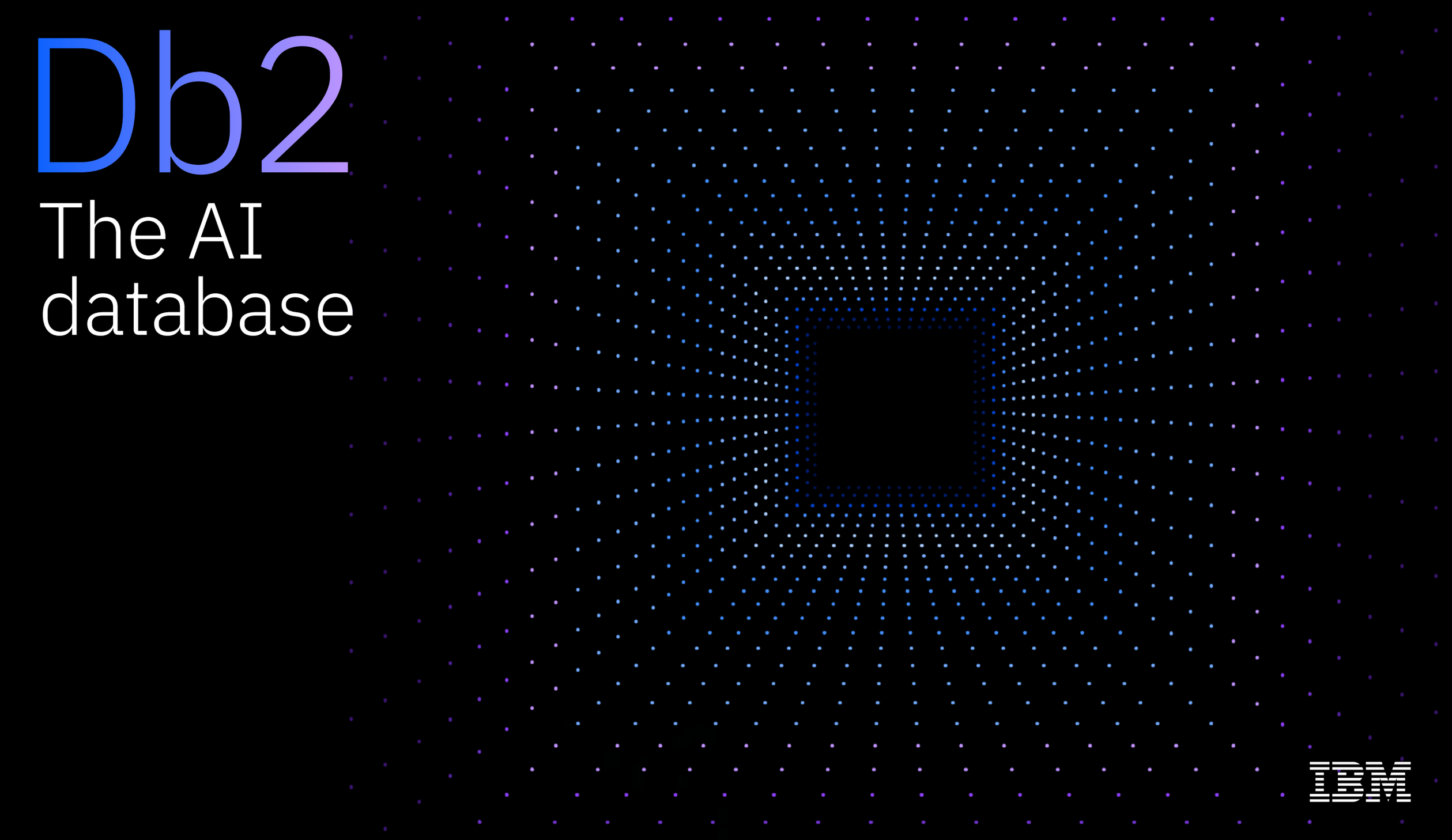
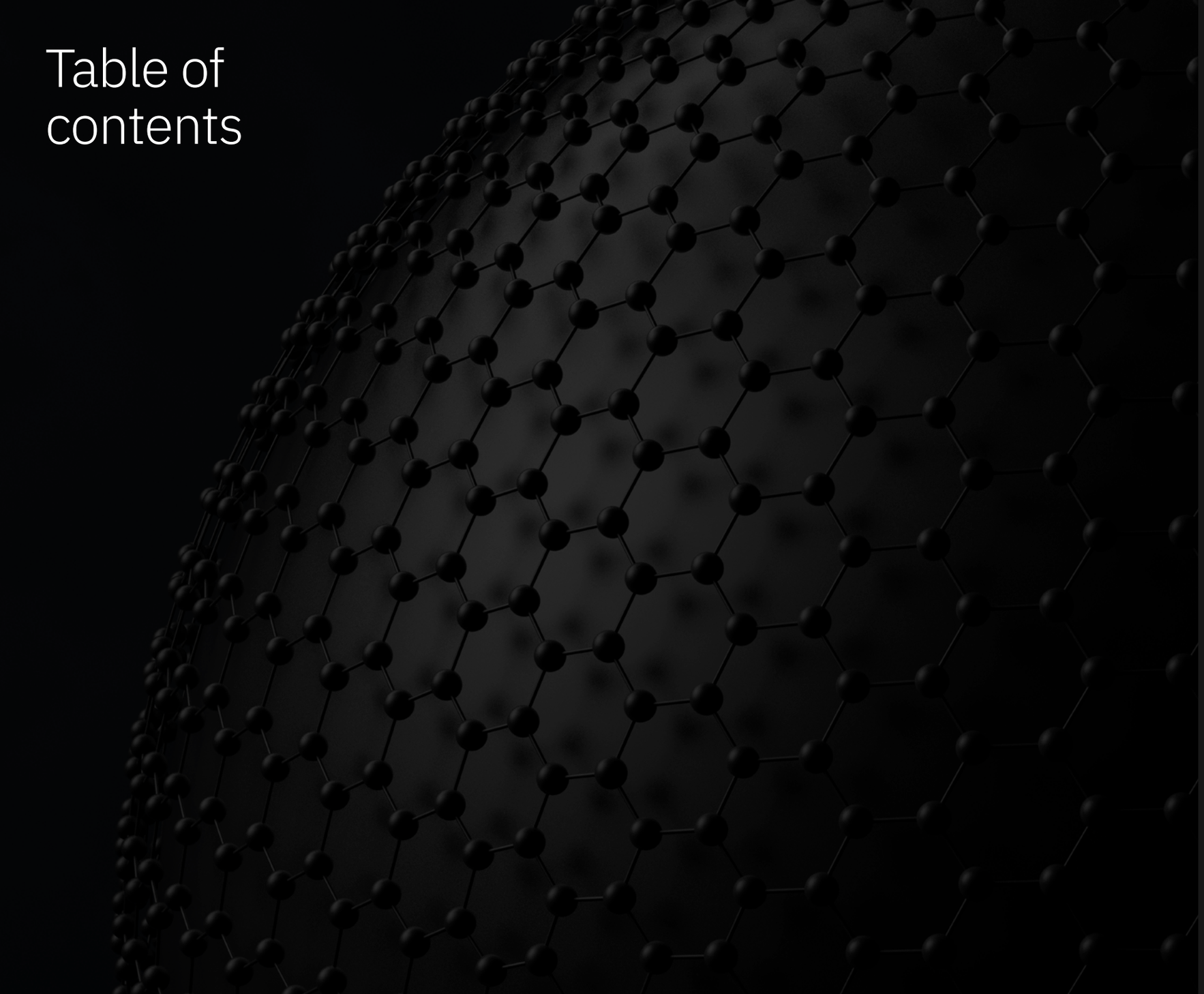
Db2

The AI database



IBM

Table of contents



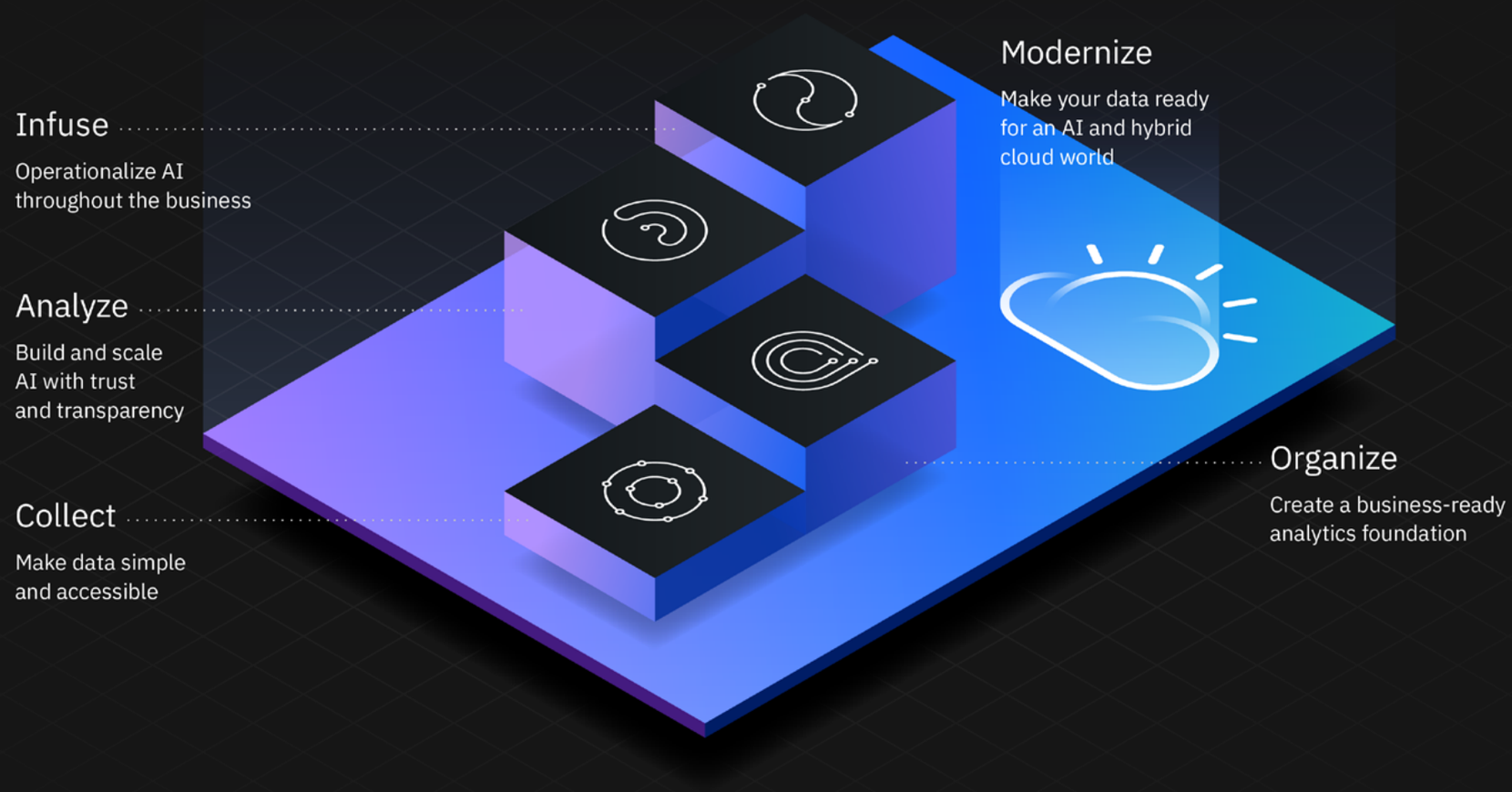
The AI database:

Powered by AI, and built for AI

Rising volumes and variety of data demand new expertise, processes and infrastructure to fuel digital transformation across industries. Machine learning (ML) and artificial intelligence (AI) are key parts of this effort as computers process data and learn from their experience without continual human intervention.

AI and ML aid traditional and nontraditional computing tasks—from analytics and logistics to natural language interaction, music composition, chemical design and fraud detection. They bring data-driven intelligence to businesses—so enterprises can understand what has happened, is happening, and is likely to happen.

The journey to AI is like a ladder where each component is a rung. Those who undertake the journey strengthen the ladder and make it more useful.



The journey to AI is founded on a robust data infrastructure that enables organizations to access, collect, organize and make all data available for analysis, no matter its type, source or deployment. With ML-driven data management, AI is both the goal and the means of achieving it.

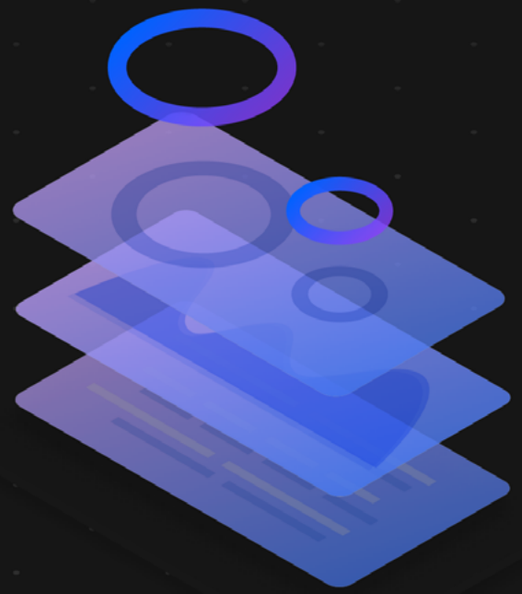
Integrated solutions infused with AI and ML make data management simpler, faster and more intelligent—enabling developers and data scientists to use data in new ways, building AI capabilities into the entire business, from operations to model building to application development.

Powered by AI

The machine learning and AI features built into IBM® Db2® fuel several game-changing data management features, including:

- Full-stack data virtualization
- Automatic workload management and resource optimization
- 10x better query performance
- Confidence-based query results

Data virtualization



What it is

A combination of data federation plus an abstraction layer. It provides the ability for all users and applications to interact with multiple data sources at once from a single access point, regardless of the type, format, size or location of the underlying data. This greater access helps to break down data silos and enable data to be used faster and more efficiently to scale AI adoption.

Why it matters

A single entry point to the data provides the following benefits:

– **Simple data access for your data professionals**
A single view of all data enables data search and access across diverse data sources: structured or unstructured, relational or NoSQL without having to spend time and resources moving the data around. Developers and data engineers have access to every possible type of data—both historical and new—and can draw insights from unexpected combinations of that data.

– **A single point of control for governance and security**
Security and governance arrangements are much simpler, more robust and less prone to failure when your admins can simply point all users and applications to a single access point for all your data repositories.

– **Reduced data transfers**
Data processing can be done in the repositories themselves, rather than transferring queried data to a local repository first. This can dramatically reduce latency and bandwidth costs.

How it works

Modern enterprise data environments depend on multiple datastores. These include transactional data in a relational datastore, sentiment data in a Hadoop cluster, historical customer data in a data warehouse, and clickstream e-commerce data in a fast data cluster.

Without virtualization, everyone who wants to interact with this data—from database administrators (DBAs) to app developers and data scientists—would need to implement a custom access method for each data store. At best, this would be an enormous headache for everyone involved. It would limit the insights that users could glean from comparing different types of data against one another (customer history with real-time browsing behavior, for example). And at worst it would risk corruption of the data and new security breaches.

Providing a single point of access to diverse data throughout the enterprise is why data federation has become so popular. However, IBM data virtualization goes beyond federation, by providing an abstraction layer and letting users define their own terminology and methodology for data access (see Figure 1).

In action

Data virtualization

The European bank ING is working with IBM to implement a single point of data access for all its users worldwide. This lets the bank manage a global data infrastructure while still delivering the performance, scalability, security and governance features they need. This also reduces disconnects between different areas of the business.

With the IBM solution in place, new data can be added to the ING platform from anywhere in the world and accessed by any of the bank's global users without the need to modify individual data access schemes or security permissions.

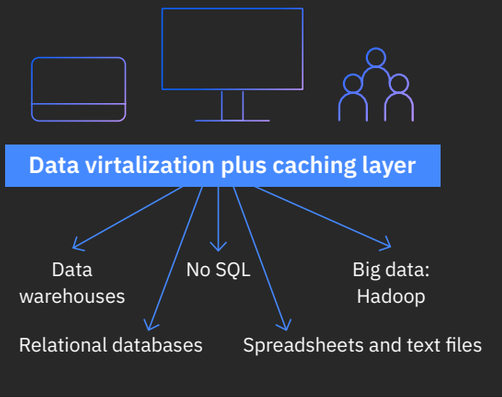


Figure 1. Representation of how virtualization permits unified access and governance of multiple data sources

Adaptive workload management and resource optimization

What it is

Intelligent technology that automatically allocates data resources to handle a variety of workloads. The technology is both automated and adaptive in that it responds to past conditions and also predicts future demands.

Why it matters

With adaptive workload management, you get a stable, high-performance system out of the box, with no need for labor-intensive tuning and setup. This is essential for modern enterprises because of the complexity of their data platforms and the high failure rate of manual workload management strategies.

When too many workloads arrive at once, a database needs to determine how to handle the extra demand. So-called “open” or manual schemas have no feedback mechanism; they simply require the user to set predetermined limits for the number or size of workloads. But these limits can break down when confronted with a complex set of workloads, and even in best circumstances require constant manual monitoring and adjustment. This results in database performance issues, under-utilization or even failure.

By contrast, Db2 detects and predicts trends in utilization, so that it can notify users or even automatically correct the issue before the trend becomes a serious problem. This makes life easier for DBAs and lowers the cost of ownership for each database by reducing the hands-on time required for configuration and tuning. IBM testing showed that this better utilization resulted in overall database performance improvements up to 30%.

How it works

Adaptive workload management incorporates machine learning in the form of a feedback mechanism, so the database constantly monitors the expected and actual runtime of various workloads, then adjusts available resources to accommodate any shortfalls. Db2 databases can run out of the box with minimal setup needed for most workloads.

If manual setup is needed for especially complex situations, this adaptive technology makes it easier. DBAs can create multiple workload classes and specify performance targets for each. The adaptive manager then monitors incoming workloads and allocates resources to hit those targets.

In action

Automated workload management

Automated, adaptive workload management specifically addresses many common pain points involved in managing an enterprise database.

For example, in data warehouse settings, the database may receive a wide range of jobs—everything from streaming ingests to low-latency reports with sub-second response times (such as web page loading) to heavier batch reports that need minutes to complete. The administrator needs to ensure the system doesn't get stuck on larger jobs and prevent urgent jobs from completing.

Adaptive workload manager makes this very easy, so administrators can create resource groupings (also called workload or service classes) and allocate a share of the system to each grouping. The database then intelligently allocates resources to hit performance targets. All this occurs without the need for manual monitoring and tuning of the workloads and resources individually.

AI can be both the goal and the means of achieving that goal.

Machine learning query optimization

What it is

The Db2 Machine Learning Optimizer provides an additional level of intelligent optimization that uses unsupervised machine learning to deliver query execution strategies that improve upon traditional cost-based query optimization.

Why it matters

Basic workload cost optimizers can suggest execution strategies for a given query, but they aren't sensitive to recent changes in the database, and they can't learn from experience. While they can recommend what the fastest execution strategy should be, they will keep recommending that same strategy even if it doesn't work as expected.

The Db2 Machine Learning Optimizer, by contrast, incorporates feedback from actual query performance to recommend execution strategies that deliver the best results, in practice, on your data infrastructure and refine the query path with each execution.

With a machine learning optimizer in place, DBAs don't have to spend their time monitoring system performance and trying to optimize queries. They can instead focus on activities that bring more value to the organization, such as implementing AI applications, developing data use strategies and helping business users throughout the organization make better use of the available data.

How it works

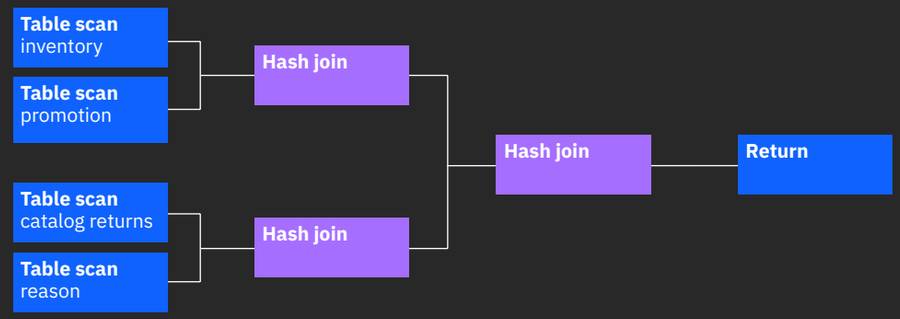
The Db2 Machine Learning Optimizer mimics neural network patterns to optimize query paths, resulting in some queries being completed 8-10 times faster (IBM internal testing).

In action

ML query optimization

A traditional cost optimizer uses statistical and resource modeling to evaluate execution strategies for a given query. It returns the first option shown in Figure 2 below: join two pairs of tables, then join the results to deliver a return. However, a machine learning algorithm can learn from experience and realize that a superior execution strategy is available: join a pair of tables, join a third table to that result, and then join a fourth to that result in order to deliver the same return.

Without machine learning



With machine learning

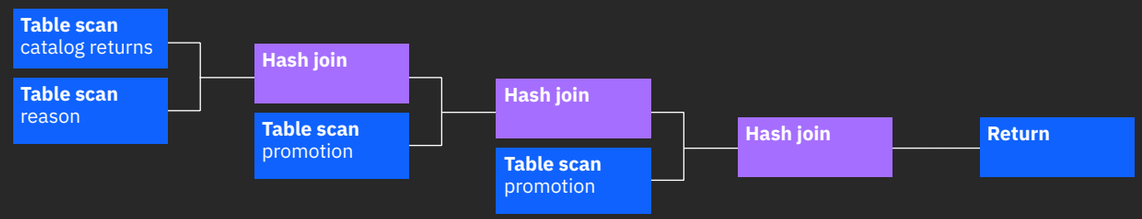


Figure 2. Example of cost-optimized (top) and machine-learning-optimized (bottom) query execution strategies

These changes in execution strategy can be applied in real time, in response to actual query executions. For time-sensitive applications like fraud detection, this real-time improvement can be critical.

Confidence-based query

What it is

An advanced feature that delivers SQL query results in terms of probabilities, or “best matches,” rather than a simple yes-or-no answer.

Why it matters

Confidence-based matching has been available in machine learning settings, but this Db2 feature extends it to SQL expressions as well. For an SQL engineer, this feature dramatically expands the range of possible data tasks that can be done without involving a data scientist—making the SQL engineers more valuable to the business and lessening the burden on already overworked data scientists.

How it works

Confidence-based query adds machine learning extensions to SQL by implementing deep feed-forward neural nets. This variety of unsupervised deep learning finds areas where both the “likeness” and the likelihood of a match are high. Confidence-based querying can be applied to:

- Similarity/dissimilarity queries
- Inductive reasoning queries (for example, semantic clustering, analogies, odd-man-out)
- Semantic group-by operations
- Pattern anomalies (for example, fraud detection)
- Images, audio, video

In action

Confidence-based query

A police database is a setting with many variables—height, weight, age, skin color, eye color, distinguishing marks and many more—and virtually all of these are uncertain because of the unreliability of witnesses. A traditional database search uses a very complex SQL query to account for this uncertainty by manually creating a “range” of values that will match a given witness statement. For example, a reported weight of 195 lb. might match suspects with actual weights between 185–205 lb. These ranges are cumbersome to use and liable to wrongly eliminate suspects if a value provided by a witness is too far outside the range. By contrast, an officer attempting to match witness statements with a suspect profile can use confidence-based query to generate a simple probabilistic SQL statement that finds the subject who best matches the overall witness profile, even when individual data points fall outside the original range of acceptable matches.

Built for AI

Db2 is also built for AI, which means your data professionals can more easily enable powerful AI-driven workload upgrades and workflow improvements. These include:

- Exploring data faster with natural language queries
- Using popular languages and code examples to build AI applications
- Modeling complex relationships with graph and SQL
- Analyzing blockchain data natively

Exploring data faster with natural language queries

What it is

Augmented Data Explorer is a “Google-like” web-based interface for exploring complicated business data. It offers natural language search for users and REST APIs for developers. A question posed in natural language will generate data visualizations and summaries that draw from all available data sources.

Why it matters

As the data landscape becomes evermore complex, more users are faced with increasingly vast data sets. The skills to develop SQL queries or analytics apps are in short supply, but more people than ever want answers from data, fast.

Augmented Data Explorer is an early-stage data exploration tool. It opens up sophisticated data queries to non-technical users, who simply need to enter a natural-language query such as “sales for 2019” or “how often does it rain in October?” This will generate an SQL statement to query all the relevant data and return the desired result in natural language form, a dynamic data visualization, or both.

In the past, business users would typically need a combination of manual processes to translate natural language requests into the required SQL query and expert advice to help with the inevitable snags that arise. Augmented Data Explorer is designed to avoid both requirements and offer new avenues to finding new value in data you already have.

Data scientists and more sophisticated users can use Augmented Data Explorer to perform initial queries on unfamiliar or large data sets, getting an idea of what those data sets contain and where investigations can start. Augmented Data Explorer also offers REST APIs so that developers can easily embed these natural language search and prediction capabilities into their own applications, as opposed to spending time coding their own solution.

How it works

Augmented Data Explorer uses ML technology to crawl and index data to reveal hidden insights. Its AI capability is focused on natural language support—understanding context, synonyms and syntax for a wide variety of disciplines and fields—as well as predicting related topics of interest based on search terms. The tool is containerized for easy deployment and management, and features a straightforward web-based interface.

It also includes graph functionality, revealing at a glance which data sets the explorer engine has actually accessed and crawled. The user can easily see what data (such as schemas and tables) is being drawn upon to produce the answers delivered by the system.

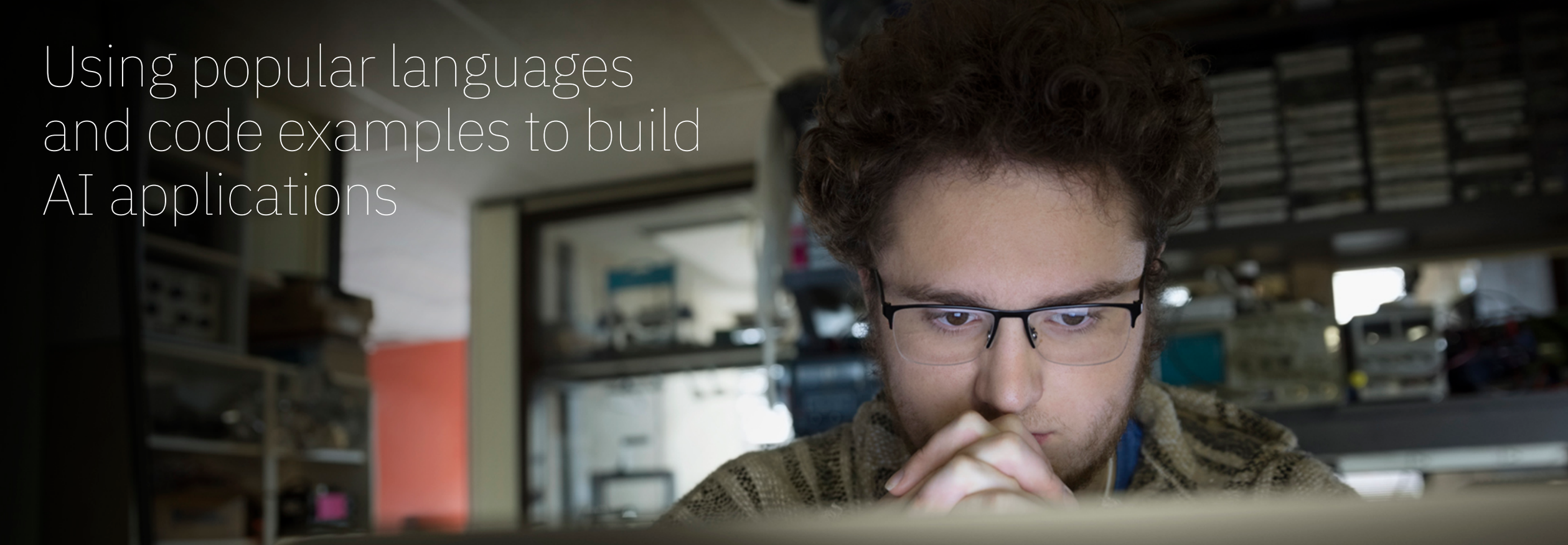
The value of Augmented Data Explorer

Augmented Data Explorer is most valuable in situations where your data users are not sure what questions to ask. It makes it easy for these users to pose questions, discover relationships within data and develop actionable insights. Furthermore, it expands the pool of people who can address these business challenges. In constantly changing business settings where users have many responsibilities, Augmented Data Explorer is the first step in tackling the overwhelming amount of data now available and deriving new value from it.

Consider a sales manager at a retail organization. This individual may have many different questions from day to day or month to month—from breaking down sales by region to understanding customer retention rates to identifying causes for shopping cart abandonment to predicting why repeat customers sometimes disappear to explaining how the weather affects inventory delivery lead times, and much more.

Augmented Data Explorer is designed to identify and quickly answer new questions, thereby extending AI capabilities to a much wider segment of business operations.

Using popular languages and code examples to build AI applications



What it is

Db2 11.5 includes native support for popular languages and libraries—including Python, JSON, GO, Ruby, PHP, Java, Node.js, Sequelize and Jupyter Notebooks. Data professionals can build AI applications in a language they already know, then seamlessly connect them to data in Db2.

Why it matters

With native support for popular machine learning languages, IBM bridges the gap between data management and rapid app development with an ML- and AI-focused data ecosystem. Developers and data scientists can keep their data in an enterprise database and still use existing, popular development languages and libraries.

Native language and library support brings multiple advantages that lead to faster and easier development of AI applications:

- Developers can leverage popular languages and libraries, along with industry-specific code examples on GitHub, to quickly build apps for their own uses.
- Developers can easily write applications that use Db2 data natively—eliminating the need for custom back-end code to access Db2 data, and eliminating one common point of failure or management for their apps.
- New databases can be rapidly deployed and scaled on cloud if needed, then connected seamlessly to apps.
- Developers have access to machine learning development environments (IBM Watson® Studio) within the same database ecosystem.
- Data scientists can spend their time working on machine learning models rather than resolving database access issues.
- Businesses can hire people with the necessary skills to analyze data without needing to learn new skills or languages just to access the data first.

How it works

The AI application development process involves several important questions:

– Where is the relevant data stored?

– How do I access and explore it?

– How do I apply machine learning practices while building the app?

– What languages can I use?

Within IBM ecosystem, there are enterprise-ready answers to all these questions. Data can be stored in Db2 or Db2 Warehouse, Event Store or Hadoop, on cloud or on premises—in virtually any combination of databases. What they all have in common is enterprise readiness, with full support, reliability and performance. The common Db2 code base and the underlying Common SQL Engine mean that applications can query data from any ecosystem component and don't need to be rewritten to run on different parts of the ecosystem. Additionally, a series of newly available drivers for the leading open-source programming languages and frameworks, it is now easy for developers to analyze and build machine learning models into applications using Db2.

Data pros can use some of the other features described in this ebook, such as the Augmented Data Explorer, to get a quick understanding of what the data contains. Once stored in Db2, data can be directly connected to the machine learning development environment in Watson Studio. Industry-specific libraries, code examples and templates give developers a head start in building a functional app. And finally, a set of native drivers converts SQL statements to a ML-language-specific model (such as Python) for further development in the chosen language.

The value of native language support

The potential market for AI-enabled applications is enormous. Just a few examples include:

Retail: Recommendation engines, targeted content, promotions

Insurance: Fraud analytics, application processing

Logistics: Fleet management, predictive maintenance

Transportation: Real-time route and path guidance based on traffic, weather, and more

A legion of developers and data scientists are working in the market to develop smart apps and algorithms, usually using Python, Java or another popular language. The data that feeds these apps resides in one or more databases. There are databases that connect natively to Python, but in general these are not equipped for enterprise scale, performance or reliability. However, without native integration between their chosen ML development languages and the underlying enterprise database, it is difficult and cumbersome for developers to connect the application itself to the necessary data.

With native integration between Db2 and popular development languages, the developer can access the data without needing to involve a DBA or SQL specialist. The result is better productivity for your developers and data scientists, along with more robust and rapidly developed apps.

AI empowers users at any level to do more with data than they ever could before.

Modeling complex relationships with graph and SQL

What it is

Db2 deeply integrates graph functionality with relational data and SQL so that graph applications can run directly from relational data, and the Db2 SQL engine can directly query graph data.

Why it matters

Graph is an impressive tool that provides important insights but was previously incompatible with transactional and OLTP systems. To get around this incompatibility, organizations would keep their normal relational infrastructure but pull certain data out, put it into a graph database and run graph apps on it there. This kind of duplication, delay and overhead is no longer acceptable. Organizations need graph-driven insights in their relational data, and they need them in seconds.

Users in many industries want to add graph insights to their existing toolkit of analytic insights, but without disrupting their data infrastructure or adding a new database solely to support graph. Db2 makes this possible.

How it works

Graph data is stored in tables within the relational framework of Db2 or Db2 Event Store. This means that the SQL engine can query graph data directly, and graph applications can query data directly in the relational tables. This architecture also supports open-source graph query languages, such as Gremlin and Tinkerpop.

The deep integration between relational data and graph capabilities in Db2 delivers several important advantages:

- Graph can run directly against existing relational data, adding a new level of insight on top of SQL-based analytics.
- SQL analytics can run directly against graph data stored in relational tables.
- Graph applications can use a custom API or Spark to connect to Db2.
- ACID transactions update the graph in real-time without disturbing existing relational applications—meaning graphs can be used for real-time transaction processing in addition to analytics.
- Industry-specific solutions are available for graph-heavy industries such as healthcare and finance.

The value of native graph support

There are many cases where graph-based insights could enable new efficiencies and workflows with relational data. For example, a retailer could use real-time graph relationships to note a pattern of returns for a particular item at multiple locations. This insight can then immediately update the retailer's transactional database and result in the item being pulled from shelves the same day.

In a different field, an insurance fraud investigator could use graph to bring together multiple data sets: number and size of claims, relationships between claimants, identities of service providers, and more. The relationships surfaced in graph can reveal the probability of fraud.

Analyzing blockchain data natively

What it is

The native Blockchain Connector API in Db2 provides unprecedented transparency and insight into the highly compressed data contained in blockchain ledgers.

Why it matters

Blockchain technology is being rapidly adopted in many industries, but this is causing analytic bottlenecks. Huge amounts of potentially valuable information are contained in compressed form in each blockchain ledger, but there has been no simple way to view and analyze that information. The Db2 Blockchain Connector changes that status quo. These compact, transparent, controllable and unalterable blockchain ledgers are now open to analysis like any other data source in the enterprise. Enterprises get all the benefits of the Db2 engine—performance, flexibility, scalability and security—without needing to develop ad-hoc reporting solutions for blockchain alone.

The blockchain connector is even more valuable because of the native connectivity and interoperability of other data types available in Db2. Enterprises can easily integrate additional data from other data stores to gain context and open up new analytical possibilities for their blockchain data. And finally, this connector opens up the possibility of using blockchain data for AI apps—a use case that previously would have been very cumbersome because the blockchain data was effectively its own silo. AI developers can now easily incorporate blockchain data sets, either as a primary data source for their apps or to provide additional granular detail.

How it works

The Blockchain Connector surfaces the transactional data that is compressed and stored within the blockchain ledger and presents it as a relational table in Db2. A robust caching strategy uses existing Db2 functionality to create a cache table for the blockchain data to improve query performance while still allowing users to fetch the newest data when needed.

The value of native blockchain support

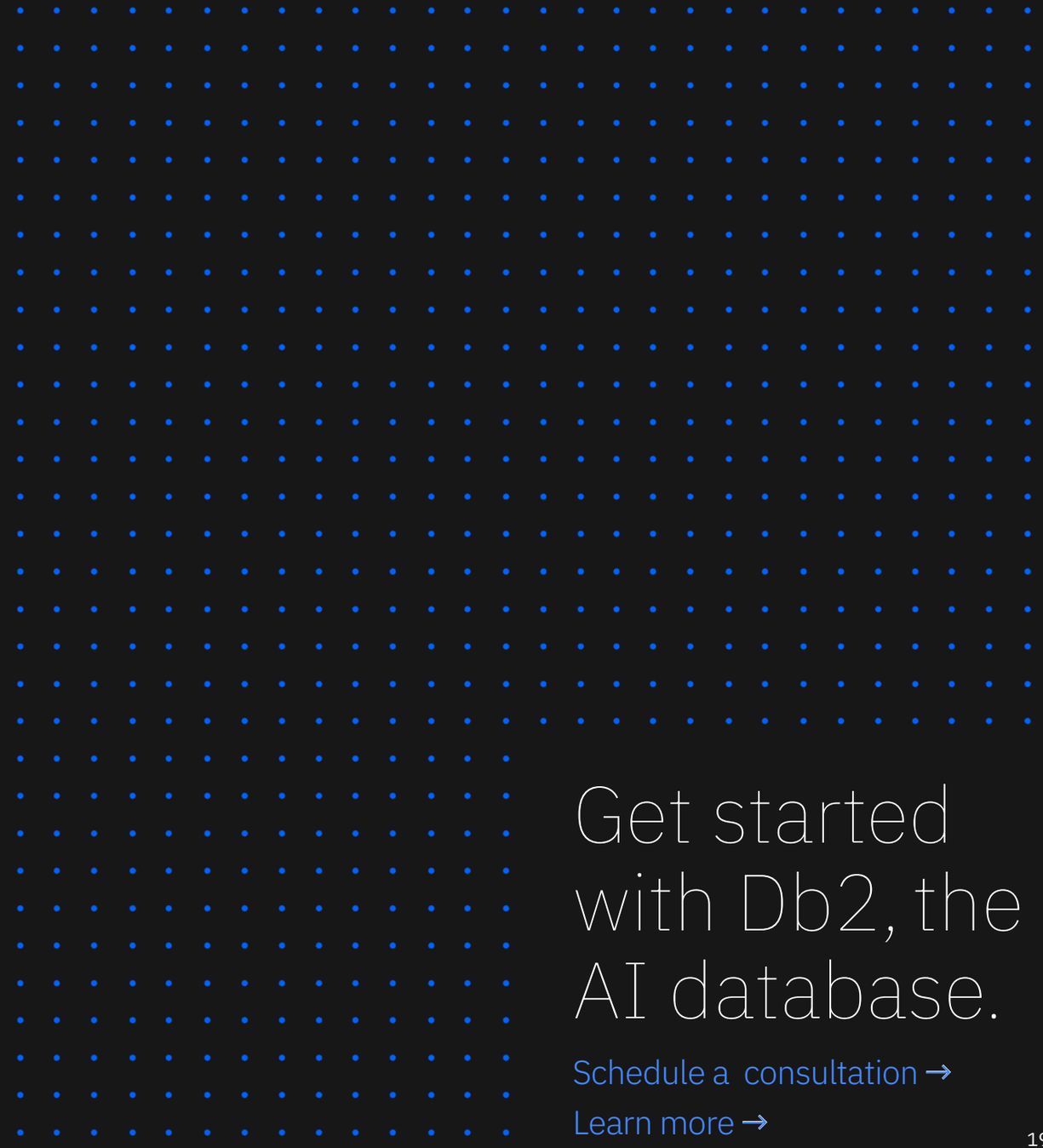
Blockchain is already being used by industries such as insurance, finance, logistics and healthcare to store an immutable record of transactions, but these industries also want to run analytics to understand their data better. For instance, insurance companies want to know how many claims were opened per customer in a particular geography; shipping companies want to track containers and vehicles; healthcare companies want to understand a patient's treatment records.

Instead of building a custom back-end to extract this data, they can simply connect Db2 to their blockchain data store and run their analytics against it as they would on any other data. Because of the inherent connectivity and flexibility of Db2, they can correlate additional contextual data with the blockchain transactional data (for example the weather during a particular shipment; the associates of an insurance claimant who might have made similar claims) and gain a much richer and fuller understanding of what each transaction means for their business. AI apps running directly on blockchain data are now a possibility.

The place for Db2 in an AI-driven organization

Whether you deploy Db2 capabilities as a traditional on-premises deployment, a cloud relational database or warehouse instance, the subscription-based Hybrid Data Management Platform, the new IBM Cloud Pak™ for Data, or some other deployment model, the underlying technology is exciting and the business benefits are dramatic.

Db2 is positioned to help make AI a reality for users in any size organization. The value proposition is clear. Build, run and manage enterprise AI, on any infrastructure: on premises or on cloud. Tap into all your data sources, regardless of where they physically are. Take advantage of AI capabilities infused into the database to deliver predictive and proactive insights for data-driven decisions. And with Db2 as their data foundation, businesses can more efficiently build and connect AI applications, gaining actionable insights quickly and making AI accessible to more people than ever before.



Get started with Db2, the AI database.

Schedule a consultation →

Learn more →