Futurum®

The Agentic Frontier: Why Converged Data Engines are the Optimal Foundation for Autonomous Enterprise AI

COMMISSIONED BY ORACLE

FEBRUARY 2026

Introduction: The End of the Polyglot Era and the Rise of the Agent

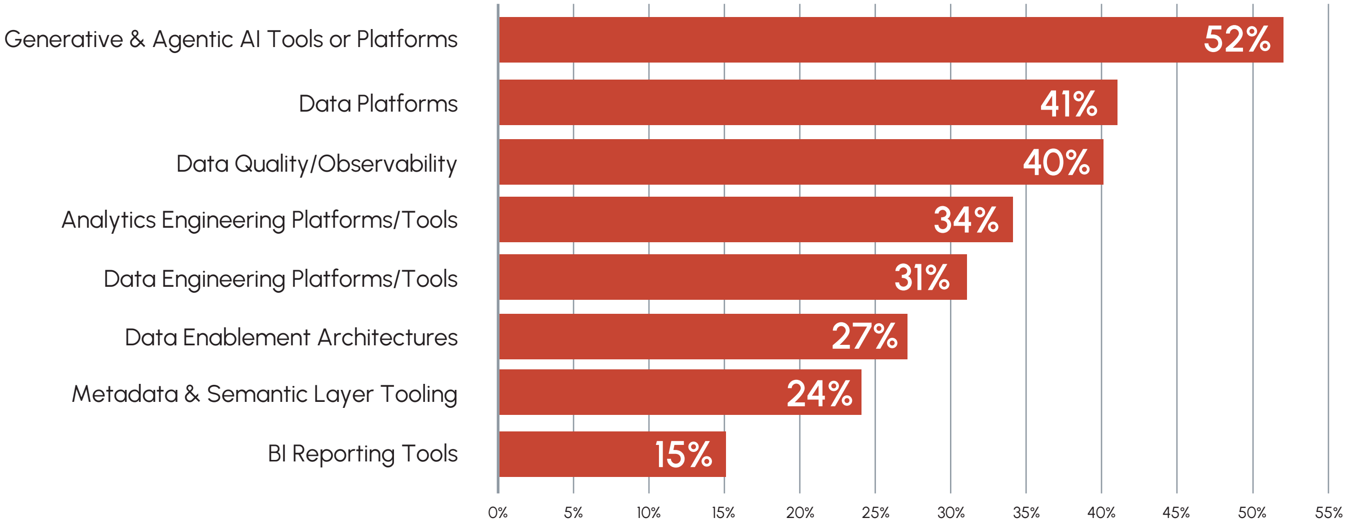
Executive Summary

The global Data Intelligence, Analytics, and Infrastructure market is rapidly accelerating and will surpass USD475 billion by the end of 2025[1], fueled almost entirely by the enterprise mandate to operationalize artificial intelligence (Figure 1). However, as organizations pivot from experimental chatbots to autonomous Agentic AI (e.g., systems capable of reasoning, planning, coordinating, and executing complex business tasks mostly without human supervision), they are colliding with a hard architectural ceiling. The polyglot persistence strategy of the last decade, where companies deployed specialized databases for every distinct data type, has created a fragmented landscape that is fundamentally hostile to the needs of modern AI.

Figure 1: Top Investment Priorities (% of Respondents)

Futurum Research data confirms that over half of enterprises have prioritized Agentic and Generative AI investments for 2025. The infrastructure chosen today must support this specific workload.

Q: Which areas were most frequently prioritized for increased investment for 2025?



Futurum Research data indicate that while most enterprises have prioritized Generative AI initiatives, the majority are struggling with static applications and data platforms rather than cognitive reasoning[2]. The friction caused by moving data between document stores, vector engines, and analytical warehouses makes for a latency bottleneck that cripples the performance of AI agents.

This report analyzes the critical architectural divergence between two leading approaches for building JSON-centric AI applications: the specialized document store model exemplified by MongoDB Atlas and the converged, multi-model engine of Oracle Autonomous AI Database. Futurum's analysis shows that while specialized stores have been widely used in the web era, companies hoping to build agentic AI at scale are now demanding a converged foundation where JSON, Relational, Vectors, and other data models co-exist in a single, high-performance data platform – exemplified by Oracle Autonomous AI Database.

[1] Futurum Research, 1H 2025 Data Intelligence, Analytics, & Infrastructure Market Sizing & Five-Year Forecast.
[2] Futurum Research, 1H 2025 Data Intelligence, Analytics, and Infrastructure Decision Maker Survey Report

The Agentic Mandate: Three Cognitive Requirements

To understand why the database market is shifting, it is essential to understand the evolving nature of enterprise applications. Traditional apps simply retrieved and processed data. But AI agents must understand that data. This shift imposes three non-negotiable requirements on the data layer:

- **Multi-Model Fluency:** Agents do not think in silos. They need to read unstructured user profiles (JSON), cross-reference them with rigid inventory data (Relational), and perform semantic similarity searches (Vectors) simultaneously.
- **Zero-Latency Context:** Retrieval-Augmented Generation (RAG) is the lifeblood of agentic accuracy. If the database requires an ETL process to move data from a document store to a vector engine, the agent is thinking with outdated memories. Context must be real-time.
- **Transactional Agency:** The "agent" in agentic AI implies action. When an AI is authorized to book a flight, transfer funds, or update a medical record, "eventual consistency" is unacceptable. The database must guarantee ACID integrity with the same rigor as a core banking system.

Consider the cognitive load of a human employee. If a customer service rep had to check a filing cabinet for a contract, log into a mainframe for billing history, and open a separate spreadsheet for current inventory, the customer would hang up before the rep had an answer.

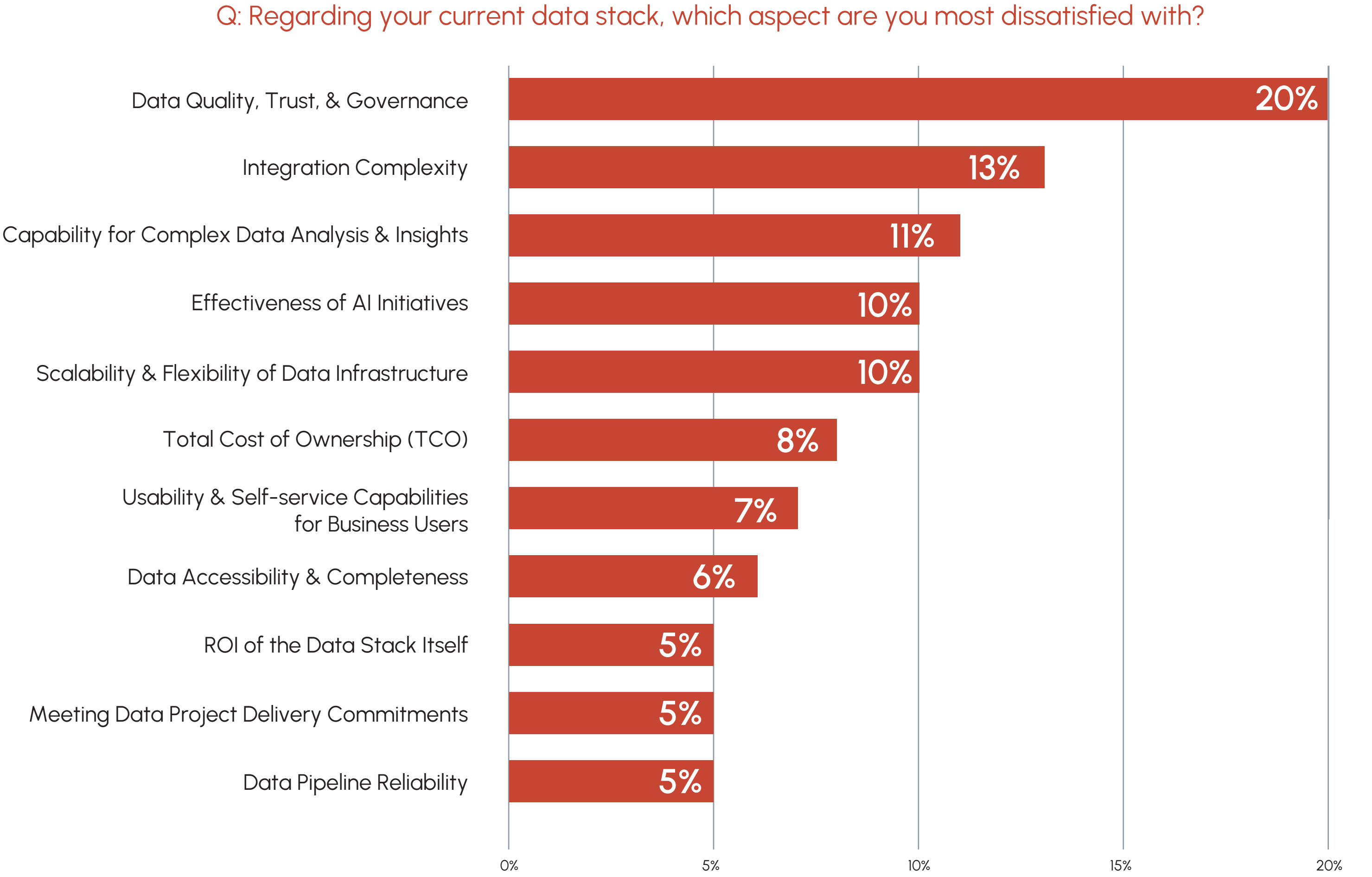
Specialized databases create this exact scenario for AI.

When data types are siloed, the AI agent spends its compute cycles acting as integration glue (e.g., fetching, formatting, and joining data across network hops) rather than reasoning. A converged engine removes this "cognitive tax," presenting the agent with a unified view of the enterprise reality.

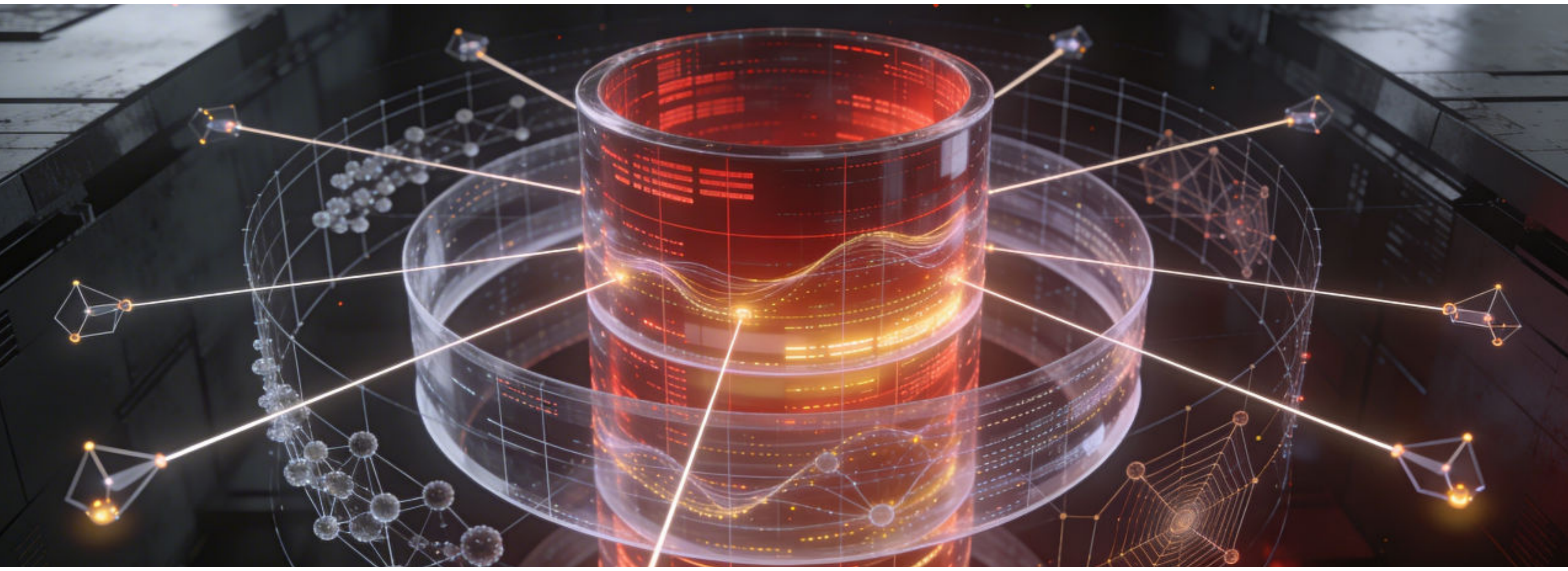
The Challenge: Enterprise builders are discovering that stitching together several best-of-breed databases, such as a document database (such as MongoDB), along with separate vector and analytical tools, creates a complicated environment (Figure 2) that is expensive to maintain and risky to rely on for a mission-critical enterprise application strategy.

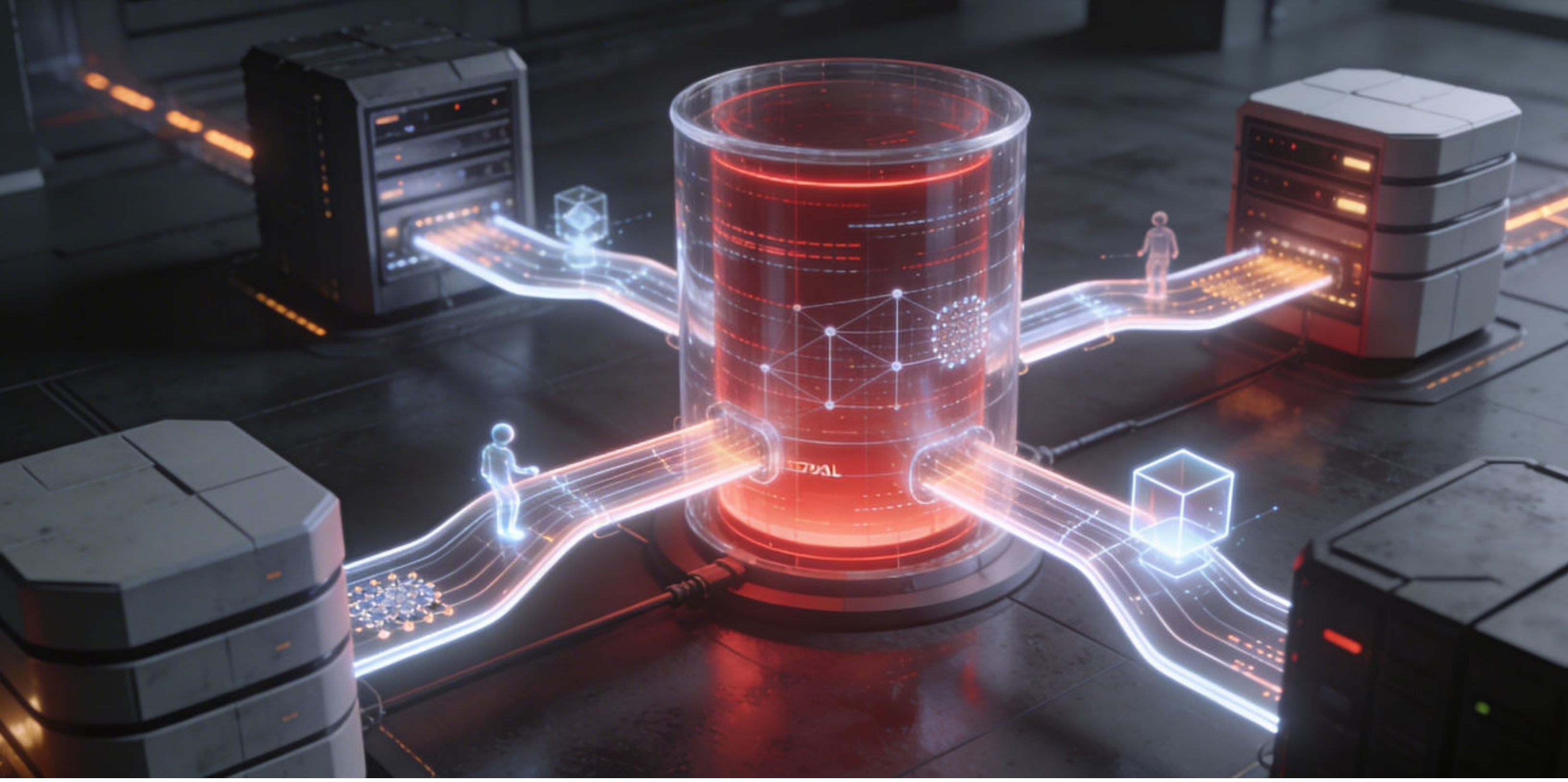
Figure 2: The High Cost of Fragmentation

Integration complexity and governance gaps are the top frustrations for IT decision makers. A converged engine directly addresses the 33% of dissatisfaction stemming from these siloed architecture issues.



Source: Futurum Research, July 2025





Oracle's Converged Data Advantage: Unifying Structure and Flexibility

The defining battleground for the next generation of applications is not about how fast you can store a document, but how effectively you can reason about it. MongoDB built its reputation on the flexibility of the JSON document model, allowing developers to iterate fast without schema constraints. However, data does not live in a vacuum. In the enterprise, JSON documents are almost always related to other data entities, and this is where the specialized document model begins to fracture.

A converged engine removes this "cognitive tax," presenting the agent with a unified view of the enterprise reality.

The Oracle Approach: JSON Relational Duality Views

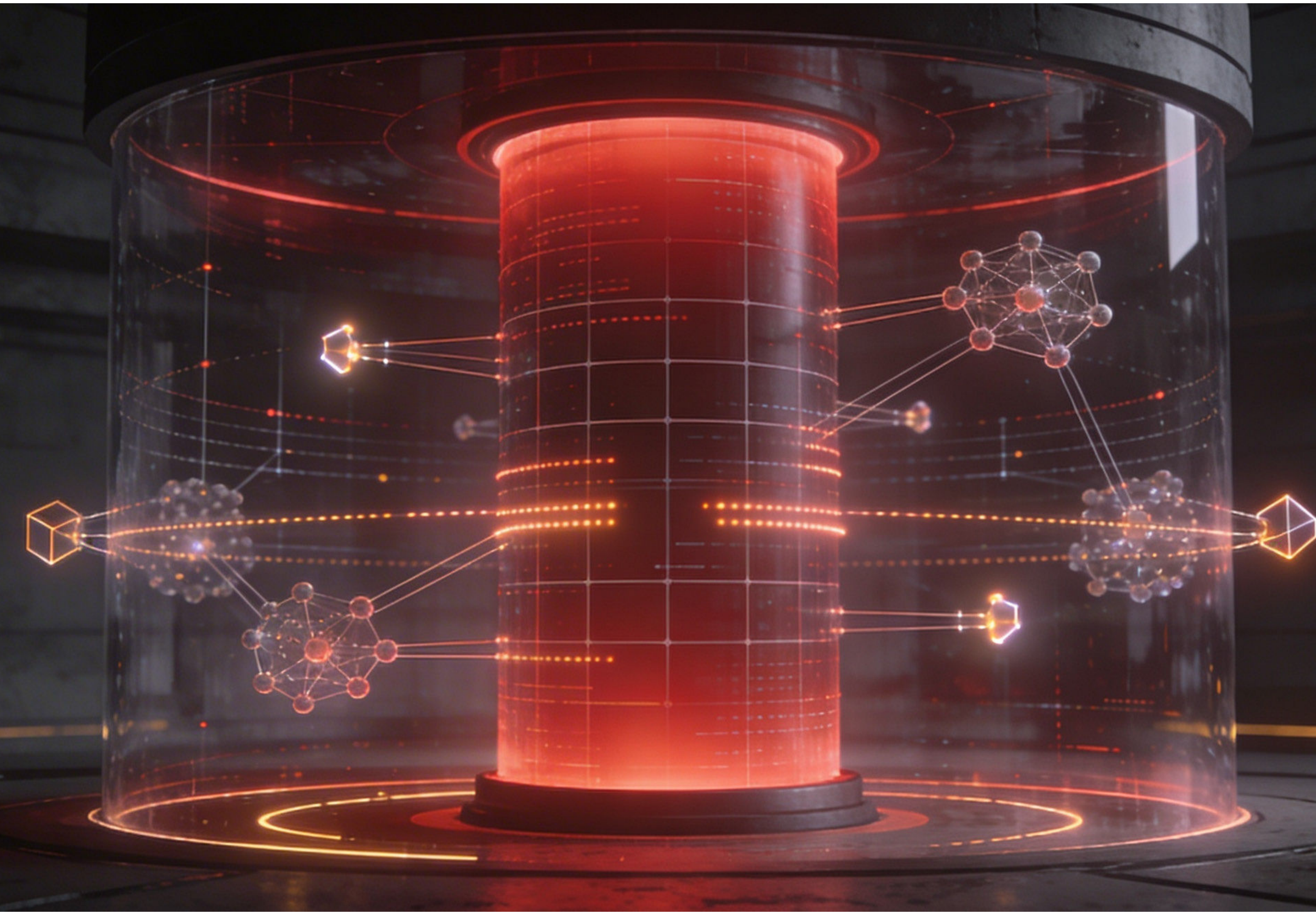
Oracle AI Database introduces a paradigm shift known as JSON Relational Duality Views. This architecture acknowledges a fundamental truth: developers love JSON for its flexibility, but data analysts and business logic require the structure of relational tables.

Instead of forcing a choice, Oracle's engine allows the exact same byte of data to be accessed as a JSON document or a relational table (or as a graph) simultaneously, without duplication. Both APIs can read and write in their native form. This is not a synchronization trick. It is a fundamental capability of the converged data engine. An AI agent can ingest a complex, nested JSON purchase order, and moments later, a fraud detection algorithm can query the line items of that order using standard SQL joins against historical patterns – all with full ACID consistency and without any data duplication or movement.

For twenty years, the industry has battled the "Object-Relational Impedance Mismatch," which is the friction between how developers write code (using objects/JSON) and how data is stored (in tables). Historically, companies have used complex Object-Relational Mapping (ORM) tools or surrendered ACID compliance to use NoSQL document databases such as MongoDB.

Oracle's approach effectively declares a truce. With Duality Views, data is stored once in a relational table that maintains full consistency across SQL, REST, and JSON APIs. For instance, when an application uses JSON to update data, all of the fields – which may span several different JSON collections – are updated at the same time in the relational tables so that database integrity is maintained across these collections. When the application gets data using JSON, Duality Views directly accesses the underlying data without any translation delays. Furthermore, when SQL is used to update rows in the relational tables, the data across thousands or millions of documents reflects that change instantly. Like a NoSQL Database, Oracle JSON Duality supports schema flexible documents. Duality even has built-in optimistic concurrency control – preventing concurrent updates from overwriting each other's changes, but without resorting to intrusive locking.

This means the AI developer can iterate at the speed of NoSQL, while the CFO gets the reporting rigor of SQL with the data-level security and governance provided by Oracle Autonomous AI Database – all from the same single source of truth.



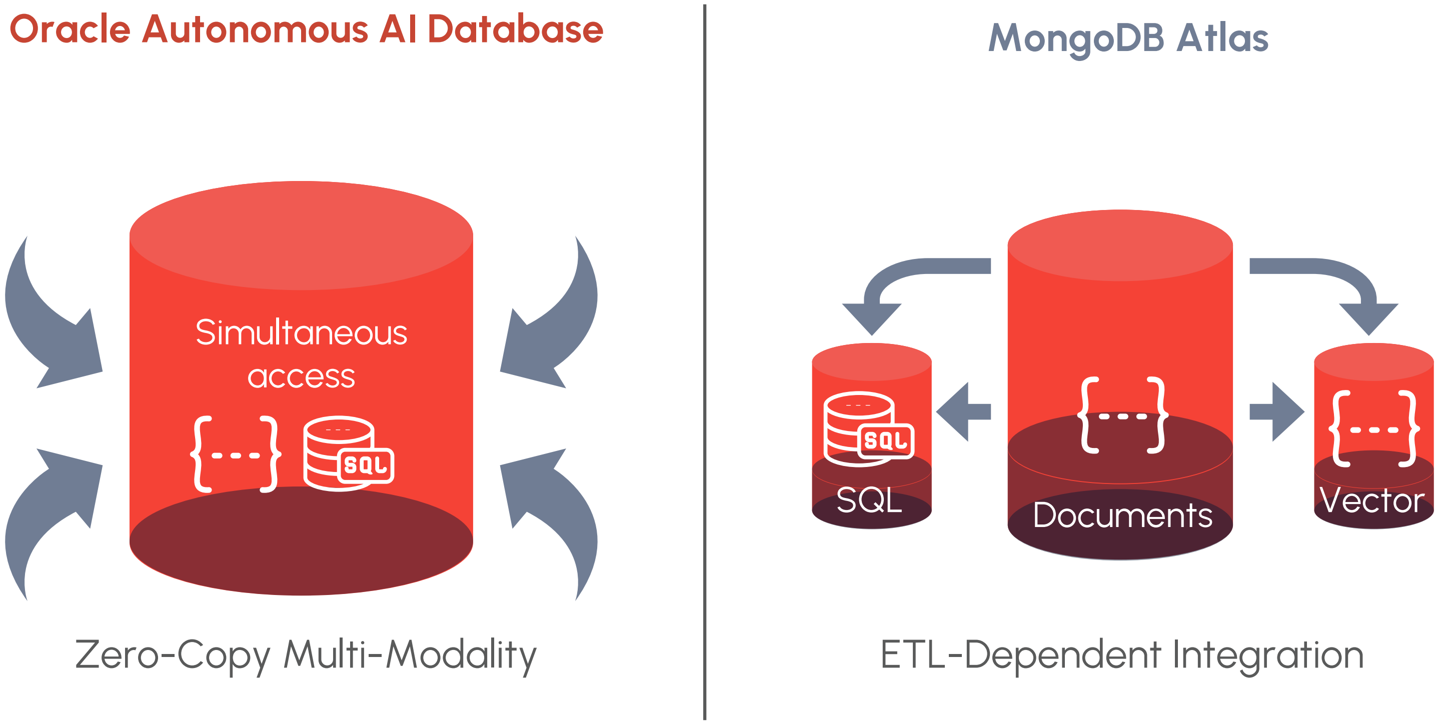
The MongoDB Approach: The Document-Only Limitation

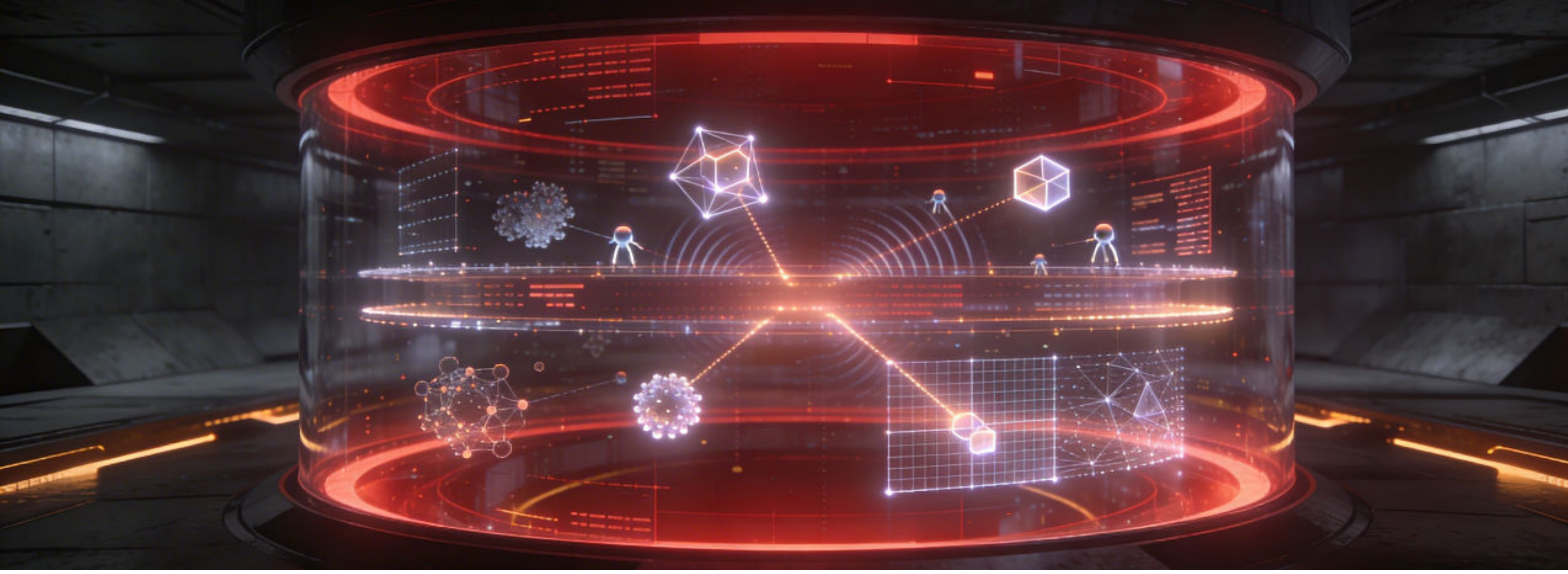
Conversely, MongoDB remains rooted in its identity as a document store: Unlike Oracle JSON Relational Duality Views, where different documents can share data efficiently and consistently, MongoDB needs to either duplicate data across documents (allowing inconsistencies and slowing down updates) or data has to be shredded into different documents and reassembled, at a cost to performance, during query time. This adds unnecessary complexity for application developers. While MongoDB has added some features to support analytics, it fundamentally treats data as documents. To perform complex analytics or joins against structured business data, developers may have to resort to complex and CPU-intensive aggregation pipelines or, more commonly, export the data to a separate data warehouse.

For the Agentic AI builder, this creates a reasoning gap. If the AI needs to answer a question that requires combining a user's unstructured session logs (JSON) with the company's structured pricing rules (SQL), a MongoDB-centric architecture may necessitate transferring data across the network. This introduces latency, complexity, and governance issues.

Oracle's converged approach in Autonomous AI Database provides a superior solution, eliminating this friction, allowing the agent to think across data modalities in a single database round-trip supported by a single optimizer that understands the different data models.

Figure 3: Comparative Analysis of Data Modality Support





The Compatibility Bridge: Modernizing Without Rewriting

One of the most pervasive myths in the industry is that moving from a document store to a converged engine requires a total application rewrite. This fear keeps many enterprises locked into legacy NoSQL infrastructure. However, the Oracle AI Database API for MongoDB fundamentally alters this equation.

By emulating the MongoDB wire protocol, Oracle allows existing applications to communicate with the Autonomous AI Database as if it were a MongoDB endpoint. Developers can continue using their preferred tools, drivers, and frameworks, while the data they generate is transparently stored, indexed, and secured as relational tables by Oracle's converged engine.

This is a platform upgrade, not a rip-and-replace scenario. This essentially decouples the application layer from the storage engine. While deep edge-case optimizations may require tuning, the Oracle Database API for MongoDB allows the vast majority of applications to run transparently, effectively allowing IT leaders to swap out the engine while the car is moving.

Oracle's Native Vector Processing: The Engine of Real-Time RAG

RAG has become the standard architecture for grounding AI models in enterprise truth. However, the effectiveness of RAG is entirely dependent on the freshness of the available data.

Oracle's Embedded Intelligence

Oracle has integrated vector processing directly into the core of the Autonomous AI Database. This means that vector embeddings (e.g., the mathematical representations of semantic meaning) are stored, indexed, and queried right alongside the operational JSON documents and relational transactions.

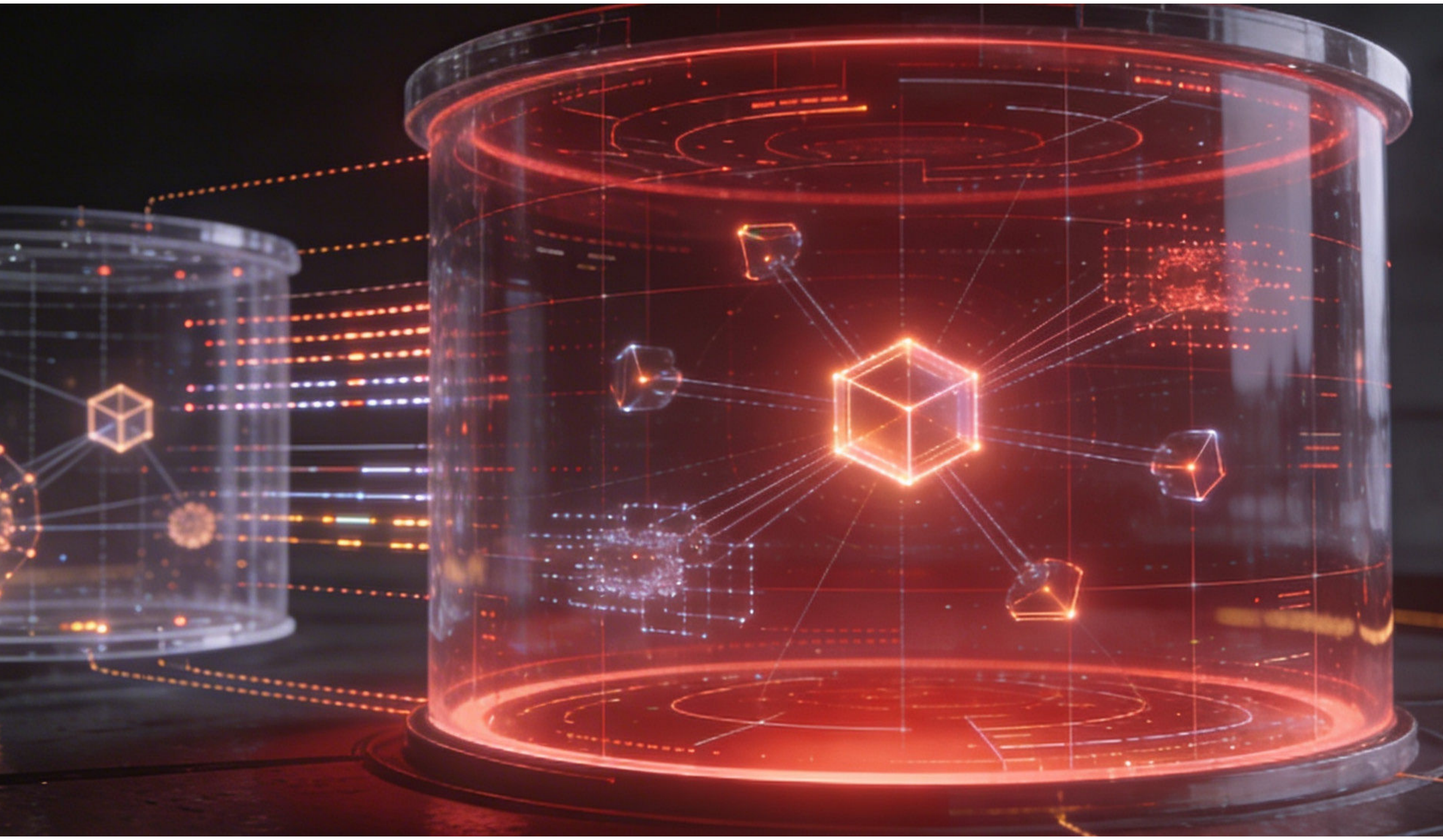
This architecture enables near real-time RAG. When a user updates their profile or a new product is added to the catalog, the vector index is updated immediately within the same transaction. The AI agent always has access to the most current state of the world. Furthermore, because the vectors live with the data, the database can execute hybrid queries in a single, highly optimized engine pass. For example, an agent may ask, "Find me products similar to this image (Vector) that are in stock (SQL) and have a 4-star rating (JSON)."

The Latency of Bolted-On Search

MongoDB Atlas offers vector search capabilities via integration with a separate vector search engine, which can introduce synchronization lags. For high-velocity applications, latency not only makes interactions with web pages slower, but worse; latency between a data update and its availability in the vector index can lead to AI hallucinations or semantic drift, where transactions appear valid but are not fundamentally true. If an agent recommends a product that went out of stock five seconds ago because the vector index hadn't caught up to the operational store, the customer experience is broken.

This is the architectural gap of the sidecar index. Because MongoDB Atlas synchronizes data from the operational store to the vector search engine (based on Lucene), there is an inherent eventual consistency window. In high-velocity trading or inventory systems, this gap creates state-vector dissonance, where the AI's vector memory is slightly out of sync with the database's operational reality.

In contrast, Oracle's in-kernel execution eliminates this gap entirely, ensuring that the AI's reasoning is always grounded in the immediate present, rather than the near past. This approach avoids hallucinations and leads to more accurate and contextually relevant answers.



Scaling Reasoning Capability: RAC vs. Sharding

As AI applications move from pilot to production, they face the brutal reality of scale. AI workloads are incredibly I/O-intensive, often requiring massive scans of data to generate context. How the database scales to meet this demand becomes a critical differentiator.

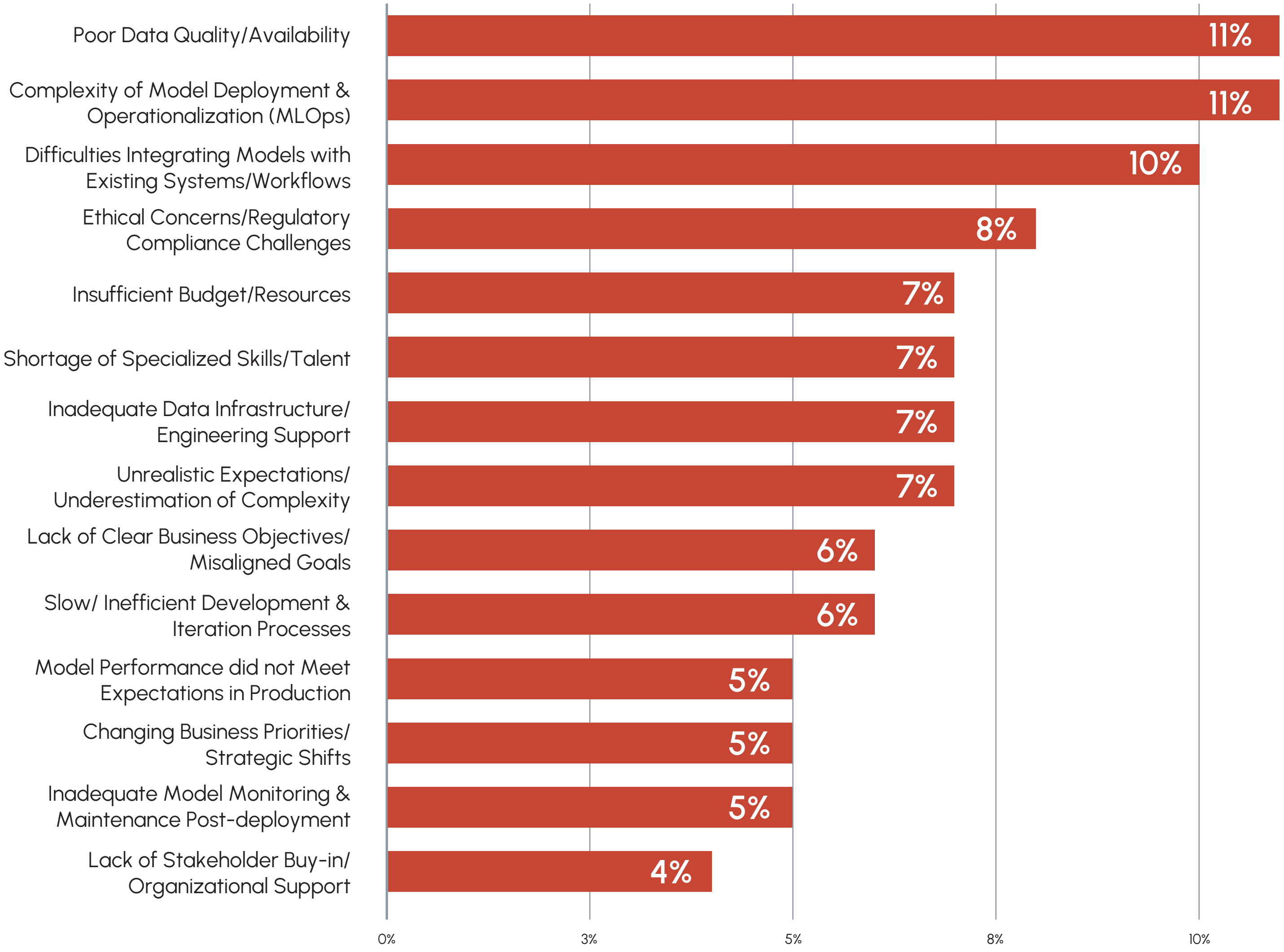
Avoiding Sharding Complexities

MongoDB's primary mechanism for scaling is sharding, which is splitting data across multiple servers based on a shard key. While effective for simple read/write workloads, MongoDB's approach to sharding can introduce significant complexity for the types of global, complex queries used by AI agents (Figure 4).

Figure 4: Why AI Projects Stall

Complexity in MLOps and poor data quality are the primary killers of AI initiatives. Consolidating the data estate on a converged engine simplifies the operational lifecycle, directly mitigating these failure risks.

Q What is the top data-specific factor contributing to the failure of AI projects in your organization?



Source: Futurum Research, July 2025

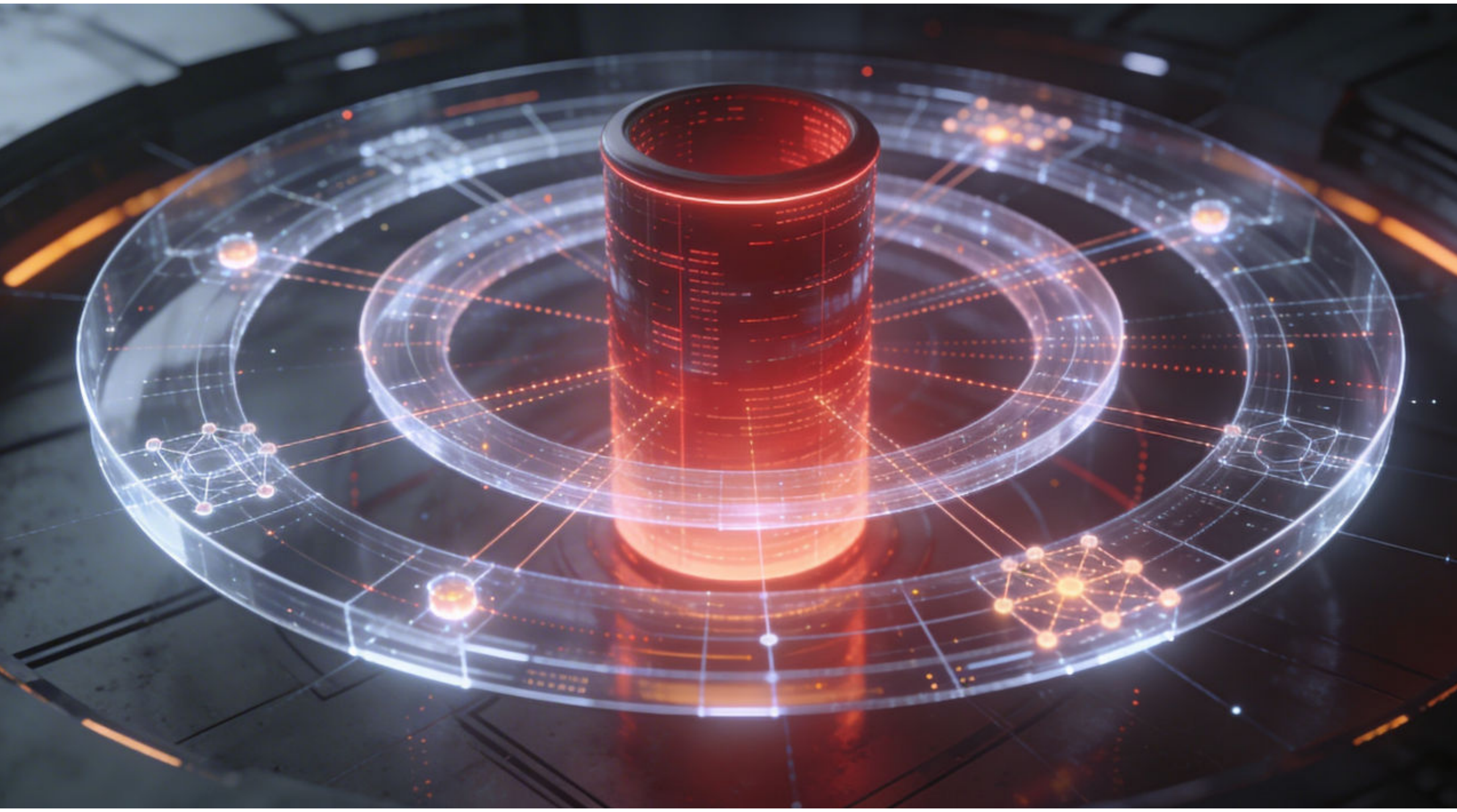
If an agent needs to query a dataset that spans multiple shards, the database must perform a "scatter-gather" operation, querying each shard individually and reassembling the results. This adds latency and computational overhead. Furthermore, managing a sharded cluster requires the enterprise builder to make upfront decisions about data distribution that are difficult to change later. It forces application logic to be aware of the physical infrastructure.

The hidden cost of sharding rarely appears in the proof-of-concept phase. Instead, it arrives on Day 2 of operations. When a specific shard becomes a hot spot due to uneven data distribution, the engineering team is forced to rebalance the cluster manually. This makes for a high-risk surgery often performed on live production systems. It forces application developers to become infrastructure architects, writing code that is uncomfortably aware of physical server topology. By forcing the application to route queries based on shard keys, you are hard-coding infrastructure limitations directly into your AI logic, creating technical debt that is painful to unwind.

Oracle's Transparent, Automatic Scaling

Oracle delivers a fundamentally different, simpler, and easier-to-use scaling experience than MongoDB. Public cloud and on-premises users see data as a single logical pool for their applications to access with Oracle Autonomous AI Database automating the scaling process. As the workload grows, Autonomous AI Database can transparently use more CPUs and compute nodes, and database storage is automatically distributed across multiple intelligent storage servers. All of this is done without requiring data to be logically partitioned or sharded – tremendously simplifying administration.

For an AI agent, this is transformative. The agent sees a single system image, which is essentially one giant pool of data. It doesn't care which server holds the data; it just queries the logical model. Autonomous AI Database's architectural transparency allows enterprise builders to scale their AI ambitions without having to re-architect their data model or manage any sort of "sharding sprawl" that can plague large NoSQL deployments.



The Visibility Trap: Managing Shadow Sprawl

Beyond the technical limitations of sharding, an operational crisis is brewing in many large enterprises: shadow IT. Because document stores are so easy for individual developers to spin up on a laptop or in the cloud with a credit card, many organizations have lost track of their data estate. They end up with hundreds of fractured, unmanaged databases, some sharded, some not, some compliant, some exposed.

The first step in an Agentic AI strategy is often simply finding the data. Autonomous AI Database's converged approach offers a remedy to this sprawl by centralizing these rogue JSON collections into a unified, multi-tenant environment. Instead of managing five hundred distinct database endpoints, the enterprise manages a single logical fleet of databases. This consolidation can make a huge difference in ensuring that the AI agent is reasoning over the whole truth, rather than a fragmented slice of it.

The Hidden Risk of Shadow IT

In the era of Agentic AI, you cannot govern what you cannot see. Many enterprises are plagued by"Shadow Mongo" – hundreds of unmanaged, fragmented database instances spun up by developers outside of central IT control. These silos fracture the enterprise truth, leaving AI agents to reason over incomplete data. In contrast, a unified, multi-tenant Oracle environment that avoids silos is more than infrastructure optimization. It serves as a prerequisite for safe, accurate AI.

Enterprise-Grade Transactional Integrity for Agentic Action

We are entering the era of software autonomy, where AI is authorized to take action, such as booking flights or managing capital. This shifts the risk profile of the database from "informational" to "financial."

The Hallucination vs. Corruption Trade-off

If a chatbot hallucinates a poem, it is amusing. If an autonomous supply chain agent corrupts an inventory transaction, it is a business disaster. Over the years, MongoDB has made strides in ACID compliance, but it is historically rooted in an "eventual consistency" mindset. In complex failure scenarios such as a network partition during a multi-document transaction, the burden of recovery often shifts to the application developer.

It is essential to distinguish between a chatty AI chatbot and a transactional agent. A chatty AI summarizes emails; if it misses one, it's an annoyance. A transactional agent manages supply chains or moves capital. While MongoDB added ACID transactions in 2018, they remain a heavy lift for the engine. Enabling multi-document transactions in a document store imposes significant locking overhead and performance penalties that can choke high-throughput AI agents. In MongoDB, ACID is a feature you turn on.

While MongoDB added ACID transactions in 2018, they remain a heavy lift for the engine.

Conversely, with Oracle JSON Relational Duality Views, ACID is the foundation that the database is built on. For an autonomous agent executing thousands of financial decisions per second, you cannot afford a database that forces you to choose between ultimate speed and basic safety.

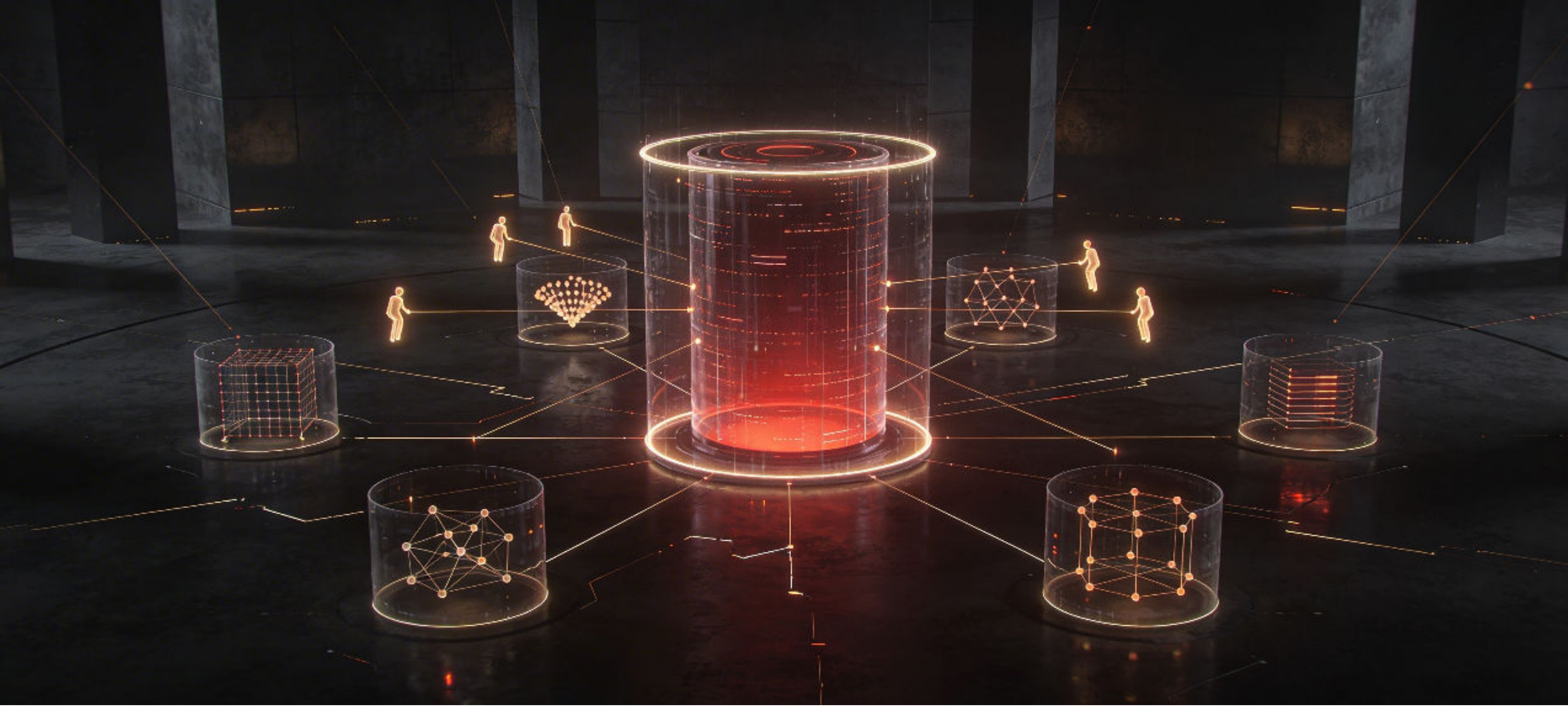
Oracle: The Gold Standard of ACID

In comparison, Oracle's consistency model serves as a guardrail that keeps autonomous speed from turning into autonomous chaos. Oracle Autonomous AI Database inherits the transaction engine that runs the world's banking and telecommunications systems. It offers stock exchange- level, mission-critical/class 1 availability and integrity. Features such as Active Data Guard help ensure that even in the event of a catastrophic regional failure, no committed transaction is lost.

For builders designing agents that move money, update healthcare records, or manage critical infrastructure, this level of integrity is not optional. Oracle provides the safety net that allows organizations to trust their agents with write access, effectively graduating them from "AI interns" to fully trusted employees.

Table 1: Summary of Competitive Advantages

Area of Emphasis	MongoDB: Specialized Document Store	Oracle: Converged AI Engine
Data Modality	Siloed & Document-Centric. Treats data primarily as JSON. Analyzing relationships requires complex application logic or extracting data to a separate warehouse.	Unified & Multi-Modal. Supports JSON, Relational, and Vectors in a single engine. JSON Relational Duality Views allow the same data to be accessed as documents or tables without duplication.
AI Integration	Bolted-On Vectors. Vector search is often a distinct integration. Introduces state-vector dissonance, a latency gap between data updates and index availability that causes hallucinations.	Native Vector Processing. Vectors are stored and indexed alongside document and operational data. Updates are transactional and immediate, providing low-latency RAG and grounded reasoning.
Scalability	Sharding Complexity. Scaling requires physically partitioning data (sharding). This forces developers to manage shard keys and rebalance clusters, creating long-term operational debt.	Scale-Out Simplicity. Real Application Clusters (RAC) provide a single system image. Performance scales transparently by adding compute nodes dynamically without changing the application logic or data model.
Reliability	Eventual Consistency. While ACID is possible, the architecture defaults to performance over strict consistency. Developers have to fully understand all of the choices and tradeoffs and then configure and code to address them. Minor misunderstandings can lead to inconsistencies that are difficult to find and debug.	Absolute ACID Integrity. Built on the engine that runs the global financial system. Oracle AI Database delivers transaction safety and data accuracy, a non-negotiable requirement for agentic action. Developers just create the app, and the database handles consistency.
Consolidation	Vendor Lock-In. The special-purpose nature of MongoDB makes it impossible to move all workloads to it. Integrating multiple models requires a polyglot environment, significant application coding, and time-consuming refactoring.	Plug-and-Play Compatibility. Oracle Autonomous AI Database's converged architecture enables massive consolidation of AI, transaction processing, analytics, graph, spatial, and document workloads. The Oracle Database API for MongoDB emulates the MongoDB wire protocol, allowing existing apps to run on Oracle with little to no code changes.



Strategic Resource Allocation: Minimizing the Data Tax

Finally, the economics of AI cannot be ignored. There is an ever-present data tax at the heart of all enterprise application deployment. This is the cost of moving, copying, and transforming data. And it is a silent killer of AI ROI.

The Polyglot Penalty

A typical MongoDB-based AI stack encompasses more than just MongoDB. For example, it includes MongoDB for documents, Pinecone for vectors, Snowflake for analytics, and Apache Kafka to glue them all together. Each of these hops incurs licensing and egress fees, latency costs, security risk, and management overhead.

Oracle's converged architecture offers a potentially massive cost advantage through consolidation. By running Relational, JSON, Vectors, and other data models in the same, integrated, high-performance data platform, Oracle eliminates the network costs of data movement. There is one bill, one security model, one availability paradigm, and one support contract.

Beyond the raw infrastructure bill, the polyglot approach incurs a massive governance tax. When your AI stack consists of four different specialized databases, you have four different security models to configure, four different backup strategies to audit, and four different compliance risks to manage. If you secure the document store but leave the vector database open, your RAG pipeline becomes a data leak.

If you secure the document store but leave the vector database open, your RAG pipeline becomes a data leak.

Oracle's converged approach applies a single security policy across JSON, Relational, and Vectors at the data layer. You define access controls once, and they persist regardless of how the data is accessed. In a regulatory environment that is increasingly scrutinizing AI, this unified governance posture is a competitive advantage.

Predictability in Consumption

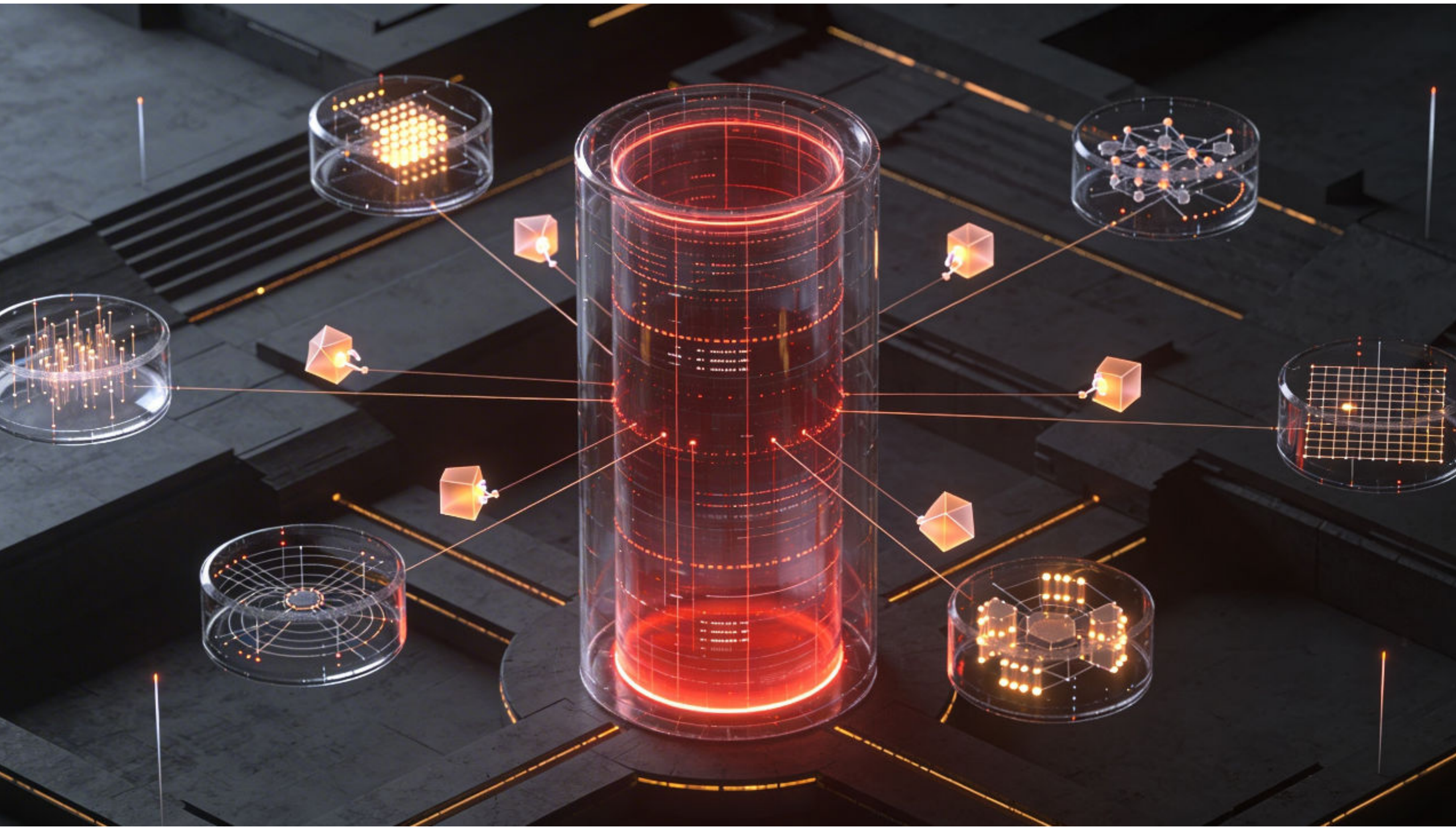
AI agents are "chatty" by their very nature. They read and re-read context logs constantly. In the consumption-based pricing models standard to cloud-native document stores, this high-velocity I/O can lead to unpredictable, runaway bills.

Oracle's Exadata infrastructure, which powers Autonomous AI Database, uses Smart Scan technology to offload query processing to the storage layer, drastically reducing the I/O that the compute nodes (and the customer's bill) have to absorb. This provides a more predictable cost profile for the read-intensive loops that define agentic workloads. Exadata is available in OCI, AWS, Azure, and Google Cloud, as a hybrid Cloud@Customer offering, and as an on-premises system, providing flexibility for organizations and their deployment selections.

The End of the Free Lunch: The Economic Shift

For the last decade, the NoSQL market was defined by growth-at-all-costs, resulting in low prices and generous free tiers. However, market analysis suggests a pivot. As specialized vendors face pressure to become profitable, we are seeing a trend of rising costs, deprecated features, and a shift from developer delight to margin extraction.

Enterprises relying on specialized stores for their AI foundation face a new kind of constraint. As these platforms strive for efficiency, the cost of running high-performance, vector-intensive AI workloads on them will increasingly become more prohibitive. In contrast, Oracle's converged model benefits from the economies of scale inherent in a massive, multi-purpose engine. By consolidating workloads on Autonomous AI Database, enterprises insulate themselves from the pricing volatility of niche vendors that are scrambling to pivot their business models.



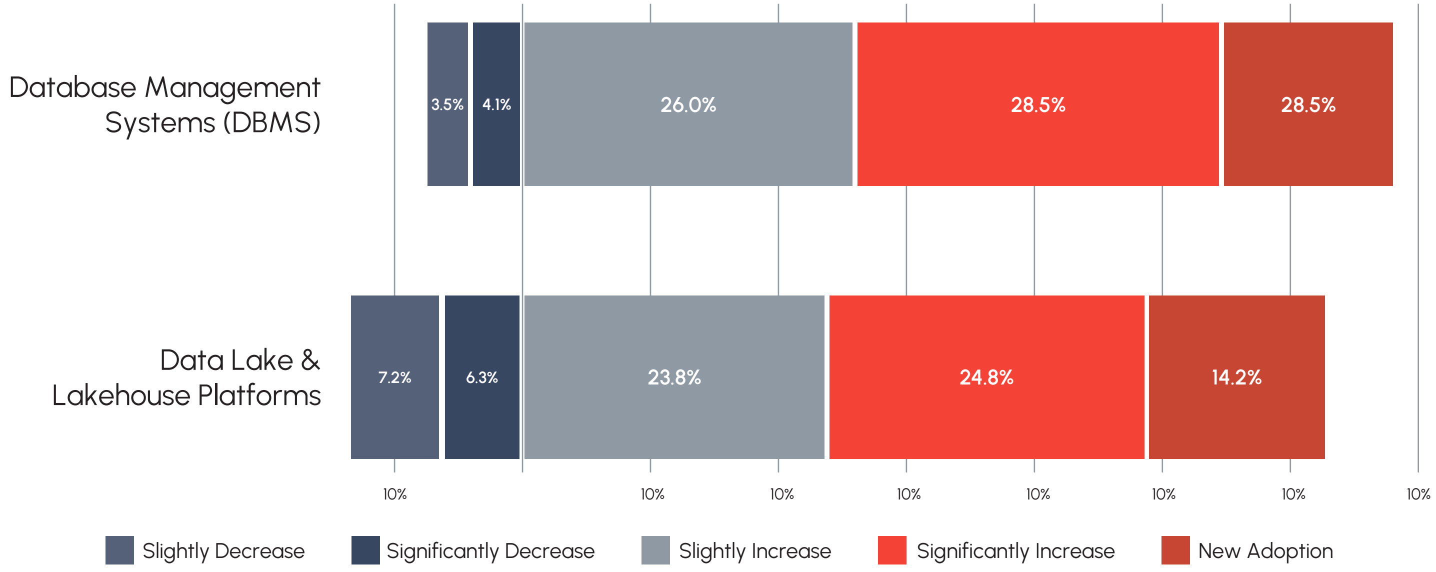
Market Leadership: The Right Choice for the Enterprise

The market data is clear: enterprises are consolidating. According to Futurum Research, Oracle maintained its dominant position in the global database management and analytics market in 2024, with revenues reaching USD24.7 billion[3]. A flight to quality drives this leadership as organizations realize that the "move fast and break things" ethos of the early NoSQL era is incompatible with the "move fast and be accurate" requirements of the AI era (Figure 5).

Figure 5: The Flight to Quality

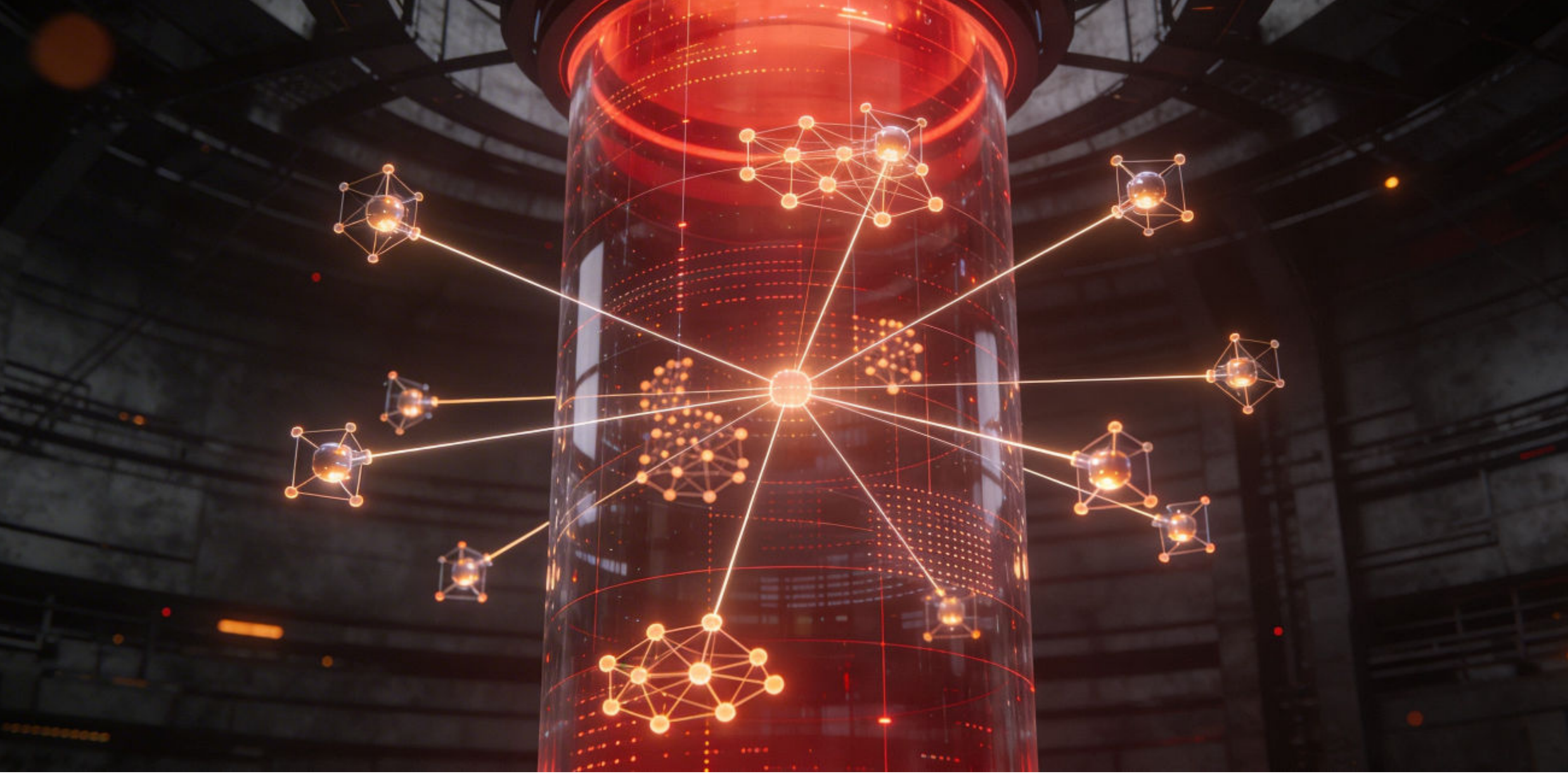
Despite the noise from niche, specialized database players, enterprise data practitioners show a decisive preference for established, enterprise-grade data platforms[4]. We are seeing a pivot from "collect everything" to "curate what matters." Organizations looking to support mission-critical, high-scale AI workloads aren't focused solely on unstructured data lakehouses. They are prioritizing database management systems to help ensure that the data feeding their applications and AI models is actually accurate.

Q: How will your technology spend on each of the following data management and analytics product areas change over the next twenty-four months?



Source: Futurum Research, July 2025

[3] Futurum Research, 1H 2025 Data Intelligence, Analytics, & Infrastructure Market Sizing & Five-Year Forecast
[4] Futurum Research, 1H 2025 Data Intelligence, Analytics, and Infrastructure Decision Maker Survey Report



Conclusion: The Foundation for Autonomous AI Innovation

For simple greenfield projects where the primary goal is rapid prototyping of document applications, MongoDB remains a strong and capable contender. Its developer ergonomics are excellent, and its community is vast.

However, for Enterprise AI Builders aiming to construct robust, agentic systems, the calculus changes. These systems require in-depth reasoning over diverse data types, absolute transaction safety, and an economic model that offsets the high cost of AI computation. In this arena, Oracle Autonomous AI Database offers a superior platform. For organizations that want a JSON-only solution, Oracle offers the Oracle Autonomous AI JSON Database at a very competitive price. This is an attractive alternative for JSON developers or organizations that want high scalability, performance, and reliability for their JSON workloads. If required, a later upgrade to a full-featured Autonomous AI Database is available with just a few clicks in the console, takes just a few seconds, and requires no changes to applications, database, or storage.

By converging JSON flexibility with relational power and native vector intelligence, Oracle enables enterprises to stop building low business value data pipelines and start building high business value autonomous agents. The future of AI is not about storing data. Instead, it is about acting on data and executing outcomes. And for that, you need a database that can do both – Oracle Autonomous AI Database.

Important Information About This Report

AUTHORS

Data Intelligence, Analytics,
& Infrastructure Practice Area,
Led by Brad Shimmin

PUBLISHER

Futurum Research

INQUIRIES

Contact us if you would like to discuss this report and The Futurum Group will respond promptly.

CITATIONS

This paper can be cited by accredited press and analysts, but must be cited in context, displaying author's name, author's title, and "The Futurum Group." Non-press and non-analysts must receive prior written permission by The Futurum Group for any citations.

LICENSING

DISCLOSURES

The Futurum Group provides research, analysis, advising, and consulting to many high-tech companies, including those mentioned in this paper. No employees at the firm hold any equity positions with any companies cited in this document.

This Competitive Assessment report was commissioned by Oracle

Futurum®

ABOUT THE FUTURUM GROUP

The Futurum Group is an independent research, analysis, and advisory firm, focused on digital innovation and market-disrupting technologies and trends. Every day our analysts, researchers, and advisors help business leaders from around the world anticipate tectonic shifts in their industries and leverage disruptive innovation to either gain or maintain a competitive advantage in their markets.

Futurum®

CONTACT INFORMATION: The Futurum Group LLC I futurumgroup.com I (833) 722-5337