

The Cloud Migration Solution: MongoDB Atlas on Azure

The Cloud Migration Solution: MongoDB Atlas on Azure

IT decision-makers (ITDMs) are feeling the heat. The pressure's on to modernize, and keep pace with relentless innovation—all while budgets shrink and legacy systems eat away at already-limited resources. It's a constant scramble to demonstrate ROI, wrangle complex data, and launch applications that embrace the latest innovations, including AI.

Maintaining on-premise database solutions often exacerbates these challenges. For ITDMs who want to make the leap to the cloud without jeopardizing critical data, MongoDB Atlas on Azure is the solution.

Migrating from on-premise to the cloud with MongoDB Atlas on Azure gives you:

Bandwidth to focus on strategic projects, application development, and innovation rather than time-consuming infrastructure management.

Comprehensive data protection that allows for easy recovery, secure access, automated backups, and simplified compliance with industry regulations.

Significant reduction in total cost of ownership by eliminating the need for upfront hardware investments or ongoing maintenance.

True alignment with organizational goals for adopting cloud-based services, leading to reduced operational overhead, increased agility, and flexible growth.

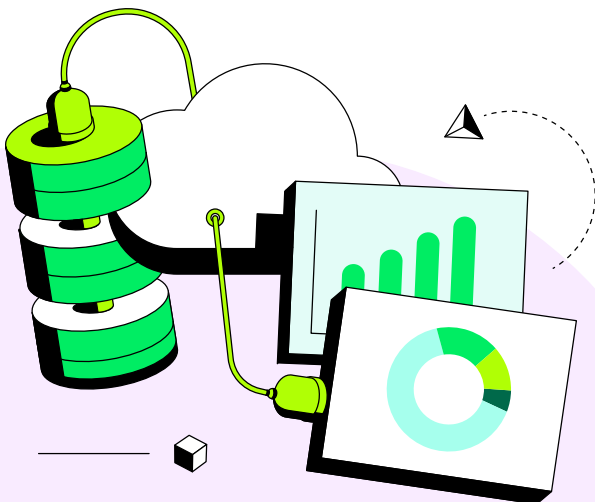
Transitioning MongoDB workloads (Community or Enterprise Advanced) from on-premise to MongoDB Atlas on Azure provides a pathway for ITDMs to modernize their existing systems, optimize their database environment, and unlock the full potential of their data.

Why MongoDB Atlas?

MongoDB Atlas on Azure combines MongoDB's robust document data platform with Azure's scalability and advanced cloud services, making it ideal for high-performance applications. By migrating, IT leaders won't just capitalize on all MongoDB has to offer, they'll also simplify the totality of their database operations.

MongoDB Atlas provides a fully managed service, which means it handles all the underlying infrastructure management tasks, such as provisioning, scaling, patching, and backups.

This frees ITDMs and their teams to focus on higher-value activities, including attracting and retaining great talent.



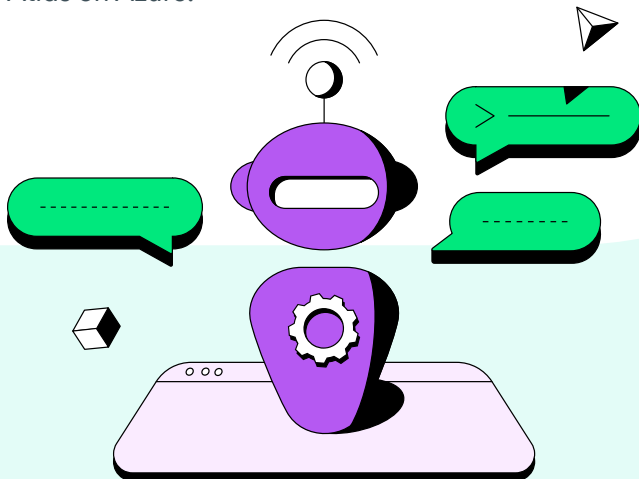
The biggest benefits of migration include:

- **Improved security** with Azure's robust security infrastructure. MongoDB Atlas integrates with Azure security services, including Microsoft Entra ID and Azure Key Vault, to provide enhanced data protection.
- **Total flexibility** to easily scale database deployments up or down based on business and data needs, ensuring optimal performance even during peak demand.
- **Pay-as-you-go pricing** that allows users to only pay for the resources they need, helping to optimize cloud costs and align with an ideal ROI.
- **Faster application development** and deployment by using MongoDB Atlas on Azure to quickly deploy and manage MongoDB databases.
- **Low-latency data access** for users around the world, ensuring a great experience for application users, and assuring data sovereignty via Azure's global presence.

Deep Integrations, Deeper Expertise

MongoDB Atlas is deeply integrated with the Microsoft Intelligence Data Platform. This integration empowers technical leaders to quickly build applications backed by enterprise data and powered by Microsoft Azure's AI, analytics, and security tools and services. Enterprises can seamlessly connect to Azure AI Foundry, Microsoft Fabric, Power BI, and Synapse Analytics for a streamlined workflow.

For migrations, dedicated experts and programs from Microsoft and MongoDB also guide the migration to Azure. This ensures a secure and optimized process that helps leaders access the full potential of MongoDB Atlas on Azure.



The Atlas Advantage: Simple, Secure, Scalable

MongoDB Atlas on Azure delivers a unified platform that simplifies data management and scaling. This streamlined approach reduces complexity and minimizes infrastructure sprawl, allowing for efficient operational oversight.

Enhanced Security

Built-in support for Microsoft Purview, Microsoft Entra ID, and Azure Key Vault ensures robust data protection and governance across both the Microsoft Azure and MongoDB Atlas platforms. This helps organizations:

- Meet compliance requirements
- Safeguard sensitive information
- Maintain data integrity

Increased Agility and Scalability

With Atlas on Azure, development teams can accelerate innovation and respond swiftly to evolving business needs, ensuring organizations remain competitive and can readily adapt to market demands.

Improved Developer Experience

MongoDB Atlas provides a fully managed developer data platform with a unified interface and API. This streamlines development workflows, boosts productivity, and enables developers to focus on building high-quality applications, ultimately improving time-to-market.

Additionally, MongoDB Atlas is integrated with Microsoft SDKs such as Semantic Kernel and Entity Framework. With these integrations, developers can build generative AI-powered applications and work with data using familiar object-oriented concepts instead of writing complex SQL queries.

Simplified Migration with Minimal Downtime

MongoDB Atlas offers a range of migration options, including fully managed methods such as Atlas Live Migrate, minimizing the operational burden on IT teams. This allows ITDMs to choose the migration approach that best fits their needs, while reducing application downtime.

Unified Search Capabilities

MongoDB Atlas has full-text and semantic search as a native capability of the database. As a result, there's no need to stand up and manage a separate sync mechanism. Avoid the pain of stale data, duplicating information in search indexes, or writing custom transformation logic by building search directly in MongoDB Atlas, leading to a 4x increase in acceleration of search deployment.

Migration Made Simple

Migration to MongoDB Atlas on Azure involves transferring data from an existing MongoDB deployment to the Atlas platform. There are several migration approaches to suit different needs and technical comfort levels. These include fully managed methods (Atlas Live Migrate “pull” and “push”) where MongoDB handles the process, or several self-managed methods.

While the Atlas Live Migrate is usually the most straightforward method, the best migration option for an individual organization depends on:

- **Dataset size**
- **Downtime tolerance**
- **Automation**
- **Security requirements**

Atlas Live Migrate (“pull” and “push” methods):

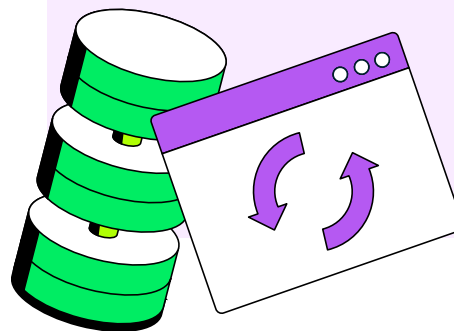
This fully-managed option is often the preferred choice. Users only need to provide database credentials and connectivity to their source cluster, simplifying migration. Here are the differences between the two methods:

Pull

Requires database user credentials and network connectivity to your existing cluster. Atlas handles the rest.

Push

Requires your existing cluster to be enrolled in MongoDB Cloud Manager.



The Migration Process

Regardless of the chosen method, the core process of migration involves transferring data from its current location to a new MongoDB Atlas cluster on Azure. By understanding the available migration methods and planning carefully, ITDMs can confidently move their MongoDB deployment to MongoDB Atlas on Azure. This process includes:

Step 1: Assess. Evaluate existing deployments, including data size, application dependencies, and security considerations.

Step 2: Method Selection. Choose the migration method that best aligns with requirements.

Step 3: Prepare. Configure the necessary network connectivity, security settings, and user permissions.

Step 4: Migrate. Execute the chosen migration method, monitoring the process for any issues.

Step 5: Validate. After migration, thoroughly test applications and data to ensure everything functions correctly in the new environment.

Microsoft Cloud Accelerate Factory

Microsoft and MongoDB have partnered to make it easier for businesses to move their MongoDB databases to the Azure cloud. This is done through the **Microsoft Cloud Accelerate Factory (CAF)**, a free program that provides the tools, resources, and expertise needed for a smooth transition.

Microsoft CAF is also partnered with MongoDB Migration Factory, a structured program that streamlines database migrations to MongoDB using proven tools, best practices, and expert guidance.

Microsoft CAF and MongoDB Migration Factory jointly deliver migrations of MongoDB Enterprise Advanced or Community Edition deployments to MongoDB Atlas on Azure in a secure, optimized, and customer-focused way. By addressing unique aspects of migration, both services provide valuable resources to minimize risk and accelerate modernization efforts.

Microsoft and MongoDB offer:

Simplified Migration. They handle every aspect of migration to ensure a smooth transition to MongoDB Atlas on Azure.

Cost Savings. Microsoft CAF is a free service, allowing organizations to focus on the migration without budget concerns.

Reduced Risk. The partnership between Microsoft CAF and MongoDB Migration Factory, as well as the structured CAF approach, minimizes technical risks and disruptions.

Comprehensive Support. CAF provides end-to-end guidance, including live and offline migration options for minimal downtime throughout the entire migration process.

Focus on Innovation. By simplifying database management and operations, and by enhancing performance, organizations can focus on innovation and extracting value from their data, rather than ongoing maintenance.

Making the Cloud a Reality

The migration from on-premise infrastructure to the cloud is no longer a question of “if,” but “when.” MongoDB on Microsoft Atlas makes this transition both necessary and effortless. With flexible migration options like Atlas Live Migrate and the support of Microsoft Cloud Accelerate Factory, moving to the cloud has never been easier.

To learn more, review, [the Azure Migration Guide](#)

If you're ready to get started, contact MongoDB at azure@mongodb.com to take advantage of this free migration opportunity and experience the ease of MongoDB Atlas on Azure with Microsoft CAF support.

