



**The Foundation for Innovation:
Why Architectural Integrity and
Distributed Databases are Crucial
for Scaling Mission-Critical,
AI-Ready Applications**



Introduction: The AI Mandate on a Fractured Foundation

Executive Summary

The Data Intelligence market is surging toward US \$475 billion, driven by a singular corporate mandate: Artificial Intelligence¹. Yet, for the 52% of enterprises prioritizing Generative AI this year², a harsh reality is setting in: AI is not a magic overlay. Instead, it is a heavy workload that exposes every crack in your data foundation.

Futurum research confirms that while ambition is high, the infrastructure is brittle. The primary causes of AI project failure—poor data quality, governance gaps, and integration complexity—are symptoms of a database layer that cannot keep up with modern demands³. Enterprises are discovering that they cannot build 2026's applications on 2015's architectural compromises.

This paper contrasts two diverging paths to implementing distributed databases for the mission-critical enterprise: the "SQL-on-KV" approach of databases like CockroachDB, and the natively relational, converged architecture of Oracle Globally Distributed AI Database. We argue that for organizations requiring true architectural integrity, the choice of database engine is the difference between scaling success to meet AI application workload demands and operational failure.

¹ Futurum Research, 1H 2025 Data Intelligence, Analytics, & Infrastructure Market Sizing & Five-Year Forecast

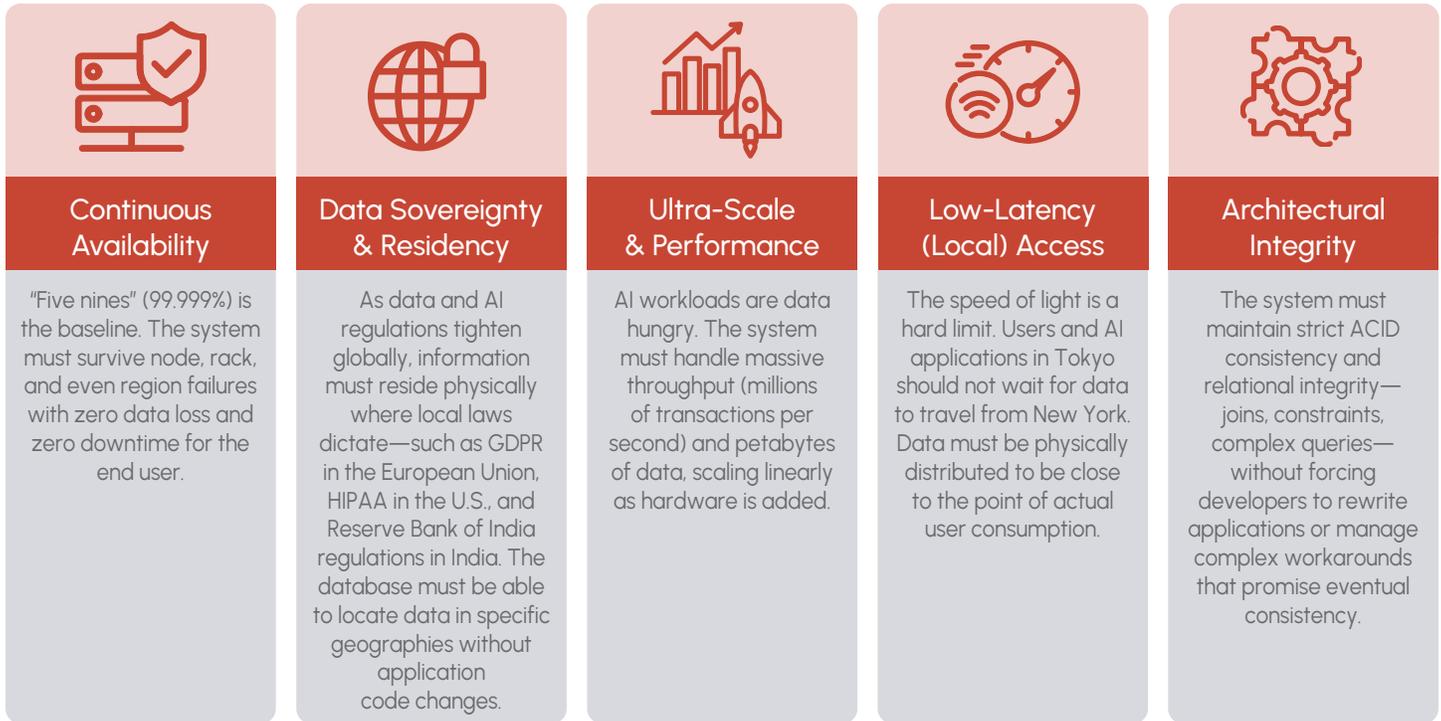
² Futurum Research, 1H 2025 Data Intelligence, Analytics, and Infrastructure Decision Maker Survey Report

³ Futurum Research, 1H 2025 Data Intelligence, Analytics, and Infrastructure Decision Maker Survey Report

The Distributed Mandate: Five Requirements for the Modern Era

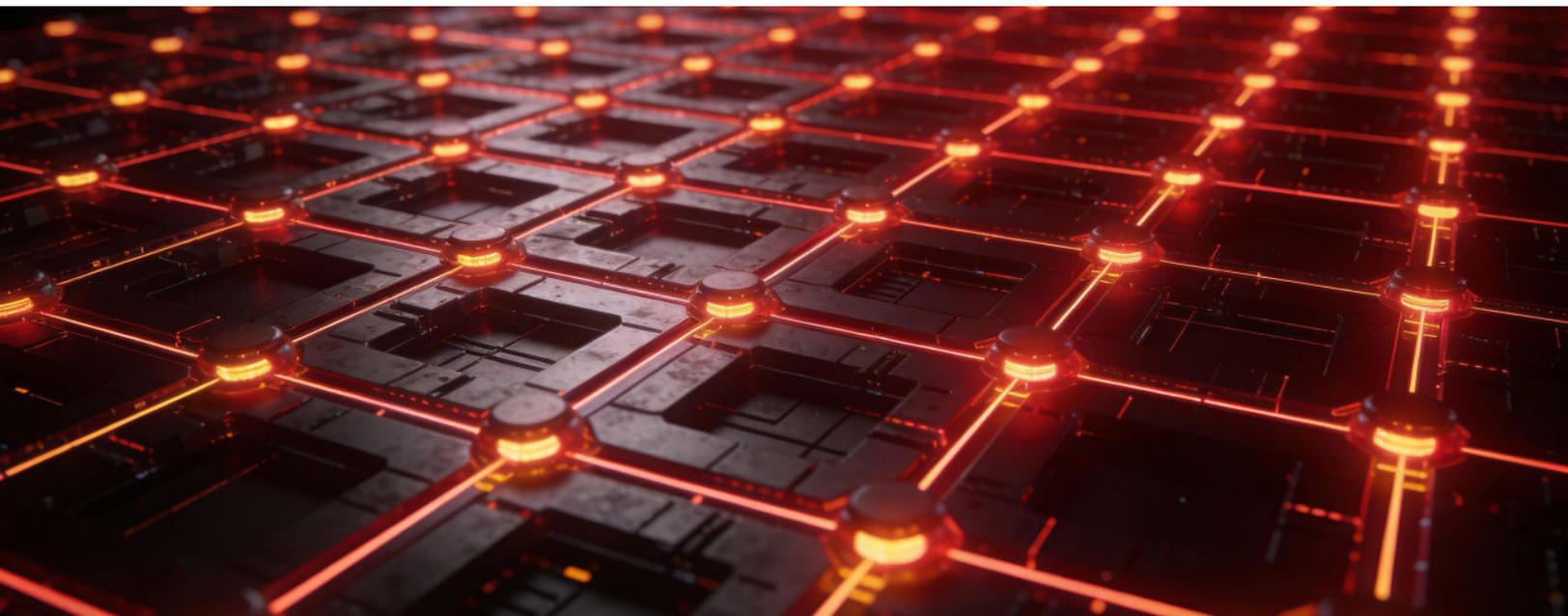
Why are enterprises moving to distributed databases in the first place? It is no longer just about storage capacity. In the AI era, the database must satisfy five non-negotiable operational requirements. If a platform compromises on any one of these, it is not “mission-critical” ready. As shown in Figure 1, modern distributed databases must simultaneously satisfy five non-negotiable requirements, spanning availability, sovereignty, performance, latency, and architectural integrity.

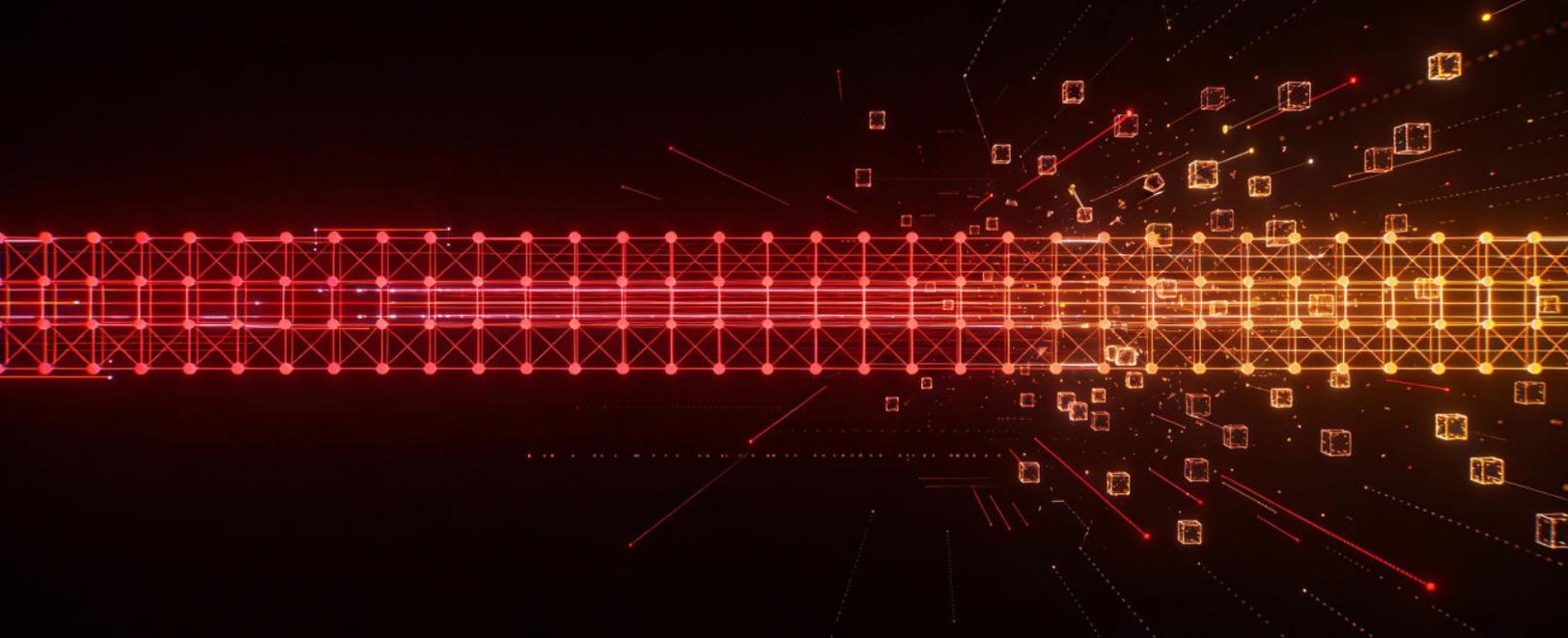
Figure 1. Five Requirements for Distributed Databases in the AI Era



The Challenge: While many vendors claim to meet these mandates, the architecture they choose dictates how well they deliver.

Source: Futurum Research





An Architectural Showdown: Relational Integrity vs. SQL on Scalable Key Value Stores

The choice of a distributed database is no longer just an IT decision; it's a foundational business decision with long-term consequences. Two fundamentally different architectural philosophies have emerged to meet the demands of global scale and availability, and their differences have profound implications for performance, functionality, and enterprise readiness.

The Oracle Approach: A Natively Parallel, SQL-Optimized Engine

Oracle Globally Distributed AI Database represents the evolution of an industry-hardened, enterprise-proven, relational architecture. Oracle's core database is architected to deliver full SQL functionality, high scalability, and high availability in a local environment, and the Globally Distributed AI Database extends the decades of R&D in query optimization, data consistency, transactional integrity, and foundational Oracle Real Application Clusters (RAC) technologies with a globally distributed model that adds flexible data distribution on a shared-nothing architecture to enable distributed capabilities that help customers achieve extreme scalability, availability, and data residency for all Oracle AI Database data types and workloads. The key takeaway is that the architecture is optimized for SQL at its core. It understands tables, joins, and complex constraints as native constructs, allowing it to manage and process data with maximum efficiency and consistency.

The CockroachDB Approach: SQL on a Key-Value Store

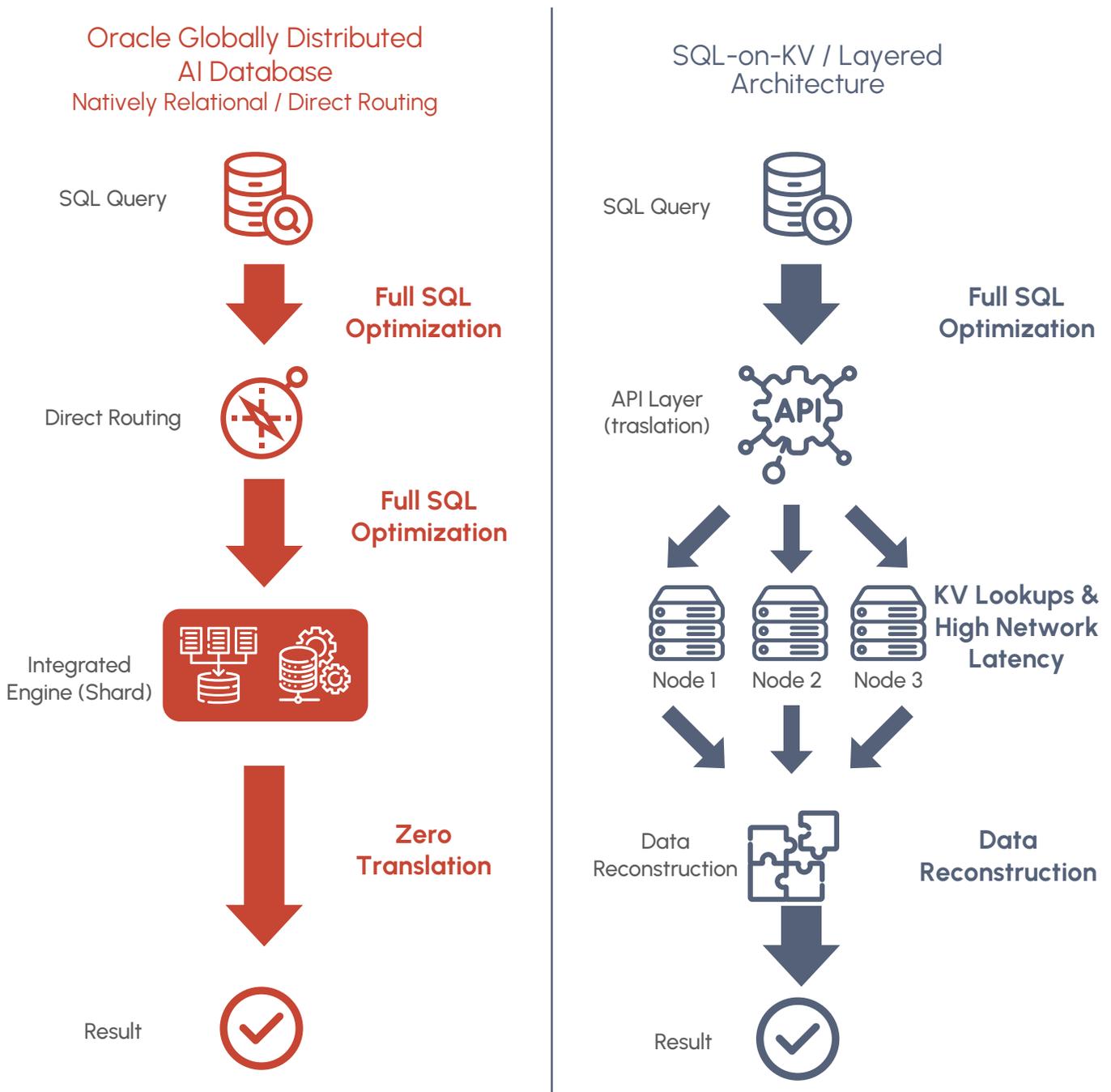
In contrast, many modern distributed SQL databases, exemplified by CockroachDB, were built on a completely different foundation. To rapidly build and ship a database that provided web-scale distributed data and high availability, they adopted an "add-on" architecture: a PostgreSQL-compatible SQL API layered on top of a NoSQL, distributed key-value (KV) store like PebbleDB.

This design creates an interesting impedance mismatch. The database actually speaks two different languages. The upper layer understands relational SQL, but the lower storage layer only understands simple "put" and "get" commands for individual keys and values. CockroachDB maps each row to a key-value pair, and secondary indexes are also stored as distinct key-value pairs. Therefore, a single logical row with multiple indexes results in multiple KV pairs. For complex tables, this mapping is a core part of its architecture.

Consequently, a single structured row of data must be shredded into numerous tiny KV pairs, and a single table is split and scattered across many storage instances, often on different nodes. Every complex operation—a join, a transaction, a constraint check—requires the SQL layer to find all these disparate pieces across the network and laboriously reconstruct the relational data before it can be processed.

The primary barrier to success isn't the data itself, but the plumbing required to move it. A recent Futurum study showed that Integration Complexity and Infrastructure Scalability together account for nearly a quarter of all concerns. These twin challenges create a technical bottleneck that prevents teams from focusing on higher-value applications from transaction processing to AI and analytics⁴. It's a design that trades architectural integrity for simplified distribution, and the performance and functional taxes can be steep. As illustrated in Figure 2, the SQL-on-KV model introduces additional translation and reconstruction steps that are absent in a natively relational architecture.

Figure 2: Comparison of Native Relational Architecture vs. SQL-on-Key-Value Design





Meeting the Mandate for Data Sovereignty and Distribution

In today's global landscape, where and how data is stored is dictated not just by availability and performance, but by a complex web of national and international regulations. As Futurum Research noted in its latest market forecast, the ability to meet these demands is now table stakes.

"Governments and large enterprises are demanding data sovereignty, localization, and cloud-neutrality. Vendors must offer multi-cloud support... and policy-aware data movement capabilities to stay relevant in a regulated environment."

– Futurum Research, 1H 2025 Data Intelligence, Analytics, & Infrastructure Market Forecast

This is where the flexibility of the database's data distribution model becomes a critical differentiator.

Oracle's Strategic Approach vs. CockroachDB's Tactical Options

Oracle Globally Distributed AI Database was engineered with this regulatory complexity in mind, offering a palette of several distinct data distribution methods—value-based, hashed, directory-based, user-defined, and composite. This allows architects to precisely tailor the data layout to the specific needs of the application. In the context of data residency, the most critical of these is Value-Based data distribution, which allows data to be pinned to a specific shard—and so to a specific geographic location—based on a data value like `Country_Code = 'DE'`. Value-based data distribution is the ideal way to help distributed databases address an organization's data sovereignty needs.

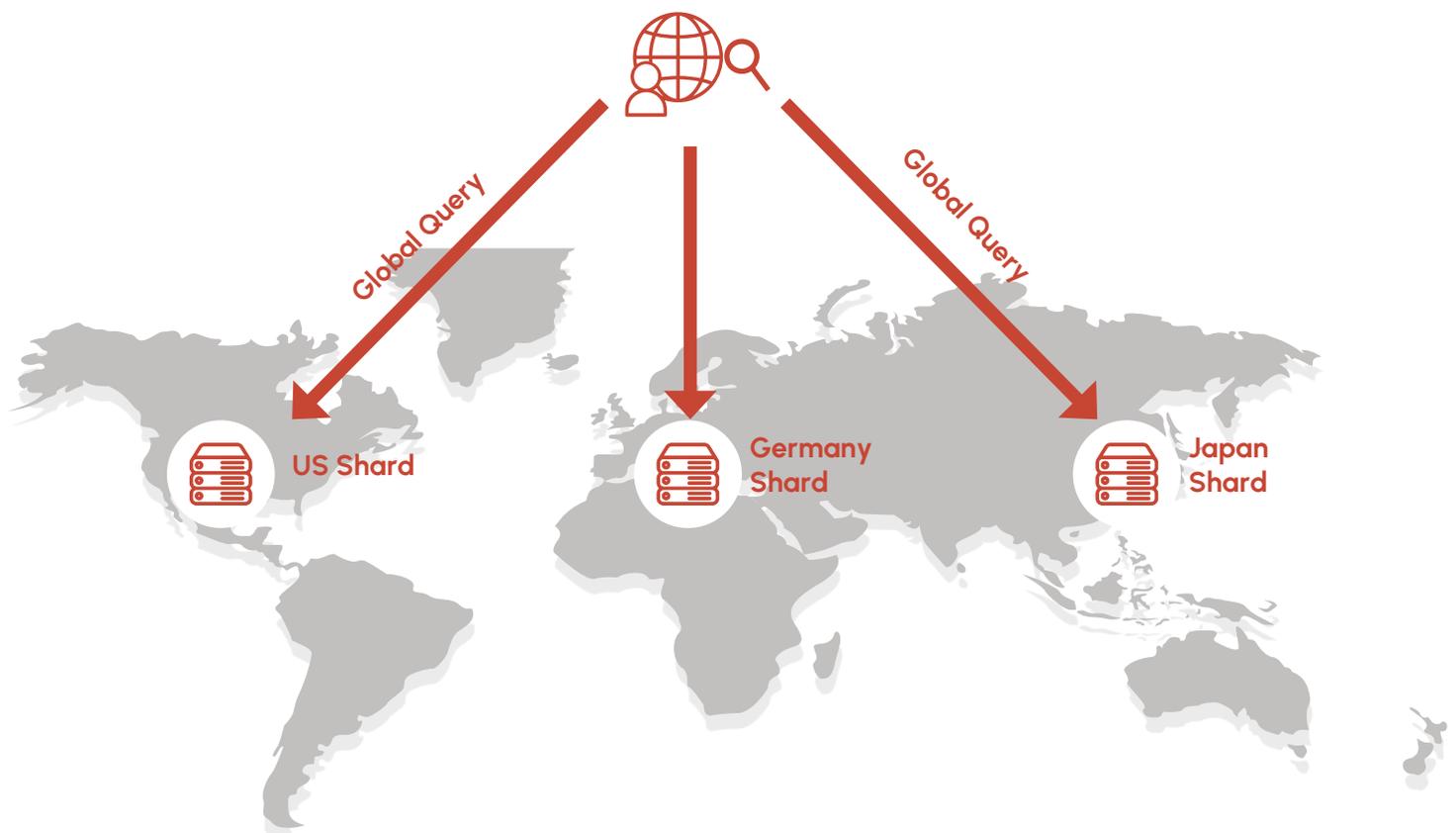
Furthermore, Oracle offers methods like Duplicated Tables to replicate small, read-mostly tables across all shards, and an automatic Data Colocation capability for "Table Families" such as Customers, Orders, Line_Items. With these features, data needed for common joins is physically stored in multiple locations and, most importantly, together on the same node with the data it's being joined with, eliminating costly cross-node network traffic and dramatically improving query performance.

CockroachDB, by contrast, offers fewer data distribution choices, most notably hash and range. Within range-based data distribution, it does provide granular controls, allowing administrators to pin data to specific geographic locations, cloud regions, or even datacenters based on row value. While sufficient for straightforward scaling, this limited toolkit provides significantly less granular control for complex sovereignty requirements.

Critically, the CockroachDB architecture has historically lacked robust support for data colocation, meaning that even simple joins between related tables were forced to become distributed operations that span the network, resulting in slower, more resource-intensive queries. It now explicitly supports REGIONAL BY ROW tables and colocated joins, which are designed to keep related data on the same nodes to improve query performance. However, the architectural reality remains that its KV nature makes achieving efficient colocation more complex than in a native relational design.

Differences in regional flexibility extend to data access as well. Oracle's Smart Client Drivers are topology-aware, routing application queries directly to the shard containing the required data. CockroachDB's architecture often requires requests to go through a gateway node, which then forwards the request, adding an extra network hop and increasing latency on every query. Distributed architectures enable data to reside closer to users while still supporting global query capabilities (see Figure 3).

Figure 3: Solving for Global Scale and Local Sovereignty



Source: Futurum Research



Operational Reality: Scale, Analytics, and Uptime

For simple, key-based operations, such as reading a single row, most distributed databases exhibit comparable performance. The real test of an enterprise-grade platform, and where the "SQL-on-KV" architecture reveals its limitations, is in meeting the demands of petabyte-scale AI and analytics applications, storage-compute separation, and "always on" availability.

The Analytics Wall: Solving for Petabyte-Scale Intelligence

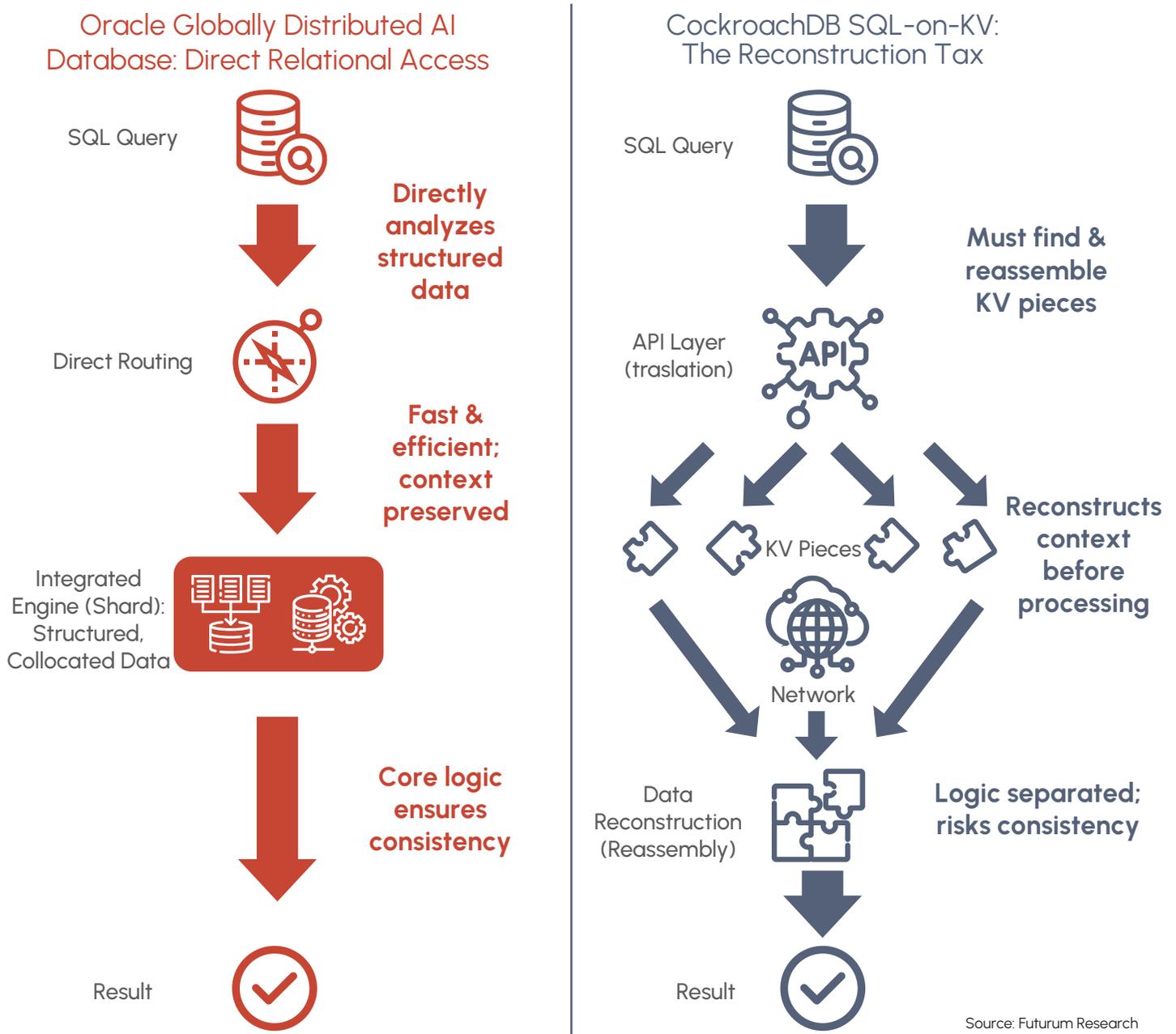
Modern AI applications don't just transact, they analyze. They require the database to ingest and scan petabytes of historical data to feed AI vector models and implement complex joins to generate real-time insights. This is where the CockroachDB SQL-on-KV architecture hits a physics problem. Because CockroachDB shreds relational data into millions of KV pairs, running an analytical query such as "Sum all sales by region for the last 5 years" requires the database to first find millions of KV pairs distributed across the network, and pull massive amounts of raw data to the compute layer to be reassembled and filtered. At the petabyte-scale, this network traffic becomes a crushing bottleneck, effectively rendering deep AI-assisted analytics impossible on operational stores. The only option available to customers using an SQL-on-KV architecture like CockroachDB is to copy their entire data set to a second, more analytics-focused database such as Oracle AI Database, complicating the solution, running analytics on extremely stale data, and incurring higher storage, compute, networking, and management costs.

In contrast, Oracle Globally Distributed AI Database's intelligent data distribution and native relational architecture bypasses this bottleneck by enabling each shard to thoroughly process co-located data to minimize cross-node communication and virtually eliminate data reconstruction overhead. Furthermore, many use cases can take advantage of Smart Scan technology found on Oracle Exadata and Exascale infrastructure to dramatically accelerate analytic queries, further reducing data movement. Instead of moving petabytes of data to the compute layer, Smart Scan pushes the SQL queries down to the storage layer. The database filters and aggregates the data where it lives, returning only the relevant results. Similarly, AI vector queries are offloaded to Exadata intelligent storage. These capabilities enable enterprises to run high-performance AI and analytics on massive datasets without the latency penalties inherent in KV architectures found in CockroachDB.

Storage-Compute Separation: Eliminating the Rebalancing Tax

A critical differentiator for rapid scaling is the infrastructure architecture. In a shared-nothing, SQL-on-KV architecture like CockroachDB, storage and compute are tightly coupled. Adding a node to increase capacity can trigger a cluster-wide rebalancing storm. The database must physically move data ranges to the new node, consuming significant network and CPU resources. This rebalancing penalty often degrades performance for live applications, effectively dampening success. As demonstrated in Figure 4, SQL-on-KV architectures require additional data reconstruction steps that increase latency and complexity.

Figure 4: Architectural Paths to Answering a Complex Query



Oracle solves this via true storage and compute separation on Exadata using the Exascale architecture. By decoupling the processing layer from the storage pool, Oracle allows the database to treat storage as a dynamic, shared resource. When a new compute node is added, it can immediately access the data without requiring performance-limiting data movement. This allows for elastic scaling of compute and storage resources to meet AI demand without downtime or brownout periods.

Always-On Availability and Data Integrity

Finally, for mission-critical systems, "high availability" must equate to "continuous availability." Both platforms leverage the Raft consensus protocol, but their implementations differ in risk profile. CockroachDB implements Raft at the low-level storage layer. While robust for basic uptime, the complexity of managing consensus across millions of tiny ranges can introduce latency variance. Furthermore, the failure of a transaction coordinator node can, in specific edge cases, lead to stalled transactions or lost writes that require retry logic in the application.

Oracle's Globally Distributed AI Database architecture delivers a true always-on experience by using Raft consensus protocols at a higher level. This provides transparent application failover: if a node fails, the user's session is instantly moved to a surviving node, and the transaction continues without the application issuing an error or the user noticing a disruption. For industries like finance and healthcare, this difference between eventual recovery and transparent continuity is the definition of enterprise ready—and in some cases, staying in business.



The Converged Advantage: Futureproofing for an AI-Driven World

The rise of Generative AI has accelerated another key market trend: the need for multi-model databases. Modern applications are not purely relational anymore. Increasingly, they require the ability to handle diverse data types and workloads such as JSON documents for semi-structured data, spatial data for logistics, and most recently AI vectors for similarity search and Retrieval-Augmented Generation (RAG).

The Hidden Cost of a Non-Converged Database

CockroachDB has always focused on being a distributed SQL database, adding select functionality as dictated by market demands. A narrow focus, however, can constrain specialized databases like CockroachDB, making it difficult for organizations to meet their broader data management goals. With limited native support for diverse data types or workloads like vectors, analytics, and machine learning (ML), for instance, organizations are forced to adopt a complex, multi-database polyglot strategy. They often set up a separate document database, a separate vector database, and a separate analytical warehouse, moving and copying data between those databases as needed. Shuffling data around leads to stale data, data latency, data corruption and data fragmentation.

Clearly, this approach can create a nightmare for governance, security, and operations. Each new system adds another silo, increasing integration complexity and making it nearly impossible to maintain a real-time, consistent, trustworthy view of enterprise data. This tends to compound the very problems IT leaders are already struggling with.

Oracle's Converged Database: A Unified Foundation for AI Success

Oracle's strategy provides a powerful antidote to this complexity. For decades, Oracle has integrated first-class support for new data types and workloads as features within its single, converged database engine. This philosophy extends directly to Oracle Globally Distributed AI Database.

With this approach, an organization can solve the challenges of global distribution, high availability, and data sovereignty once and have those benefits apply to all of their workloads. A single instance can manage mission-critical transactions, as well as store and query JSON documents. And beginning with Oracle AI Database, Oracle can manage and search massive, distributed AI vector indexes. This allows enterprises to combine a vector similarity search with a relational query on customer data in a single, globally distributed, and transactionally-consistent operation or perform a RAG operation combining data from a Large Language Model (LLM) with internal data across locations. This unified approach dramatically simplifies architecture, streamlines governance, and provides the stable data foundation that AI initiatives desperately need to succeed.

Closing the AI Foundation Gap

Enterprises cannot succeed with their #1 priority—AI—if they don't solve for their #1 frustration—poor data quality & governance. Oracle's converged distributed database simplifies governance and improves data consistency across all workloads—relational, JSON, analytical, and AI vector—directly reducing the risk of AI project failure.

Market Leadership and Conclusion: The Right Choice for the Enterprise

In a dynamic and rapidly growing market, technology choices must be grounded in both future potential and proven stability. An examination of the market landscape shows a clear preference for enterprise-grade, reliable solutions from established leaders.

A Proven Leader in a Dynamic Market

According to Futurum Research's market analysis, Oracle maintained its dominant position in the global database management and analytics market in 2024, with revenues reaching US \$24.7 billion, significantly outpacing its nearest competitors. This leadership is not accidental; it is built on decades of innovation and a deep understanding of the reliability, security, and performance that mission-critical enterprise applications demand. As shown in Figure 5, Oracle maintains a significant lead in database market revenue relative to its closest competitors.

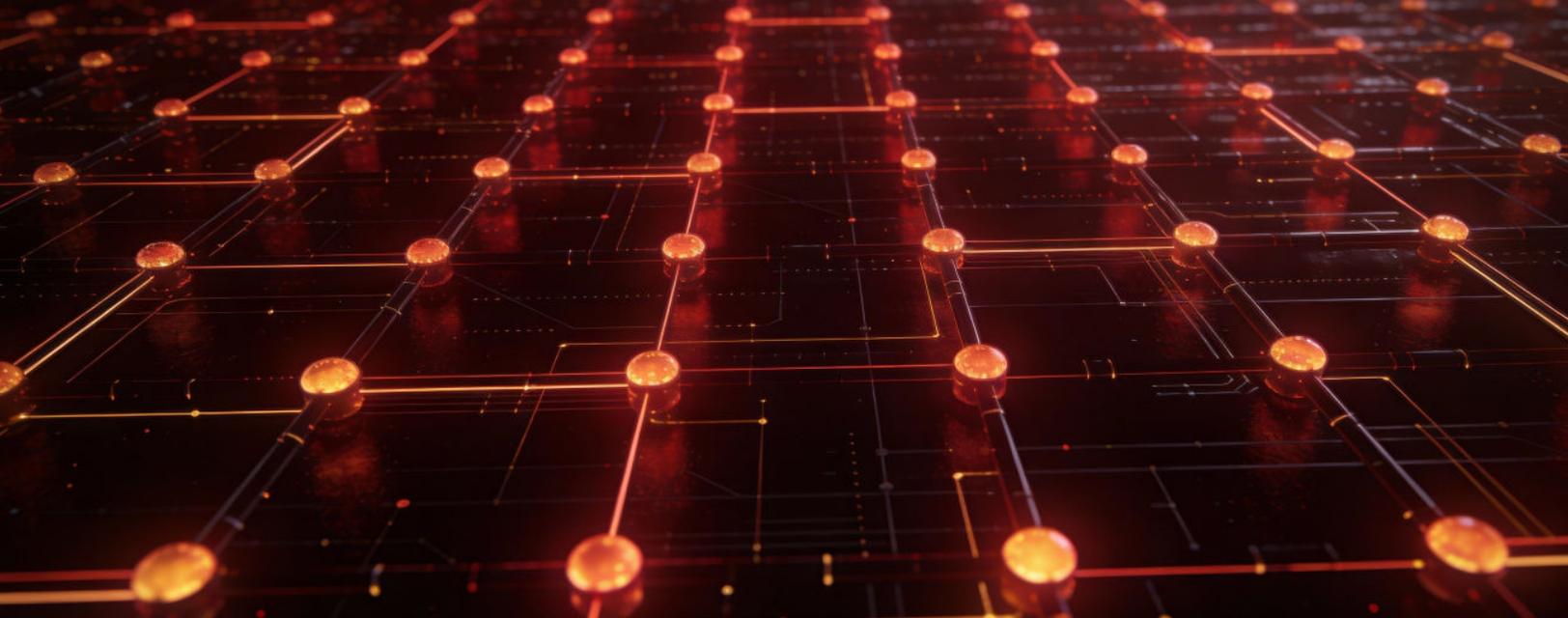
Figure 5: Global Database Market Revenue Comparison by Vendor.



Source: Futurum Research, 1H 2025 DIAI Market Sizing & Five-Year Forecast

This leadership extends to its strategy for the modern, hybrid world. With flexible deployment options that span on-premises, multicloud, and fully autonomous cloud services, Oracle meets customers where they are. This aligns perfectly with Futurum's survey data, which shows both Public Cloud (52%) and Hybrid (47%) as the dominant enterprise deployment strategies⁵. Strategic partnerships, such as Oracle's unique multicloud database partnerships with AWS, Azure, and Google Cloud that make Oracle AI Database services available inside their data centers on Exadata infrastructure. This multicloud push underscores Oracle's commitment to providing customers with choice and performance, regardless of their application's cloud provider.

⁵ Futurum Research, 1H 2025 Data Intelligence, Analytics, and Infrastructure Decision Maker Survey Report



Conclusion: An Architecturally Superior Choice for the Mission-Critical Enterprise

The race to infuse AI into every facet of business has put the underlying data platform under an intense spotlight. While distributed databases like CockroachDB have successfully addressed the challenge of horizontal scaling and basic, rapid availability, their "SQL-on-KV" architecture forces significant compromises in performance, full SQL functionality, and operational simplicity when it comes to AI applications.

Our analysis concludes that Oracle Globally Distributed AI Database represents a more mature, robust, and architecturally sound foundation for modern enterprise AI. Its superiority is rooted in three key areas:

- 1 A True Relational Engine:** Its natively parallel SQL architecture avoids any performance-killing "impedance mismatch," delivering superior performance for the complex transactional, analytical, and AI workloads that define modern business..
- 2 Unmatched Flexibility and Control:** With a comprehensive toolkit of data distribution and replication methods, it provides the granular control needed to solve for complex data sovereignty and precisely tune for performance in a way that rigid, one-size-fits-all approaches cannot begin to help address.
- 3 Converged Simplicity:** By supporting all modern data types and workloads within a single, unified data engine, it provides a powerful antidote to architectural complexity, simplifying governance and providing a stable, future-proof platform for AI innovation.

For enterprises building the next generation of mission-critical global AI applications, where performance, data integrity, flexibility, and addressing localization mandates are non-negotiable, Oracle Globally Distributed AI Database stands as the clear and superior architectural choice when compared to CockroachDB.

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

This document, including any supporting materials, is owned by The Futurum Group. This publication may not be reproduced, distributed, or shared in any form without the prior written permission of The Futurum Group.

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 | [futurumgroup.com](https://www.futurumgroup.com) | (833) 722-5337