Application-Driven Intelligence

Defining the Next Wave of Successful Modern Apps

July 2023

US 866-237-8815 • INTL +1-650-440-4474 • info@mongodb.com.

Table of Content

Executive Summary

To compete and win in the digital economy you need to make your applications smarter. Smarter apps use data, AI, and analytics to engage users with natural language, generate insights, and autonomously take action. The results are twofold. Firstly your apps drive competitive advantage by improving customer experience and satisfaction as they interact with your business. Secondly your apps unlock higher efficiency and profitability by making intelligent decisions faster.

For these smarter, AI-powered apps to deliver value, they have to work with live data directly within the flow of the app, in real time. Think:

- Serving hyper-personalized offers and using natural language to interact with customers via chatbots while they are browsing an ecommerce site.
- Detecting and blocking fraudulent transactions during payment processing.
- Dynamically adjusting pricing while a user is hailing a ride.
- Real-time captioning and translation from audio and video streams.

Alongside smarter apps, the business also wants automated decisioning and predictive insights into operational processes so they can react to what's happening "in the moment". Think:

- Monitoring inventory to optimally time the launch of a flash sale or adjust supply chain orders.
- Summarizing support cases and extracting sentiment from online reviews to continuously inform product and promotional improvements.
- Predicting equipment failure from analysis of sensor telemetry.
- Generating and testing multiple copies of SEO-optimized web pages to drive uplifts in discovery and clickthrough rates

While these examples illustrate different use cases and industries, they all have one key demand in common: the need to provide real-time intelligence, ML model inference, and action using current data in live systems. Their outputs control application behavior and provide ground-truth context to Generative AI.

This class of intelligence and analytics is very different from the traditional BI reporting queries consumed by humans staring at dashboards or predictions generated in batches by offline ML models on historical data.

To build this new generation of intelligent applications, we need to do things differently. No longer can we rely on just copying data out of our operational systems into centralized analytics data warehouses and data lakes. While this approach is not going away anytime soon, it's not enough on its own. That's because moving data between different systems takes time and creates separation between analyzing application events and taking actions.

Instead we have to bring a new class of AI and analytics processing directly to the source of the data – to the applications themselves. We call this **application-driven intelligence.** It's an approach that both developers and analytics teams need to be ready to embrace, adopting the new tools,technologies, and workflows needed to deliver value by working with fresh data in real time.

Current state of analytics and AI

It's often said that applications run the business while analytics manages the business. Traditionally, these functions have co-existed in separate domains built by teams with different skills, serving audiences with different needs, with data duplicated and stored in systems optimized for different tasks. Digging into these differences, Table 1 below compares how applications and centralized analytics systems work with data.

As the table shows, centralized analytics systems are designed to do a number of things really well. But they were never designed to meet the demands of operational applications serving thousands of concurrent users. They do not provide the fine grained, millisecond-level access to subsets of records typical of application queries. Neither do they support data sets that the application is constantly adding to, updating, and then querying in real time to retrieve the latest business context for an LLM prompt or model inference.

What analytics systems are designed to do is run heavy duty, highly complex BI reporting and data science queries serving a finite, internal audience. These queries scan hundreds of thousands to millions of records, each taking minutes and often hours to complete. The reporting queries power "manage the business" dashboards consumed by humans and the training of specialized, purpose-built machine learning models. In some cases, the outputs of these queries and models are loaded back to operational databases where they can be consumed by applications. In these cases it is important to remember that

these analytics outputs have been generated in batches against aged and potentially stale data. They are not grounded with the latest, real-time state of the business.

	Application (Operational Database)	Centralized Analytics (Data Warehouse / Data Lakehouse)
Built by	App developers	Data engineers, data analysts
Used by	Application users (humans, algorithms, context retrieval for online LLMs and ML inference)	Business analysts, data scientists, LoB decision makers, offline ML training
Query pattern	Random access to subsets of data in a table (OLTP)	Sequential scans across most or all of the data in a table (OLAP)
Avg. records processed per query	One to low thousands. KBs to MBs of data	Tens of thousands to millions. GBs to PBs of data
Data latency (freshness)	≤ Seconds	Minutes to months
Query latency	< 1 second	≥ minutes and hours
Query and user concurrency	≥ Thousands	Tens
Data update frequency	Continuous	Multiple times per day or nightly
Number of data sources	Consumed from one application	Consumed from multiple applications

Table 1: *Operational databases powering applications and data warehouses powering centralized analytics are designed and optimized for different workload demands.*

Beyond the technical differences between applications and analytics, you also have to consider the impacts of departmental silos. Developers are often embedded in teams that work in lockstep with business functions. In fast moving digital markets, they are continuously adding new application functionality and fixes demanded by users.

On the other hand, analytics teams tend to be housed in a centralized back office department that is balancing the demands of many different constituents. That is because every part of the business has an insatiable appetite for more data, analytics, and ML models. Each new report requires the analytics team to:

1. Identify the appropriate data sources containing the requisite data.
2. Configure ETL pipelines to move data from its source into the data warehouse or data lake.
3. Design the schema and queries, and in some cases select and train the ML model, needed to satisfy the request.
4. Update any associated data catalogs and metadata with details of the new data flowing into the data warehouse or data lake.
5. Rinse and repeat each time new application functionality demands a schema change to the source system or the business wants to track a new metric.

It's not unusual then for new analytics requirements to be added to an ever growing backlog, taking weeks or even months to be delivered.

The dissonance in rate of delivery between applications and analytics further inhibits the shift to application-driven intelligence.

What needs to change?

To deliver smarter apps and improve the business' ability to react to new data and events, we need to make intelligence and analytics part of the application. This shift is what we define as application-driven intelligence.

Application-driven intelligence isn't trying to bring the data warehouse or data lake into the application. Rather this is a new class of AI and analytics that delivers intelligence, insights, and automation for thousands of concurrent users at low latency on fresh data. Often the inferences, prompts, and analytics outputs are consumed by software and models to generate contextual experiences, unlock immediate insights, and drive actions. They are not dashboards used to manage the business or signals to train and tune offline custom models. Therefore analytics and AI queries in the app are touching tens to low thousands of records, not millions. They are working with live data being processed by the application, not data that has been ingested and joined from dozens of different data sources over months or even years.

To meet these requirements, there are a set of capabilities essential to success. For developers this means a data platform — *designed for them as developers* — that makes it easy and fast for them to process application data in all sorts of new and interesting ways:

- Tools and APIs that help them build more sophisticated queries against live data of any shape and structure.
- Programmatic APIs and connectors that integrate the apps they are building into the business' chosen ML models and broader AI ecosystem.
- Indexing and storage formats optimized for analytical processing, eliminating the need to write complex and brittle application-side logic.
- Mechanisms to separate operational from analytical processing so the application doesn't slow down, along with the ability to land insights and inferences close to users.

But does this drive to enable developers mean that analytics teams no longer matter? Quite the opposite. The reality is that the most valuable, relevant, up-to-date data lives in applications, and the analytics team needs to get access to it to deliver real time insights. They need access to this live data in a way that is isolated from the application itself while using their existing skills and tools they are already familiar with. And they need to be able to do this without the overhead of ETLing data back into a centralized data warehouse.

Table 2 below summarizes the required capabilities needed for application-driven intelligence.

Application-Driven Intelligence
Foundational Capabilities
1. Flexible data model to natively store, enrich, and index data of any structure – regular transactional application data, time series measurements, geospatial coordinates through to high dimensionality vector embeddings.
2. Versatile query engine with idiomatic APIs supporting almost any query shape. Supporting point queries for simple inference through to sophisticated data processing pipelines to search, aggregate, transform, and enrich structured and unstructured data for model and analytics engine consumption.

Application-Driven Intelligence	
3. Distributed cloud-native architecture to scale out large data sets, parallelize complex queries across nodes and data partitions, isolate operational from analytical workloads, and land insights close to users. 4. Integration with MLOps platforms, LLMs and frameworks, data visualization tools, streaming platforms, data warehouse and data lakes. 5. End to end security and governance. Access controls, audit logs, and encryption of data in-flight, at-rest, and in-use. Protects user privacy and corporate IP without losing the ability to query data.	
Unique Capabilities	
Developers	**Analytics Teams**
Low latency, continuous processing of live operational data in-motion (streaming) and at-rest (database) within the application	High-throughput processing of thousands of records using existing SQL-based tools and skills – without impacting the application
Searching and retrieving fresh application data to prompt LLMs for generative AI and semantic search	Query and blend live data with archived application data, without ETLing it back to centralized systems
Support data structures optimized for the ingest, storage, and analysis of high velocity time series measurements and event streams	Support data structures optimized for integration into ML model training and OLAP engines

Approaches to delivering app-driven intelligence

All of the hyperscalers offer an array of data management services that could collectively meet the required capabilities needed for application-driven intelligence. But these are isolated primitives, rather than integrated platforms. This means your teams have a lot of custom engineering ahead of them, having to:

1. Stitch together multiple databases to handle the different data requirements common in operational apps (i.e., documents, tables, vectors, time series

measurements, key-value pairs, graphs), each accessed with its own unique query API.
2. Build ETL data pipelines to move and transform data between different databases and tiers of data storage.
3. Spin up a federated query engine to work across each data tier, using its own unique query API.
4. Integrate serverless functions to react to real-time data changes (inference or vector encodings to retrieve the latest context for model prompting).
5. Stand up their own API layers to expose data to consuming applications.

All of this complexity places enormous overhead on your teams. It results in fragmented and inefficient developer experiences, a multitude of operational and security models to deal with, a ton of data integration work, and lots of data duplication. It slows your time to market, while increasing your costs and risk. And now you are now deeply locked into that hyperscalers entire stack – from low level infrastructure to high level software, AI, and analytics services. This lock-in restricts your freedom and drives up your switching costs should you wish to move to another provider in the future.

You could try to mitigate lock-in by assembling all of the data management components from a suite of best-of-breed technology vendors. But now the fragmentation and sprawl is magnified across the different technologies and vendors you choose.

The complexity of either hyperscaler or self-assembly approaches can risk deterring teams from even trying to embrace application-driven intelligence. This leaves the business disadvantaged, unable to compete with more agile and innovative competitors.

But MongoDB gives you a better way.

How MongoDB helps

MongoDB's developer data platform, built on MongoDB Atlas, unifies operational, AI, and analytical data services to streamline building AI-enriched applications.

Atlas puts powerful AI and analytics capabilities directly into the hands of developers in ways that fit their workflows, frameworks, and languages. With Atlas they land data of any structure, index, query, search, and analyze it to provide model context and inference in any way the app needs, and then archive prompts and reasoning steps for long term

model memory. All while working with a unified API and flexible data model, without having to build their own data pipelines or duplicate data.

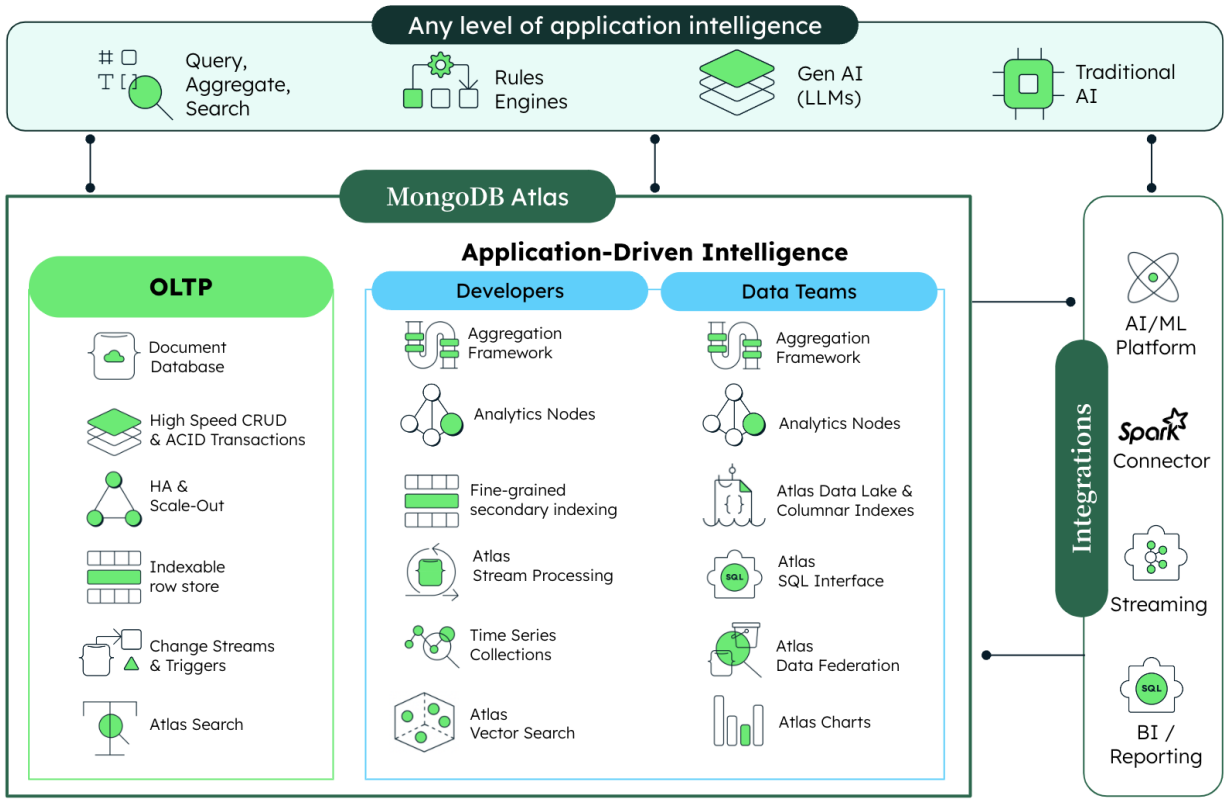

Figure 1: *MongoDB Atlas unifies transactional, AI, and analytical processing with a unified, developer-native API and flexible document data model in a multi-cloud data platform.*

At the same time, analytics teams get access to live application data using their preferred SQL-based tools. They can work with this fresh data without interfering with the application, and have the ability to generate insights for business owners and feed training data into their custom models.

Atlas supports any level of app intelligence – from querying records and retrieving information with semantic and keyword search to aggregating and transforming data, feeding rules-based engines and feature stores through to augmenting LLMs and traditional AI models with live data. Atlas automatically optimizes how data is ingested, processed, and stored, maximizing the efficiency of the application's operational, AI and analytical workloads. These capabilities are packaged in an elegant and integrated multi-cloud data architecture.

MongoDB Atlas runs on over 110 regions on AWS, Azure, and Google Cloud. This provides you the freedom to run your AI workloads wherever you need. You can co-locate your apps close to users for data sovereignty and latency, or have the flexibility to harness the latest AI innovations from any of the hyperscalers while Atlas handles all of the data movement for you.

Enabling Application-Driven Intelligence

However you are harnessing AI - from embedding the latest generative AI in your apps to training and serving your own machine learning models. Atlas gets you to app intelligence faster. From prototype to enterprise-ready, you can ensure your apps are grounded in truth with the most up-to-date operational data, while meeting the scale, security, and performance users expect.

Key capabilities include the following.

Unified operational, AI, analytical, and streaming data services
Simplifies the AI data lifecycle with a single data model and single query API on top of a highly scalable and secure multi-cloud platform that works with data in-motion and data at-rest.

Flexible document data model
Developers value the document model because it maps to objects in code and can be modified on-demand. This allows them to continuously innovate and experiment with new parameters and data types.

Documents can model data of any structure – from the vast diversity of regular application data and model features to vector embeddings composed of several thousands of dimensions. Any of these structures can be modified at any time to support the addition of new data types and application or model features without lengthy upfront schema design or painful migrations.

In a world where your competitors have access to many of the same models as you, your own proprietary data is key to your market differentiation. Documents give you the flexibility to rationalize and harness that data in ways that traditional tabular data models simply can’t.

Optimized data storage and tiering
Achieve low-latency for online inference stores and knowledge retrieval along with high throughput for offline feature stores – all delivered in a single platform.

Atlas integrates data tiering, query federation, and row and column indexing with a horizontally scalable, operational database. Dedicated analytics nodes isolate long running, complex queries from your operational database, ensuring each service gets the resources it needs.

Expressive, developer-native API
Enhance productivity across developers and ML/AI teams with a single, expressive query API that simplifies the workflow from data preparation, through model training, inferencing, and knowledge retrieval, to serving applications.

From simple CRUD operations to keyword and vector similarity search through to sophisticated aggregation pipelines for analytics and stream processing, the Query API provides developers the flexibility to query and compute data anyway the application needs. This avoids the productivity impact of developers having to constantly context switch between different query languages and drivers, while also keeping your technology footprint compact and agile.

Native vector and keyword search
Accelerate augmenting applications with generative AI by leveraging natively integrated vector search (preview) and keyword search without additional infrastructure to provision, sync, secure or manage.

Extracting insights from the firehose of streaming and time series data
Whether you're building reactive customer experiences or continuous analytics, Atlas Stream Processing (preview) enables processing of high-velocity and unbounded streams of complex, in-motion data using the same data model and query interface that's used for your database.

With MongoDB's time series collections, data is automatically columnarized and compressed for storage, I/O, and analytics processing efficiency across IoT, clickstream, and logging data.

Extensive ecosystem of integrations
Simplify building AI-enriched applications with integrations to an extensive AI partner ecosystem spanning leading MLOps platforms to open source LLMs and frameworks.

Services like Atlas Triggers can be used to programmatically call embedding services or model inference as soon as data changes, using a fully reactive, event-driven architecture.

Further integration is provided by the Spark Connector and Kafka Connector. The Atlas SQL Interface allows data teams to connect their preferred tools to MongoDB, and with the PyMongoArrow extension they can load MongoDB query result sets formats suitable for popular data science & ML libraries like Scikit-learn, Pandas and Numpy

Learn more
Review MongoDB's AI/ML resource center to view all of the latest best practices for app-driven intelligence, and download our Generative AI whitepaper to learn how MongoDB Atlas can be used to harness the power of LLMs in your apps.

Industry Analyst Research into Application-Driven Intelligence

Forrester has conducted its own research into platforms capable of integrating operational, analytical and stream processing. Evaluating 15 vendors across 26 criteria, the Forrester Wave™: Translytical Data Platforms, Q4 2022 named MongoDB as a Leader, citing:

"Overall, MongoDB is good for customers that are driving their strategy around developers who are tasked with building analytics into their applications."

You can access your complimentary copy of the Forrester report here.

In its Analyst Perspective, Ventana Research recommended that *"organizations evaluating potential database providers for new, intelligent operational applications include MongoDB Atlas in their evaluations."*

Positioning with Centralized Analytics

Application-driven intelligence powered by MongoDB complements existing analytics processes built around centralized data warehouses and data lakehouses. It **does not** replace them.

By mapping required capabilities to use cases, you will see these different classes of AI and analytics serve different purposes.

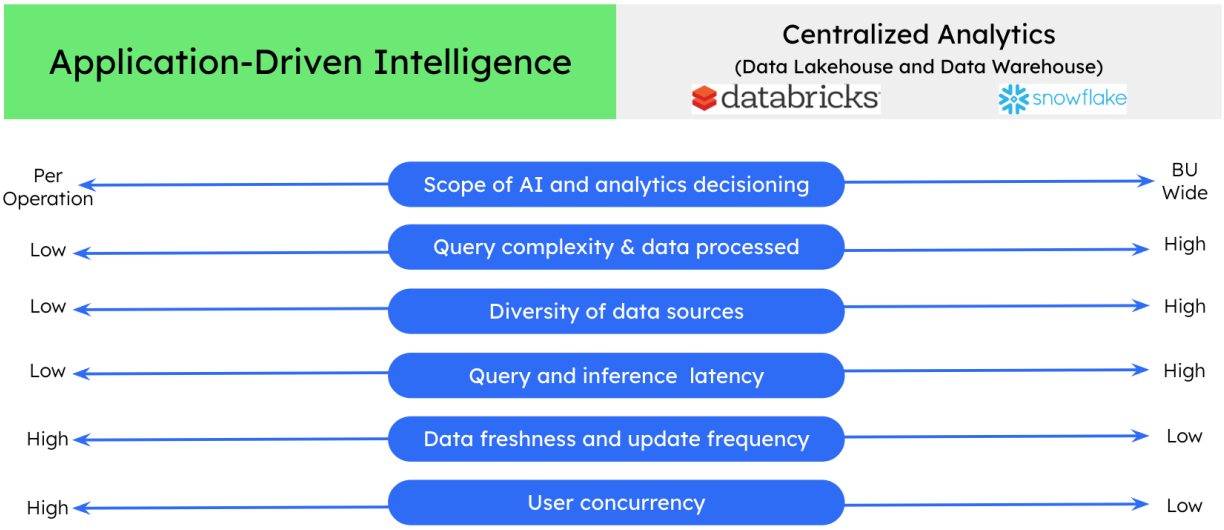

Figure 2 *identifies required capabilities across a spectrum of different classes of AI and analytics processing.*

Based on a number of real world MongoDB deployments, consider the need to create hyper-personalized customer offers in a grocery e-commerce system. The centralized data warehouse or data lake runs a weekly analysis of the vast troves of first and third party customer data ingested from systems across the company. This analysis is blended with available inventory to create a basket of 20-30 potential offers, uniquely personalized for each customer.

These offers are then loaded back into our customer database powered by MongoDB, where each customer profile is enriched with offers for that week. When the customer next visits the site, or when the marketing team runs its weekly email promotion, only the most relevant 3-5 offers are displayed.

Application-driven intelligence powered by MongoDB infers which offers are most relevant to the customer based on a set of real-time criteria. For example, actual stock availability and which items a shopper might already have added to their basket or have purchased in the past couple of days. This real-time decisioning is important as you do not want to serve an offer on a product that can no longer be fulfilled or on an item a customer has already decided to buy. For this process to work, inferences need to be made against the latest data and returned to the app in under a second.

This example demonstrates why it is *essential* to choose the right tool for the job. Specifically, in order to build a portfolio of potential offers, the centralized data warehouse or data lake is an ideal fit. Such technologies can quickly scan and process hundreds of TBs of customer records, purchase history and inventory data in a single complex query, feeding the results into a model that creates the offers in offline batches.

However, the same technologies are completely inappropriate when it comes to predicting which specific offers are most relevant to customers in *real time* while they are browsing the ecommerce store. Centralized analytics systems are *not designed* to serve thousands of concurrent user sessions. Nor can they access real time inventory or basket data to make low-latency predictions in milliseconds. Instead, for these scenarios, application-driven intelligence powered by MongoDB is the right technology fit.

Application-Driven Intelligence in Action

MongoDB counts 43,000+ customers globally across all industry sectors and sizes, including more than 50% of the Fortune 100. Many customers started out using MongoDB as an operational database for both new cloud-native applications as well as modernized legacy apps.

More and more of these customers are now building new applications and enriching existing ones with application-driven intelligence powered by MongoDB.

As Table 3 below shows, multiple industries and use cases benefit from app-driven analytics powered by MongoDB.

Customer	Use Case	Results
Telefonica	Digital transformation with cross-industry AI and IoT solutions over 5G and future 6G networks	MongoDB provides app-driven intelligence for 10s of millions of devices today supporting 150,000 operations per second across 30TB of data
Rent the Runway	Feature store for AI and robotics warehouse automation	Reduced warehouse processing times by 67%, increasing garment availability for rental. Provides real time analytics for the business
Continental	Autonomous driving safety systems – braking and tyres	*"In the end we were able to tame this deep learning beast with this flexible database"* Deep Learning SME, Continental
Marks and Spencer	Customer loyalty application – moving from points to offers-based rewards	Reduced time to build 1m offers from 1 hour to 5 minutes, served at 10x lower latency. Program is driving 8x higher customer spend
Bosch	IoT: sensor data collection and analysis for connected vehicles	*"From my history [of doing this for] 22 years we never had the capabilities to do this before."* Senior technology evangelist Bosch Global Software
Iron Mountain	Intelligent data processing platform using AI/ML + MongoDB to transform unstructured data for enrichment and analytics	Time series analytics supports workflow automation, data lineage, building knowledge graphs, and reporting
Toyota Financial Services	Fraud detection, customer onboarding, federates mainframe data into web and mobile services	*"MongoDB helps us make better decisions and build better products."* Division Information Officer, Toyota

Getting Started

Application-driven intelligence is defining the next wave of successful modern applications. Developers need to work with data and AI/ML models in ways that were previously the domain of dedicated analytics and data science teams. At the same time, those same data teams will need direct access to source operational data in order to create fresher models and unlock real-time business visibility.

The MongoDB Atlas developer data platform is engineered to help both teams ride this new wave – leading to smarter, more intelligent apps and automated business processes that can react and respond more quickly to fast changing operational data.

The best way for your teams to get started is to sign up for an account on MongoDB Atlas. From there, they can create a free database cluster, load their own data or our sample data sets, and explore what's possible within the platform.

Head to MongoDB's AI/ML resource center to view all of the latest best practices for app-driven intelligence. In addition, the MongoDB Developer Center hosts an array of resources including tutorials, sample code, videos, and documentation organized by programming language and product. MongoDB also offers a range of instructor-led and self-paced training programs for developers and data engineers:

- The recommended path for instructor-led training from MongoDB Professional Services takes attendees through MongoDB fundamentals into data storage and retrieval, advanced query and data processing, and Atlas Data Federation.
- MongoDB University provides online, self-paced training that gets attendees started with MongoDB, data modeling, querying and the aggregation framework, indexing, performance tuning, and more.

In addition to training, MongoDB also provides a range of consulting services that can work with your teams at any stage of your project – from initial design and architecture through to optimizing applications already running in production.

Collectively, these resources and services will help you better meet user expectations with smarter apps while driving better decisions and actions with AI and analytics.

Safe Harbor

The development, release, and timing of any features or functionality described for our products remains at our sole discretion. This information is merely intended to outline our general product direction and it should not be relied on in making a purchasing decision nor is this a commitment, promise or legal obligation to deliver any material, code, or functionality.

US 866-237-8815 • INTL +1-650-440-4474 • info@mongodb.com.