but established enterprises with significant IT investments.

A 2024 industry survey adds operational texture to the picture. It revealed that more than 70% of data teams rely on five to seven different tools just to get through their daily workflows. About 10% are juggling more than ten. The result is mounting cognitive overload and constant integration headaches that bog down decision-making. For end users, the experience is often just as frustrating. They need

to navigate a patchwork of platforms, which makes it harder to find the data or insights they need, eroding the return on the data they collect.

The productivity toll is measurable across roles. The *2025 State of Analytics Engineering Report* states that although 70% of analytics and data professionals use AI to help write code and documentation, 57% still spend most of their time maintaining or organizing datasets, the same level as in the prior year. The promise of AI-assistant augmentation hasn't yet freed these professionals from the grunt work. Data scientists face similar challenges, spending about 60% of their time just cleaning and organizing data; another 19% is spent gathering the datasets. This leaves roughly 20% for actual analysis and insight generation, the part of the job that drives business decisions and growth.

Utilization lags accordingly. Another survey has found that between 60% and 73% of all enterprise data never gets used for analytics. And it's not just about volume; it's about access and alignment. Eighty-three percent of respondents said their organizations suffer from data silos, and nearly all of them (97%) believe those silos are hurting performance. Many users don't even know what data exists within their organization, let alone how to access or apply it.

This fragmented, siloed reality is the starting point. It's where most enterprises find themselves today—wrestling with complexity instead of extracting value.

But AI is about to change everything.

How AI Changes the Data Stack

AI is flipping the traditional model on its head. It's not just making data easier to access—it's changing how the entire data stack operates. Smart features can be woven into every layer, such as natural language interfaces that translate user requests into SQL queries, or AI assistants that handle data preparation and analysis. This makes working with data feel more like having a conversation than wrangling code. This isn't just a nice upgrade—it's a reinvention of how companies get value from their data.

From Dashboards to Dialogue

For much of the modern data era, business intelligence (BI) has been defined by static dashboards and the technical expertise required to navigate them. Pulling meaningful insights often meant waiting in line for scarce data analysts to run SQL queries or create tailored reports. This system empowered only a fraction of the workforce—those with the tools and training to interpret raw datasets. For the rest, decision-making often lagged behind, hampered by delays and a dependency on intermediaries.

But this bottleneck is now easing. AI-powered conversational analytics allows users to ask questions in plain English, such as, "Show me global sales in November," and then refine with tweaks such as, "Now just show me EMEA." But the results are based on a foundation of strong governance and compliance.

Beyond Conversational Analytics

Beyond one-off questions, agentic AI coordinates multistep work: planning, writing code or SQL, running checks, and proposing changes, and is poised for explosive growth. Research from Capgemini found that 50% of enterprises plan to implement AI agents in 2025, with adoption expected to reach 82% in 2028. Nvidia CEO Jensen Huang has said that this technology is "a multi-trillion-dollar opportunity."

True, such predictions should be taken with a grain of salt. This is especially the case with dynamic categories like agentic AI. But as seen with the growth in usage of tools like ChatGPT, Claude, Gemini, and Microsoft Copilot, user expectations have shifted. The natural language interface has not only become a standard for AI systems but also a key feature for many traditional applications.

Something similar may happen with agentic platforms. As major AI developers roll out new capabilities, they will be exposed to enormous user bases. This will help cement expectations for user interfaces (UIs). Users will become accustomed to systems that solve problems autonomously in the background, with periodic queries for approvals. Expectations are normalizing around systems that don't just answer; they act within guardrails. This is why enterprises need to keep an eye on the emerging trends with UIs.

The Evolution of Data Interfaces

"What were our top-selling products last quarter?"

On the surface, this is a straightforward query. But until recently, answering it was extremely challenging. It required SQL knowledge, database access, understanding of table structures, and often multiple queries to aggregate and filter the right data.

That's now changing. The rise of natural language interfaces is beginning to democratize access to enterprise data, allowing anyone—not just analysts or developers—to ask questions in plain English and get meaningful, real-time answers.

A quiet revolution is happening in how we interact with technical systems. While this trend is still in its nascent stages, there are some innovative tools shedding light on what to expect in the future.

A Glimpse into the Future with Agentic Development

Modern AI integrated development environments (IDEs) and notebooks point to the pattern of using structured data and metadata. Take Cursor as an example of where interfaces are heading. Since launching in March 2023, it's grown from zero to $500 million in ARR, hit a $9 billion valuation, and now processes over one million queries per second while producing nearly a billion lines of production code daily.

What Cursor shows us is the shift from manual prompting to goal-driven collaboration. Users state what they want. The system then figures out how to get there, proposes a plan, drafts code, runs tests, and opens a pull request (PR).

Here's what this could look like for data engineering. Suppose you've been assigned to build a weekly ETL pipeline to aggregate customer activity, enforce data quality standards, calculate summary metrics, and push the final output into production. Under traditional workflows, you'd be piecing together SQL queries, Python scripts, data validation routines, and CI/CD configurations by hand. But in an agentic system, the process starts differently. You define your goal in natural language:

> Create a weekly customer activity ETL pipeline. Include data quality checks for nulls and duplicates, calculate weekly active users, and push summary tables to the analytics warehouse.

From that point, a multi-agent system gets to work. It scans your project, taking into account schema definitions, naming conventions, current pipeline structures, and your warehouse configuration. This comprehensive understanding allows it to outline a detailed, coherent plan. For instance, an agent proposes creating a new model, `customer_weekly_activity.sql`; drafting Python scripts for anomaly detection; and preparing orchestration configs aligned with your tech stack. The plan comes annotated with rationales and implementation steps.

Once you approve it, an agent transitions seamlessly into automated development. It writes out the SQL aggregation logic, test scripts to check for nulls and duplicates, CI/CD configurations tailored to your stack, and optional README updates. Everything is formatted to match your project's style guidelines, versioned correctly, and staged for review.

Next comes execution and validation. An agent runs the full pipeline in a sandboxed or development environment. It executes the SQL, initiates the data quality scripts, and runs your test suite along with any CI checks. Upon approval, the agent applies the change and re-runs the pipeline. This kind of real-time feedback loop ensures that problems are caught and resolved before—not after—human review.

Finally, when the pipeline passes all checks, an agent handles deployment. Depending on your preferences, it might open a pull request with a summary of changes and test results, or directly push the updates to your orchestration layer. You receive a preview of diffs, validate the output, make any final adjustments, and merge. You do this without ever needing to write deployment scripts or manually trigger the pipeline. Such a sophisticated system would lead to high levels of productivity and agility.

However, for it to operate safely and autonomously in complex environments, you need more than just instructions and advanced AI. There must be structured data and metadata (or *structured context*): the schemas, semantics, relationships, permissions, and lineage that describe how your data works.

Why Structured Context Is Critical

GenAI has earned its spotlight largely thanks to what it can do with unstructured data. These generative models are impressive at summarizing documents, generating content, translating languages, and pulling insights from dense, text-heavy sources like emails, research papers, and internal reports. The ability to sift through and make sense of unstructured information has captured considerable attention, and rightly so.

But here's where things get interesting. The emergence of agentic AI has pushed enterprise companies to rethink their entire data foundation. They're no longer just chasing better conversational interfaces. They're gearing up for intelligent enterprise automation at scale.

In this shift, structured context moves from "nice to have" to "non-negotiable." It's what makes AI ready for enterprise workflows and processes. You can think of structured context as the operating system for the next generation of AI.

The Two Critical Integrations

To make structured context actionable, it needs to connect seamlessly to systems that hold and describe your data. This is why two integrations are essential:

Structured data
: This refers to the rows in your data warehouse or lakehouse. This is information from customer relationship management (CRM), enterprise resource planning (ERP), human capital management (HCM), and other enterprise systems.

Structured metadata
: This includes data about models, sources, lineage, dependencies, tags, and governance tags. This is essentially a map of your data ecosystem.

With structured data and metadata, AI agents can discover tables, understand relationships, check quality, and plan safe actions. This allows for powerful conversational analytics, which is powered by planning, reasoning, and autonomous decision-making. But it also has a layer of governance. It is what keeps actions safe: policies and permissions embedded in metadata that determine who can access which datasets, which transformations are permitted, and

how sensitive fields must be handled. In short, structured context is the difference between plausible answers and trustworthy actions. For example, suppose someone in marketing has a question about churn rates and prompts a standard chatbot for information. The answer will show the details about the data's origin, as well as the definition of the terms. This will be based on the underlying structured metadata.

Another use case involves AI-powered agentic development. A data engineer can leverage the structured metadata to carry out a complex data migration. The system will understand how to manage features like stored procedures and the differences among the databases.

Three Key Capabilities

Structured context equips agents with three key capabilities:

- Memory via metadata so they know what assets exist and how they relate
- Boundaries via clear definitions, permissions, and rules so they don't wander outside guardrails
- The ability to take useful actions with validated tools to read/write safely

When you combine these, agents evolve from being just chatbots and start acting like reliable teammates. They can plan, reason, and execute tasks at scale. We're already seeing agents autonomously modify data pipelines, fix errors, manage migrations, and spin up new data products, such as by using orchestration frameworks like Airflow, Dagster, or Prefect. They can be always-on, embedded in the infrastructure. All this is driven by structured inputs and aligned with business logic.

That's the real unlock. As companies move from pilot experiments to production-grade agentic AI, this structured foundation ensures their systems aren't just delivering impressive responses but taking trustworthy, business-critical actions. The magic happens when you combine agentic AI with structured metadata, creating scalable, accurate, and governed systems that organizations can truly rely on. The practical "how" of exposing structured context to AI systems is covered in Chapter 2.

When AI Gets It Wrong

Weak governance and poor inputs have predictable consequences. Inside many companies, AI initiatives underperform: studies often cite that more than 80% of AI projects don't succeed. That's not just a little worse than traditional IT projects; it's nearly double the failure rate. And in most cases, it comes down to something incredibly basic: bad data. Models built on unreliable or incomplete information don't just underperform; they can quietly drain revenue. Gartner puts the average cost of poor data quality at around $12.9 million a year per organization. In some cases, companies lose as much as 6% of their annual revenue from flawed AI outputs.

Real Consequences of AI Failures

High-profile examples show the impact of flawed AI. For example, a major airline was taken to court after its chatbot promised a bereavement fare refund to a customer who had just lost a loved one. The customer followed the chatbot's advice, only to be told later that the discount didn't exist. Oddly, the company argued that the chatbot was its own legal entity and responsible for its own behavior. The tribunal didn't buy it. The airline had to pay the customer over $600 and was held accountable for the AI's error.

Lawyers have faced sanctions for submitting briefs citing fictional cases. News sites have published AI travel content directing readers to unsafe destinations. And it's getting worse—OpenAI's newest models hallucinate at higher rates than their predecessors, with error rates hitting 48% and 33% for the o4-mini and o3 models, respectively.

The pattern is hard to ignore. As organizations rely more heavily on AI to make sense of their data, the quality of that data and the structures in place to manage it become mission-critical. Generative models can be powerful tools, but without clear constraints and reliable inputs, they risk misleading as much as they help. When AI gets it wrong, it's not just a glitch. It can lead to legal exposure, reputational damage, and costly operational missteps. The response is not merely "better models," but *disciplined governance*, like quality checks, lineage, permissioning, and policy enforcement throughout the pipeline.

Regulatory Considerations

Of course, when an AI implementation fails, it can trigger serious regulatory consequences. Regulators are making these guardrails explicit. Under the EU Artificial Intelligence Act, especially Articles 10 and 27, these failures point to deeper compliance risks that organizations cannot afford to ignore.

Article 10 mandates that high-risk AI systems use datasets that are complete, accurate, representative, and error-free. Organizations must document everything: data sources, annotation methods, quality checks, and bias mitigation. Without this documentation, you're noncompliant.

Article 27 goes further, requiring Fundamental Rights Impact Assessments (FRIAs) that examine fairness, dignity, and nondiscrimination—especially critical for AI used in hiring, credit scoring, healthcare, or education. Companies must map data flows, retention policies, oversight mechanisms, and risk mitigation, sometimes reporting to regulators.

While FRIAs can be aligned with Data Protection Impact Assessments (DPIAs) to avoid redundancy, they still require organizations to map out not just where data comes from and how it's used, but also how it might affect people, including vulnerable groups. Companies must document data flows, retention policies, oversight mechanisms, and risk-mitigation steps. In some cases, they're also required to report FRIA outcomes to regulatory bodies, especially if fundamental rights could be impacted.

Regulators are also pushing for continuous oversight. This means companies must implement full data lineage tracking, robust metadata infrastructure, role-based access, and systems capable of automatically flagging issues as they arise. If a dataset changes, such as with schema shifts, updates in source systems, or inconsistent labeling, organizations are expected to respond in real time.

What the EU Artificial Intelligence Act makes clear is that data quality and governance can no longer be treated as just an IT or performance issue; they are legal obligations that need to be baked into every layer of the AI pipeline.

Articles 10 and 27 effectively turn structured data governance into a compliance discipline. To meet these standards, companies must build systems that deliver not only clean and reliable data but

also transparency, traceability, and accountability. They need to be embedded into every layer of the AI pipeline. Real compliance means more than checking boxes. It means engineering a governance framework that is automated, auditable, and built to scale.

The Industry Response

The good news is that the technology industry is actively working to fix what's broken in this new AI-infused era. The influx of venture capital into the space is proof enough.

One major trend is the consolidation and unification of data stacks. After years of best-of-breed fragmentation, many organizations are seeking more unified platforms to reduce complexity. Vendors are converging on this context-first, governance-forward model. Quality and policy controls are moving closer to where queries run and transformations execute, so guardrails are enforced in flow rather than audited after the fact.

Industry observers predict that GenAI will drive a shift from highly fragmented stacks toward unified data platforms. Data quality and observability are receiving an AI boost as well. Monitoring dozens of pipelines and tables for issues can overwhelm human teams. Therefore, vendors are adding AI/ML capabilities to detect anomalies or quality issues in real time. New features in cloud data platforms can automatically monitor freshness, null spikes or other data health metrics, and alert teams before issues propagate downstream.

Behind these improvements sit the same ingredients emphasized previously: well-modeled data, accurate metadata, enforceable policies, and continuous signals about data health.

The Changing Role of Data Engineers

No doubt, this deeper integration of intelligence into the data stack signals dramatic changes for the role of data engineers. Tasks that once defined the profession, such as building ingestion pipelines, wrangling schemas, writing ETL logic, monitoring data flows, and managing break-fix incidents will increasingly be handled by intelligent systems. These systems will go beyond routine automation, learning to optimize themselves, detect anomalies in real time, resolve issues independently, and enforce governance frameworks.

Yet this evolution does not mean data engineers are becoming obsolete. As routine responsibilities fade into the background, engineers are shifting their focus to more strategic concerns. They're now tasked with designing resilient and adaptive data architectures, validating the integrity and semantics of AI-driven pipelines, maintaining rigorous standards for data quality, and embedding ethical and compliance principles into the core infrastructure.

In this new environment, data engineers are evolving from system operators to system stewards. The work now demands fluency in AI-native tools, semantic data modeling, governance strategy, and the supervision of autonomous agents. Collaboration is also evolving, as engineers engage more deeply with product, compliance, and analytics teams to ensure that data infrastructure aligns with broader organizational goals. The shift isn't about doing less, but about doing more of what truly matters.

For this new era of data engineering to become a reality, there must be a solid foundation where large language models (LLMs) effectively interact with structured data and metadata. Otherwise, the capabilities remain shallow and disappointing, as they're not grounded in the relevant data, processes, and workflows of the enterprise.

Takeaways

AI is reshaping the enterprise data stack from the ground up. What used to be a rigid setup, built for specialists and stitched together with siloed tools, is starting to feel more conversational, smarter, agentic, and quicker to respond. It's about weaving intelligence into every layer: data ingestion, governance, and how users interact with the data. But for all that to work, you need a modern metadata foundation.

CHAPTER 2

The Structured Context Interface for Governance and Trust

In just a few years, prompt engineering has turned into a cottage industry. There are many books on the topic, as well as seemingly endless blogs and LinkedIn posts. Then there are the workshops, YouTube videos, and courses.

The major LLM providers, such as OpenAI, Anthropic, and Google, have published their own guides. For example, Google has written a 68-page handbook by Lee Boonstra. It covers topics like how to change the creativity of responses and craft prompts for multistep reasoning.

Yet all this points to something interesting: prompt engineering is a clear sign of the limitations of chat-based LLMs. They are brittle, as they rely on probabilistic transformer models. After all, with an application like ChatGPT, Claude, or Gemini, you will likely get a different response with the same prompt. Even minor changes in a word or punctuation can result in wildly different outputs.

True, this unpredictability is fine for summarizing long PDF documents or evaluating customer reviews. But it is far from ideal when it comes to working with complex enterprise workflows, which need to be reliable and consistent.

As for agentic AI, it certainly holds much promise to address the issues of chat-based interfaces. This should mean that prompt engineering will fade in importance. But successfully deploying agentic AI systems will be challenging. According to a report from Gartner, more than 40% of these projects will be canceled by the end of 2027. The firm lists a variety of reasons for this, including the high costs, ineffective risk controls, and unclear business objectives. Despite this, Gartner remains optimistic about agentic AI. The firm forecasts that at least 15% of daily work decisions will be made autonomously by 2028.

But success with generative and agentic AI for data engineering requires a structured context interface, a reliable way for AI systems to interact with structured data and structured metadata under policy (Figure 2-1). This will unlock major gains in productivity and agility because it brings consistency, permissions, lineage, and definitions directly into the loop where assistants and agents operate.

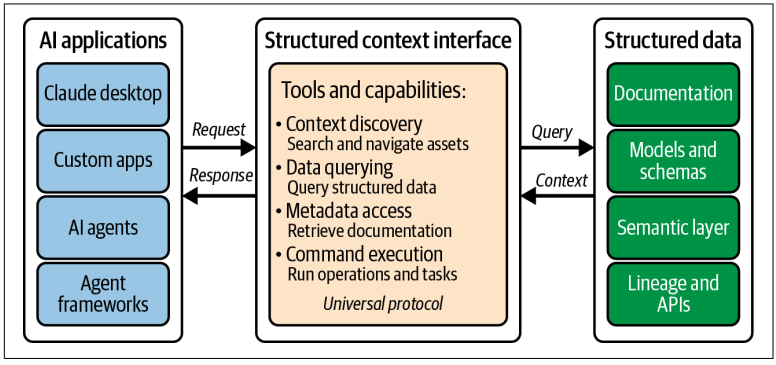

Figure 2-1. How the structured context interface works

The Tools: Model Context Protocol as the Plumbing

Simply put, a structured context interface is about the interactions between structured context and AI, which is usually an LLM or small language model (SLM). There are different ways to make the connections, but the one that has been gaining significant adoption is model context protocol (MCP).

Understanding MCP

In November 2024, Anthropic introduced the open MCP standard. The company defined it as a way "for connecting AI assistants to the systems where data lives, including content repositories, business tools, and development environments."

Before this, the development of agentic applications was cumbersome because of the need to write integrations to implement tools with LLMs. This became known as the "M × N problem." That is, for every *M* LLMs that must be supported, you need *N* connectors for each tool (Figure 2-2).

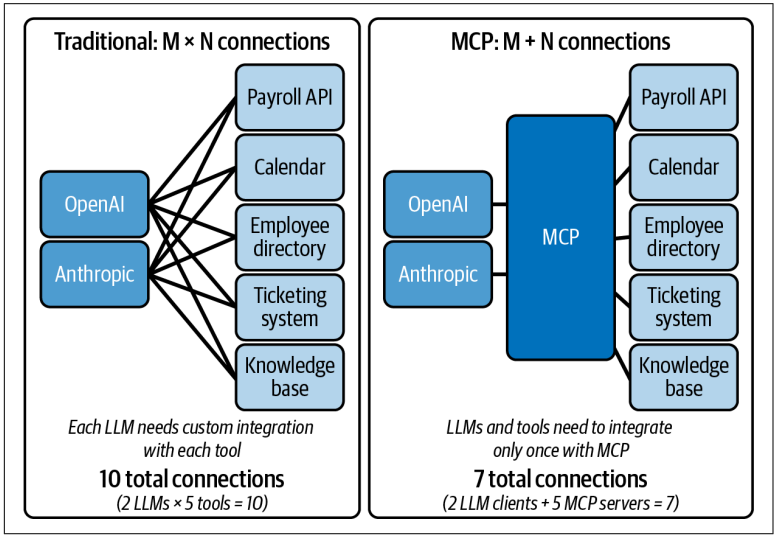

Figure 2-2. Traditional integration versus MCP

For example, suppose you are building an HR assistant. Let's say you will use two models, one from OpenAI and another from Anthropic. The application will also use five tools: a payroll API, calendar, employee directory, ticketing system, and knowledge base. For this, you will have to build 10 connectors, for example, by using OpenAI's and Anthropic's software development kits (SDKs).

By contrast, using MCP changes the formula to M + N. This is how it would work for our HR application:

- You will deploy five MCP servers, or one for each tool.
- You will use two MCP clients: one for OpenAI and one for Anthropic.

This has reduced the connectors to seven. However, there are more advantages to using MCP. Keep in mind that there are thousands of servers available as open source repositories. Moreover, OpenAI, Microsoft, and Google have adopted the standard.

Another benefit of MCP is that there is a clean separation of responsibilities between the server and client. The server focuses on the backend tool logic and data access, whereas the client makes connections with a JSON-RPC interface. This modularity provides for more maintainability and scalability, as there is much less complexity when adding models and tools.

MCP Alternatives and Ecosystem

In terms of a structured context interface, MCP is not the only approach for integrations. There are alternatives. But currently, MCP is in the leadership position and is building a strong ecosystem.

Something else to consider: a structured context interface is essentially an architectural pattern. This means there are various methods for the implementation. For example, a dbt platform can operate as an MCP server, allowing for the integration of AI applications with data warehouses. It's available to users via Cloud CLI, API, the Discovery API, and the semantic layer. With it, you can access private APIs, text-to-SQL, and SQL execution. What this means is that you can connect a dbt project with structured data and metadata to any MCP client, like Claude Desktop Projects, Cursor, custom apps, or agent frameworks. This makes structured context addressable and enforceable through a consistent, auditable interface.

How the Interface Works in Practice

With this structured context interface, an LLM will recognize when it needs to answer a question with structured data and metadata and which tool to call. Here are some use cases:

General knowledge

Prompt: What is the capital of the United States?

Response: The LLM will know the answer based on its training data and model knowledge.

External or real-time queries

Prompt: What is the weather going to be in Washington, DC, tomorrow?

Response: The LLM will understand that this will require external information that is not likely to be in structured data. Instead, there will be a call for a web search to retrieve the weather data.

Enterprise metrics

Prompt: How many customers do I have in Washington, DC?

Response: The LLM will recognize the need for structured context. For this, there will be a call for the `query_metrics` tool to access the correct information.

However, in an enterprise environment, a structured context interface needs to carry out these types of functions with strong governance and trust, whether a human or an agent initiates the change. Let's look at this in more detail.

The Governance: Making AI Safe and Compliant

Once AI assistants can act (not just answer), governance becomes a prerequisite. Yet despite its importance, most organizations are falling short. Gartner's 2025 survey paints a stark picture. Only 12% of companies have put a dedicated AI governance framework in place, and 55% still have no formal structure at all.

This gap comes with real consequences. According to Gartner, poor governance can lead to increased costs, failed AI projects, and damaged reputations. Without guardrails, AI and agent-based systems can introduce bias, violate privacy laws, trigger regulatory fines, and erode trust among both users and customers.

Deloitte's research adds another dimension. They found that organizations that implement an effective governance system tend to see higher AI adoption rates and stronger revenue growth. Here are a few best practices to keep top of mind with governance:

Policy enforcement and access control
: Role-based and attribute-based controls should govern AI agents and human users alike. This ensures only the right people and processes can access sensitive data.

Lineage, versioning, and auditing
: Track every dataset change from ingestion to transformation to final consumption by AI agents.

Embedded structured metadata and quality scores
: Your structured context interface should include rich structured metadata that agents can surface alongside the data. This empowers both agents and users to assess the relevance and credibility of inputs.

Legal and regulatory compliance
: Codify policies to mask or anonymize personally identifiable information (PII), enforce data minimization, and honor user rights under laws like the General Data Protection Regulation (GDPR), including erasure requests.

A modern structured context interface will enforce these approaches. For example, MCP supports OAuth 2.0 as its main authentication and authorization system. Next, a semantic layer will define an organization's metrics and dimensions, allowing for strong governance. Finally, a structured data interface will have SQL validation. This can be built into an MCP server.

Building Trust: The Gap Between Adoption and Confidence

As AI systems start making more decisions on their own, trust has never been more important. Without it, even the most sophisticated tools can end up unused. A recent KPMG study, "The American Trust in AI Paradox: Adoption Outpaces Governance", published in 2025, makes this clear. While 70% of US workers are leveraging AI's benefits and 61% say it's already improving their jobs, about 75%

still feel uneasy about its potential risks. In fact, only 41% say they truly trust these systems.

This gap in trust shows that simply rolling out AI tools does not guarantee they'll be integrated effectively. Without strong governance, transparency, and clear controls, these tools can easily be pushed aside.

The Shadow AI Problem

The survey also found that nearly 44% of US workers are using AI outside official company channels. This often involves uploading sensitive data to public platforms. What's more concerning is that 58% rely on AI outputs without checking them for accuracy or compliance, and 57% admit they've made mistakes in their daily work because of AI. Then there are 56% who say they hide their use of AI from their managers—they will pass off AI-generated work as their own.

All of this points to a deeper problem. When policies are unclear, training is lacking, and oversight is limited, AI tools turn from productivity boosters into hidden risks. Without greater transparency, solid evaluation processes, and a culture that treats AI as a governed partnership rather than a secret shortcut, organizations will continue to struggle with fragile trust and results.

Core Pillars of Trust

Trust is about having systems and practices that give both people and AI agents confidence that data and outputs are accurate, explainable, secure, and aligned with the organization's goals and values. Here's a closer look:

Data quality and validation
: Trust starts with reliable data, which must be tested, validated, and monitored at every step. Automated checks for uniqueness, unexpected schema changes, distribution shifts, and stale or unusual data act as constant safeguards.

Explainability and transparency
: AI outputs only have real value if people understand how they were generated. This means using explainability tools that show the reasoning behind model predictions. They should also be backed by rich metadata and context.

Continuous oversight and feedback loops
: Trust must be maintained over time through constant monitoring of model performance, drift detection, and dashboards tracking indicators like test pass rates, incident resolution times, and metadata completeness.

Building Trustworthy AI Starts with a Culture of Transparency

For any enterprise that aims to build AI systems people can trust, it starts with a culture of transparency and alignment. Trust needs to be woven into everyday teamwork.

More organizations are seeing the value of bringing different groups together. Data engineers, analysts, privacy and legal teams, and business leaders are working side by side to define what "trusted" really means, whether that's agreeing on clear data definitions or setting the right risk limits. When teams align on these fundamentals from the start, they break down silos, reduce misunderstandings, and avoid ethical or regulatory pitfalls.

Embedding trust early in development, or "trust by design," is another critical success factor. This includes building bias checks, structured metadata tagging, and explainability reviews directly into CI/CD pipelines. These proactive measures surface potential issues sooner and foster a shared sense of accountability.

Of course, trust is only meaningful if it's measured. It's important to track metrics such as how often shared data assets are used, mean time to detect and fix incidents, and data retrievability rates. Direct user feedback on confidence in AI systems is becoming part of the equation, too. For example, the icons for thumbs-up/thumbs-down for the responses from ChatGPT are a powerful use case of feedback from users.

Some companies are taking it further by publishing internal trust dashboards. These dashboards correlate governance activities with user confidence surveys to pinpoint where improvements are needed.

The Payoff: Why This Is Strategic, Not Just Technical

Structured context for AI insights creates a trusted, reliable data layer that organizations can utilize across teams. This common foundation provides consistent key performance indicators (KPIs) and metrics that help different departments work together more effectively while maintaining regulatory compliance.

What does this actually look like day to day? Teams across the organization—from marketing to finance to operations—can query the same governed metrics and get consistent, trustworthy answers. When everyone's working from the same definitions and validated data, collaboration improves and compliance becomes automatic rather than an afterthought.

The C-Suite Takes Notice

This major shift is also reaching the C-suite. Gartner reports that 70% of chief data and analytics officers (CDAOs) are now leading enterprise AI strategies. For example, the share of CDAOs reporting directly to CEOs has jumped from 21% in 2024 to 36% in 2025.

These leaders understand that technology is not intrinsically valuable. It has to drive real business outcomes. Their role is to ensure that data architecture, governance, and trust are closely aligned with the company's broader goals.

The Architecture for AI-Native Data

What does an AI-native data architecture look like in practice?

Unified semantic layer
: A single source of truth for metrics and dimensions that both humans and AI agents can query consistently. This eliminates the confusion that comes from multiple definitions of the same business concept.

Embedded governance
: Policies and controls are built into the data layer, not bolted on afterward. This means access controls, data quality checks, and compliance rules are enforced automatically as data flows through the system.

Real-time quality monitoring
: Automated systems that detect and often fix issues before they impact downstream users. These systems use machine learning (ML) to learn normal patterns and flag anomalies immediately.

Transparent lineage
: Complete visibility into data flow from source to consumption, enabling both debugging and compliance. Every transformation, every join, every aggregation is tracked and auditable.

Federated but coordinated
: Different teams maintain their domains while operating within a common framework. This balances autonomy with consistency, letting teams move fast while maintaining enterprise standards.

This isn't just technical architecture—it's organizational transformation. Companies that get this right will have AI systems that are fast, accurate, trustworthy, and compliant.

The Business Impact

Organizations with well-implemented structured context interfaces and governance see measurable benefits:

Faster time to insight
: Natural language queries eliminate the analyst bottleneck.

Higher data utilization
: When data is discoverable and understandable, it actually gets used.

Reduced compliance risk
: Automated governance means fewer regulatory violations.

Improved decision quality
: Trustworthy data leads to confident decisions.

Accelerated innovation
: Teams spend less time on plumbing and more time on value creation.

The companies making this investment today will be the ones whose AI systems actually work when it matters most.

Takeaways

Prompt engineering highlights how unreliable chat-based models can be. You're never sure what they'll come back with. Structured context interfaces shift this dynamic. They allow us to create AI workflows that are stable, scalable, and governed right from the outset.

They're strategic tools that connect business objectives with AI capabilities. As companies move from experimental pilots to systems that sit at the heart of their operations, their success will depend on how well they build and manage these interfaces. Without them, AI remains a risky black box. With them, it can become a trusted partner in making decisions that matter.

The enterprises that win in the AI era won't be those with the best models or the most data. They'll be the ones that build the strongest foundation—where structured context, clear governance, and earned trust come together to enable intelligent automation at scale.

The choice is clear. Build the right foundation now, or spend years trying to retrofit governance and trust into systems that were never designed for them. As enterprises step into this new era, success will not come just from picking the right AI tools. It'll come from rethinking the data stack itself, designing it to support trust, agility, and growth in a world where AI is quickly becoming the norm.

CHAPTER 3

Optimizing Data Discovery for AI Systems

At a high level, the concept of discoverability is straightforward. It's about how easy it is to find, understand, and trace the data, logic, and decisions for an AI system. Ultimately, this helps to build trust with users, customers, and regulators. It can also improve usability. When it becomes easier to get relevant information, there is likely to be more adoption and stronger ROI.

Yet discoverability is something that does not get enough attention. Consider a McKinsey survey. It found that 40% of enterprises said that explainability was a major risk, yet only 17% looked to address it seriously. The survey also showed that 91% of the respondents thought their organization was not "very prepared" to scale generative AI responsibly and safely.

This helps to explain why AI often fails to live up to expectations, as highlighted in a dbt Labs survey. Even though 80% of the teams reported using AI, the accuracy for natural-language-to-SQL queries remained inconsistent. Then again, the LLMs lacked access to semantics and lineage. There was also the problem of context being scattered across warehouses, BI tools, and wikis.

Discoverability is complex and challenging. But it is critical for effective AI. In this chapter, we'll see what you need to focus on to successfully add discoverability to your AI system.

Overcoming Discoverability Gaps in AI Systems

Enterprises often struggle with data fragmentation, inconsistent semantics, inadequate metadata, poor lineage, and weak discovery mechanisms. In other words, they lack the key components of structured context needed for their AI projects to succeed. In this section, we'll highlight these common discovery gaps and introduce practical solutions, including discovery indexes, semantic layers, and a Discovery API.

Scattered Data and Context

Many enterprises have a maze of data warehouses, BI tools, catalogs, notebooks, and wikis. This data fragmentation means that there is no central reference point for AI systems. Without a reference point, AI systems pull in random assets, making retrieval unreliable and inconsistent.

A better approach is to create a semantic layer that consolidates the organization's data sources and metadata. This includes models, sources, exposures, owners, tags, tests, freshness indicators, and lineage. For example, a dbt Discovery API can expose this structured context. It allows AI agents or other applications to reliably query and act on consistent and governed data. Once this is in place, the next step is to create a *discovery index* of these sources. The index serves as an AI system's definitive starting point, the first call when it needs to answer a query or resolve an ambiguity.

Inconsistent Semantics or Metric Drift

Inconsistent semantics, or *metric drift*, happens when the same KPI or metric is defined differently across teams, tools, or business units. For example, "customer churn" might include only subscription cancellations in one team's dashboard, while another team counts both cancellations and payment failures. When an AI assistant or a natural-language-to-SQL engine queries this data, it can produce different answers depending on which definition it uses.

This can be particularly troublesome for systems that rely on transformer-based AI models. Without access to an organization's formal definitions, these models predict meaning statistically. This means they generate responses that *seem* likely rather than what's

actually correct. As a result, this can lead to misinterpretations of key metrics and relationships.

The solution is to implement a semantic layer. As we noted in Chapter 2, the semantic layer acts as a single source of truth for metrics and dimensions that both humans and AI agents can access. The semantic layer provides standardized naming conventions for domains and entities, metric definitions, and more. Whether the request comes from a BI platform, an AI assistant, or an SQL query, everything draws from the same set of definitions. In other words, instead of allowing each team to create its own version of key metrics, the semantic layer establishes consistency and alignment across the organization.

Thin or Stale Metadata

Thin or stale metadata is when datasets lack clear ownership, contain weak documentation or tags, or have outdated freshness checks and tests. The lack of clear sensitivity labels—which classify data based on its confidentiality or regulatory requirements—makes the problem worse. This leaves AI models unable to distinguish between deprecated, low-quality data and business-critical information. Without a strong metadata foundation, agents won't know what assets exist, how they relate, and whether the data is trustworthy. The result is unreliable outputs, wasted analyst time, and reduced confidence in data-driven decisions.

Enforcing rich YAML metadata across all datasets addresses this gap. Each data asset's metadata should include:

- Clear ownership details: the person/team responsible for the asset
- Descriptive documentation: where the data comes from, when it was created, etc.
- Meaningful tags: relevant, precise, and consistent keywords and phrases (e.g., customer churn or monthly revenue)
- Automated freshness, uniqueness, and not-null value tests to ensure your data is up-to-date and doesn't contain duplicate or missing values
- Well-defined sources, exposures, and model contracts to strengthen accountability and reliability

This rich metadata should be published through documentation portals and exposed via a Discovery API to ensure both humans and AI systems can easily access standardized, up-to-date information.

Opaque Lineage and Change History

Opaque lineage and change history is when it is challenging to see upstream dependencies, recent changes, or the potential blast radius of an update. Because of this, agents and users may choose the wrong datasets or models without understanding the consequences.

Your data's lineage and change history should be fully transparent in your catalog and search interfaces. Understanding not only what has changed but also where the change originated helps teams and AI assistants evaluate the reliability of an asset before using it. Plus, clear visibility into recent updates and dependencies means fewer surprises and faster troubleshooting when something breaks.

To achieve this, organizations should leverage end-to-end Directed Acyclic Graph (DAG) lineage. DAG lineage maps the complete flow of data from sources through transformations to downstream dashboards and applications.

In a typical modern data stack, this flow progresses from raw sources through staging models, intermediate transformations, and marts. They are fed into a semantic layer that powers analytics and ML applications. You can find an example in Figure 3-1.

When combined with testing outputs, reports, and change tracking in CI pipelines, this gives a clear picture of how updates impact the entire system. Triggering automated impact analysis on top of this DAG lineage ensures that teams understand the blast radius of every modification before it reaches production.

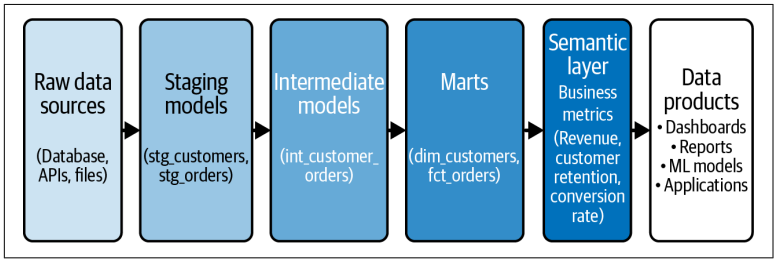

Figure 3-1. A DAG lineage map

Search That Ignores Structure

A common data error is when search returns the wrong information because it lacks the proper context for the data. This happens when a keyword or vector search is not grounded in the data's structure, meaning, or business rules. When AI embeddings retrieve tables, they may be close but still wrong. These inaccurate searches can lead to poorly informed business decisions, improper use of data, or even unauthorized access to data.

To prevent these issues, you need an effective structure and signals that guide human users and AI applications. This requires enriching the search with a combination of everything described in the previous discovery gaps section: semantics, lineage, and quality indicators, plus owner and policy metadata for better accountability. The goal is to get results that are contextually similar but valid and reliable.

The ideal method for enabling this is a strong Discovery API to expose the rich metadata.

Governance: Making AI Safe with Data

Finding the right data is only half the battle. There also needs to be a focus on the governance of the AI agent. They need to use data securely and in a way that humans can trust.

For example, an AI agent may retrieve the correct assets yet bypass the masking or row-level security, which exposes sensitive PII data. To avoid this problem, all the queries need to be routed through a system like the dbt Semantic Layer, which automatically applies security rules.

Another use case is with self-service access, such as for tickets. In this scenario, provisioning can block an AI agent from properly functioning. However, by using a request-access flow, the system can automatically issue short-lived, scoped roles and revoke them after use. This gives the AI agent the access it needs without creating long-term security risks.

AI agents can also run into trouble when they execute expensive or unsafe SQL queries without any checks. To prevent this, every run should start with a preview or dry run. This allows for catching

errors early, estimating the costs, and executing the query in a sandbox by default.

Trust in the results generated by AI also depends on evidence. If an answer has no supporting proof, humans are less likely to find it useful. Every result should include context such as the metric definition, the owner, the data source, lineage, and test status. This makes the output more trustworthy and reusable across teams.

Finally, agents should never be able to push direct edits that bypass review. A safer practice is to require all changes to go through pull requests. This ensures continuous integration tests run and a human approves the change before it merges. That way, AI can contribute to workflows without undermining governance.

Preparing for Better Discoverability

To build a strong foundation for discoverability and governance, there are some best practices to consider. First, you need *metadata hygiene*. This means that every tier-1 asset—any critical dataset, model, or dashboard for decision-making—needs to have an assigned owner, a clear description, relevant tags, sensitivity labels, and freshness. With this structure, humans and AI systems can identify if a dataset is reliable and appropriate for use. Then, a Discovery API can be used to locate the metadata and make it available for use in catalogs, search, and AI agents.

Next, you need well-defined metrics and dimensions for the semantic layer so that AI agents can accurately resolve natural-language intents by using governed concepts, not ad hoc tables. This requires standardized taxonomies and naming conventions for domains, entities, metrics, and columns. With this level of consistency across teams, searching becomes sharper and the AI reasoning more accurate. Publishing a discovery index from the modeling system will support these outcomes. For example, using a tool like dbt artifacts with a Discovery API will allow you to rank signals for freshness, the test pass rate, usage, sensitivity, and lineage depth.

Finally, you need to capture and surface lineage and change history. When users and agents can see upstream dependencies and "what changed" before they act, they are less likely to pick the wrong dataset or introduce errors. Retrieval itself must be context-aware: filtering by schema, semantics, and lineage first, and only widening

the scope with embeddings when necessary. Relying only on vector similarity searches—which match text based on meaning rather than structure—is too risky in an enterprise setting.

Takeaways

As seen in this chapter, there are many risks when discoverability and governance are weak or ineffective. The challenges include scattered data, inconsistent semantics, stale metadata, opaque lineage, and unstructured search.

These problems become even worse when using complex generative AI models or agents. They often fall back on guesswork, which can lead to inconsistent or wrong responses, security failures, and loss of trust.

A disciplined approach is needed. This involves strong metadata hygiene, semantic alignment, lineage visibility, and context-aware discovery. At the same time, governance needs to be embedded directly in how AI agents request, use, and act on data. The goal is to strike a balance between speed and safety. Some of the best practices include self-serve access flows, preview-first execution, PR-only changes, and evidence-backed responses.

The path forward for enterprises is clear: there must be serious investments in discoverability and governance. When enterprises get this right, AI agents can shift from being fragile black boxes to becoming trusted copilots that accelerate insight, strengthen compliance, and unlock real business value.

CHAPTER 4

The Value of a Structured Context Foundation

A report from International Data Corporation (IDC) predicts that enterprises will spend $151.1 billion on generative AI solutions by 2027. These massive investments put intense pressure on executives. A 2025 Dataiku/Harris Poll survey found that 74% of CEOs said they could lose their job in the next two years if they do not demonstrate tangible results from their AI investments. But making the right strategic decisions is increasingly challenging. The technologies are complex and quickly evolving. Then there is the hype and exaggerated AI claims, which make it difficult to evaluate solutions.

So, what do you do?

To break through the noise, in this chapter we'll first explore how the three core AI capabilities are converging around a structured context interface. We'll then look at the playbooks of leading organizations that are already having success with this approach. Next, we will examine the emerging standards and trends that are shaping the future of the ecosystem. Finally, we'll outline practical steps for implementing these strategies to drive measurable impact from your AI investments.

AI Capabilities Are Converging Around Structured Context

Throughout this report, we've seen how three core AI capabilities—conversational analytics, copilots, and agentic workflows—are fundamentally changing the data stack. We've shifted from static to dynamic dashboards and natural language interfaces with data, allowing for ways to augment human work and orchestrate complex, multistep processes autonomously.

But you shouldn't look at these three core capabilities in isolation. They are converging, allowing for better ways to interact with data. At the heart of this process is the structured context interface that provides AI systems with the right data, metadata, and governance.

Let's look at an example of how these capabilities converge in a real-world workflow. Suppose a sales manager opens a copilot interface and asks, "What's the pipeline velocity by segment for the last 30 days?"

Behind the scenes, several coordinated AI capabilities come into play:

Conversational analytics layer
: The copilot interprets the request and passes it to the semantic layer, which translates it into the proper query. The structured context ensures metric consistency; that is, the pipeline velocity uses the same definition across all systems.

Proactive assistance
: The copilot anticipates what the user might need next, such as more advanced cohort analysis or a suggested SQL query.

Agentic workflow in the background
: While the user explores the insights, an agent monitors the data pipeline. It detects issues that could affect accuracy, like duplicate lead-to-account joins, and automatically initiates a remediation workflow. For this, the agent may suggest a pull request to update the deduplication rule, run new data quality tests, and validate changes in a sandbox before human approval and merge.

Together, these elements show how structured context acts as the connective tissue between conversational analytics, copilots, and agentic workflows.

Strategic Moves from Leading Organizations

The revenue operations example shows the potential of the structured context interface as the center of AI capability convergence. But how does this work in practice? Let's look at some strategic moves from leading organizations that demonstrate the rewards of getting it right.

Database Migration

A modern AI platform requires a strong data foundation. But some organizations are reluctant to take on database migrations because they can be extremely complicated for data engineers, requiring significant planning, tedious activities, and high costs. However, avoiding migrations can mean missing out on the opportunities of digital transformation and cost savings in the long term.

Take the case of a company that wanted to move an extensive Microsoft SQL Server implementation to Snowflake. If this project had been done with a manual conversion, experienced data engineers would typically expect it to take 18 months or longer. By using AI migrations agents that utilized dbt MCP tools containing the right structured context, the process turned out to be fairly smooth and only took about two months. The AI models did much of the work and testing. The company was able to quickly realize the substantial benefits of Snowflake, both in terms of lower costs and the ability to effectively roll out AI and analytics projects.

Enhance Discoverability with MCP

Another interesting use case is how Norges Bank Investment Management (NBIM), which manages Norway's $1.8 trillion sovereign wealth fund, used structured context and MCP to significantly improve discovery.

In 2022, the CEO of the fund, Nicolai Tangen, set forth a mandate for all 670 employees to leverage AI, according to a report from Bloomberg. To ensure success, NBIM invested heavily in training and reskilling employees to support the transition. It also set up an

"AI enabler team" that helped to put on seminars, workshops, and courses.

NBIM implemented Claude, Microsoft Copilot, Perplexity, Cursor, OpenAI Deep Research, and Google AI. These tools helped with tasks like monitoring news, earnings transcripts, and financials. Employees were also able to query data in natural language.

As for MCP, this was critical for effectively managing the organization's complex data environment, as the protocol allows seamless connections with LLMs and various tools. MCP also provided the capability for authentication and other security features.

While the AI initiatives at NBIM are in the early stages, the results are standout. The fund reported a 15% increase in efficiency and annual savings of about 213,000 hours. About 300 of the staff write code with the help of AI.

Where the Ecosystem Is Headed

The successes of NBIM and the Microsoft SQL Server to Snowflake migration don't just highlight the benefits you can realize today. They also point toward the future. It's still too early to declare the dominant standards for generative and agentic AI for the enterprise because we're still in the "Wild West" phase of the technology lifecycle:

Rapid evolution of the technology
: Architectures and orchestration frameworks are still being refined. There isn't even an agreed-upon definition of "agent."

Limited production experience
: Most deployments remain pilots or narrow use cases. Until enterprises accumulate more real-world lessons, it's difficult to know which pain points actually require standardization.

Fragmented vendor landscape
: Microsoft, OpenAI, Anthropic, Google, and countless startups are pursuing divergent strategies for agent architecture and integration.

Regulatory uncertainty
: Governments are still developing AI governance frameworks. These will strongly influence which standards become mandatory and which are optional.

Market dynamics
: Right now, vendors are competing on differentiation, not interoperability. Standards tend to emerge when the competitive edge shifts from core capabilities to ecosystem effects and ease of integration.

This trajectory is typical for new enterprise technologies. Cloud computing, microservices, and mobile all went through years of fragmentation before dominant standards became entrenched. The same is likely true for the modern AI era.

But, as AI adoption grows, there are some important standards and trends, such as MCP and agentic AI, simplified natural language interfaces, and semantic layers, that are emerging as the frontrunners.

MCP and Agentic AI

MCP and agentic AI are seeing significant investment and growing adoption. For example, Walmart used MCP to help create super agents that manage four major stakeholders: associates, customers, developers, and suppliers. This made it easier to interact with the company's complex patchwork of data sources and applications.

In May 2025, Walmart announced that it was prioritizing the company's use of agentic AI:

> For retailers, the potential for agentic AI is immense. Imagine complex personal shoppers that understand nuanced preferences, dynamic store environments that adapt in real-time based on customer needs, and self-optimizing logistics networks that ensure products are always in the right place at the right time. This is why we're so invested in building the right foundation today, to not only meet but anticipate the needs of tomorrow.

The financial services tech company Block, which operates apps like Cash App and Afterpay, is another early adopter of MCP and agentic AI.

When Anthropic was developing MCP, Block provided feedback and helped define it. Block also created its own open source platform, called Goose, to leverage the protocol across its organization. Thousands of employees use it daily to develop agentic AI applications. Block has reported that this has saved employees 50% to 75% of their time spent on common tasks like querying internal data

using Snowflake, using GitHub and Jira for software development workflows, and using internal APIs for specialized use cases like compliance checks and support triage.

Simplified Natural Language Interfaces

Two months after its launch in 2022, ChatGPT reached 100 million users. That was the fastest growth ramp for any app, ever. At the time of writing, ChatGPT logs 800 million weekly active users. This is more than twice the combined numbers for Meta AI, Gemini, xAI, Claude, and Perplexity.

The ubiquity of GenAI has set expectations for users. The ChatGPT interface has become the default mental model for AI interaction, even in the enterprise. Employees expect a simple, natural language–driven experience. Companies that fail to embed this accessible interface into their workflows risk falling behind, no matter how sophisticated their backend architecture might be.

ChatGPT's approach to agentic AI is likely to become a standard as well. You merely click the plus icon on the chat interface and select Agent Mode. Then a dropdown appears with options for reports, actions, spreadsheets, and presentations.

Semantic Layer

The semantic layer has great potential for being a standard for enterprises adopting agentic AI, because it addresses the foundational problem of inconsistent business definitions. As we've seen in this report, this approach provides a single, unified source of truth for key metrics and dimensions.

A structured data interface grounded in reliable data and metadata is the key for agentic AI to move beyond simple queries and perform complex, multistep tasks safely. The semantic layer is a core component of this interface. It provides the necessary governance and context for AI agents to function effectively. This means that an agentic AI system will provide answers based on the underlying evidence.

Next Steps

Understanding these emerging standards and trends is important, but the more immediate question is "Where do I start?" The following 10 steps provide a practical guide to building the foundation discussed in this chapter and throughout the report:

1. *Build a Center of Excellence (CoE)*
   This should be a small group, say under 10 members. But it is a good idea for the CoE to span across functions like security, analytics, data engineering, and go-to-market teams.

2. *Run a readiness assessment*
   A good approach is to use a security maturity model (L0 to L4)—a framework that measures how developed and consistent an organization's security practices are—for these dimensions: semantics, discovery, policy, execution, lineage/quality, provisioning, and observability.

3. *Prioritize by impact*
   Begin with the main KPIs. They should be grounded in the common questions asked by stakeholders.

4. *Set metadata hygiene standards*
   Define and enforce clear-cut rules for tier-1 assets. These include owner, documentation, tags, sensitivity, tests, and freshness service level objectives (SLOs). You should also integrate the checks directly into CI pipelines.

5. *Deploy a semantic layer for pilots*
   This involves standing up the semantic layer in pilot domains. Then publish a discovery index ranked by owner, freshness, test pass-rate, usage, sensitivity, and lineage. After this, enable governed natural language querying in read-only mode.

6. *Wire evidence into every answer*
   You want to attach answers to definitions, lineage, freshness/tests, and policy notes.

7. *Use controlled AI agents*
   You should have PR-only agents. They can propose diffs, tests, and docs. But there must be human approval before there are CI builds in staging.

8. *Bake policy into execution paths*
   First, there should be strong identity and access controls, such as OIDC/OAuth and RBAC/ABAC. Next, there needs to be enforcement of masking and row-level security. Then default all execution to sandbox mode.

9. *Publish a quarterly trust review*
   This includes documentation, tests, and semantic coverage. The review should also track the percentage of answers backed by evidence, provide lessons from incidents, and close gaps.

10. *Train stakeholders in governed usage*
    Teach users how to frame governed questions in terms of metric names, grains, and filters. Show them how to interpret evidence panels confidently.

Takeaways

AI is reshaping the data stack at its foundation. Conversational analytics, copilots, and agentic workflows are converging on a structured context interface that puts data, semantics, lineage, and policy front and center.

When you pair this foundation with disciplined governance, AI systems move beyond demos and prototypes to impactful production applications. The result is a safer, more explainable system that can plan, execute, and improve real workflows while maintaining trust.

By taking this approach, you can unlock the scale, consistency, and reliability that truly deliver transformative benefits. It streamlines operations, accelerates innovation, and makes data a driver of growth rather than a hidden liability. This is the shift that turns AI from a promising experiment into a core engine of enterprise value.

About the Authors

Tom Taulli is a consultant to various companies, such as Aisera, SnapLogic, and TadHealth. He has written several books, including *AI-Assisted Programming: Better Planning, Coding, Testing, and Deployment* (O'Reilly). Tom has also taught IT courses for UCLA, Pluralsight, and O'Reilly. For these, he has provided lessons in using Python to create deep learning and machine learning (ML) models. He has also taught on topics such as natural language processing.

Tom Grabowski guides AI platform strategy to integrate ML and agentic capabilities into the modern data stack at dbt Labs. Previously, he spearheaded AIOps and ML products at Elastic and founded LogLogic and RapidEngines. Tom's work reflects a lifelong commitment to bridging data architecture, AI innovation, and practical enterprise adoption.

Sai Maddali is director of product management, AI and platform, at dbt Labs.