Data Resilience Strategy with MongoDB Atlas

MongoDB.

Table of Contents

Introduction

Data is the core currency in today's digital economy. Yet, according to industry research, 43% of companies that experience major data loss incidents are unable to resume business operations.

Data loss incidents can take a variety of forms, but when it comes to database technology, they generally fall into three categories: catastrophic technical failure, human error, and cyber attacks.

- Catastrophic failure includes natural disasters and any other scenario that permanently destroys all the servers in your production system. If you keep all your servers in the same data center, a fire or flood that destroys them would qualify. This is typically the type of failure for which we implement a backup strategy.
- Secondly, while they may be less newsworthy than catastrophic failure, the reality is that human error or flawed processes account for the vast majority of IT outages. Humans introduce application bugs or accidentally delete data. A bad code release that corrupts some or all of the production data is an unfortunate but common example. In the case of human error, the errors introduced can propagate automatically to all of the nodes in your database cluster, often within seconds.
- And the third category is cyber attacks. Usually cyber attacks involve a combination of malicious actors and ransomware, where the nature of attacks is now moving from just locking systems and holding the data for ransom to deliberately hacking into systems to steal the data and sell it to third parties. Another attack vector as companies become more dependent on third party vendors, is increasing vulnerability through the IT supply chain. This vulnerability can include a simple act of downloading a software patch from a trusted IT vendor that unknowingly includes a threat in the software.

Depending on the industry that you are in, there can even be regulatory frameworks designed to implement safeguard against such cyber attacks that you must comply with. This compliance can be the driving factor in how you structure your disaster recovery strategy. In today's climate, these regulatory frameworks and regulations instituted by governments and regulating bodies are becoming more stringent with increasingly strict guidelines. As an example in the financial industry, as more financial institutions depend on third-party service providers to perform or support critical operations they are expected to adhere to Appendix J: Strengthening the Resilience of Outsourced Technology. Appendix J outlines some of the key elements, such as planning and testing for significant disruptions and cyber events, to ensure a strong Business Continuity plan.

Another example is the Digital Operational Resilience Act (DORA). It is a proposed legislation by the European Union (EU) aimed at strengthening the operational resilience of the financial sector in the digital age. The DORA is intended to update the existing regulatory framework for the financial sector by addressing the challenges posed by digitalization, cyber threats, and other operational risks. Another important policy that many industries are expected to adhere to is the The NIST Cybersecurity Framework which is based on five core pillars or functions: identify, protect, detect, respond and recover. This is published by the US National Institute of Standards and Technology with the goal of providing standards, guidelines and best practices to manage cybersecurity risk. While determining whether a specific framework is considered mandatory or voluntary is dependent on the industry, or even company, regulations, it is clear that these frameworks are only going to continue to become more important over time.

In addition to the need for data resilience as outlined above, properly setting up backups can also enable cloud-forward agility and help improve developer productivity. Modern application teams need to move super fast all while still ensuring mission-critical safeguards are still in place. Backups can be used to create new environments for development, staging, or QA without impacting production (e.g You can quickly hydrate a new sandbox environment with data by restoring a backup from your production environment). This practice enables development teams to quickly and easily test new features, accelerating application development and ensuring smooth product launches.

Experiencing some level of data loss from a catastrophic failure, human error, or cyber attack is not a question of if it is going to happen, but rather a question of when. Therefore, the preparation and focus should be on how you can prevent and mitigate the impact from that as much as possible. If companies do not mitigate these risks properly, they can become un-operational for several days at a time due to an event.

For all of the reasons mentioned above, a strong disaster recovery and data resilience policy become very critical and from a database standpoint, high availability and backups play a critical role. Your strategy and regulations could drive a complete need to re-architect how your database clusters are distributed, how often you take backups, where your backups live, and how you can guarantee their availability. MongoDB Atlas can help you achieve all of these things easily.

Below we'll outline disaster recovery considerations and then specific recommended approaches for cluster distribution and backups in MongoDB Atlas.

Disaster Recovery Considerations

Disaster Recovery is a game of trade-offs that balances the availability profile that you require and how much you are willing to spend on it. There is no one size that fits all and hence we have a lot of options within Atlas for an in-depth approach. We have sensible defaults that mean you are protected by default, but also give you an array of options to precisely match your requirements for each specific application.

In preparing a disaster recovery strategy, organizations typically start by evaluating their recovery point objective (RPO) and recovery time objective (RTO). The RPO indicates how much data the business is willing to lose in the event of an incident, while the RTO indicates how quickly it will recover. As not all data is created equal, this RPO and RTO should be evaluated on an application-by-application basis. For example, you'll likely have different requirements for your mission-critical customer data than you will for your clickstream analytics. The requirements for RTO, RPO, and the retention period for which you need to keep your backups will drive the economic and performance cost of maintaining backups.

Now when we look at MongoDB Atlas, we offer common protection safeguards out of the box that will meet the demands of many applications. These are complemented by additional capabilities that can be used to provide even higher levels of availability and resilience for your most critical workloads. These capabilities can be bucketed into two phases: prevention and remediation.

- **Prevention:** Features that will help you plan and prevent data loss risks due to disasters. Includes features such as multi-region and multi-cloud clusters, encryption at rest, queryable encryption, cluster termination protection, backup compliance policy, and the ability to test resilience.
- **Remediation/Recovery:** Features that will help ensure your recovery time objectives are met and ensure minimal data loss. Some out-of-the-box features are: Continuous cloud backups with point in time recovery, automated backup schedule and retention, and multiple region snapshot distribution.

Prevention

The best strategy for any type of disaster recovery is to eliminate as much risk as possible of a disaster happening in the first place. This applies to cloud provider or hardware outages, ransomware attacks, and human errors.

Prevention methods for a cloud provider or hardware outages

Out of the box, MongoDB Atlas offers full resilience to the failure of a single node, as well as partial resilience (read availability) to two node and cloud region outages. This resilience is provided by a minimum of three data nodes per replica set that are automatically deployed across availability zones (AWS), availability zones/fault domains (Azure), or zones (GCP). Distributing a deployment across data centers creates multiple points of failure such that one node can have a failure without having any effect on the other two nodes. This provides continuous application uptime in the event of partial cloud outages, and routine hardware or software maintenance.

While this may meet the resilience that some applications require, other applications require more resilience to failure depending on the level of data resilience and disaster recovery you are aiming for. For example: the need for data to reside in a particular geography, and the level of uptime expected. In order to prevent more severe outages, up to an outage of your entire primary cloud provider, from affecting your application, MongoDB Atlas offers multi-region and multi-cloud deployments. Multi-region clusters deploy cluster nodes across multiple cloud regions, while multi-cloud deploy cluster nodes across multiple cloud providers. With Atlas, you can choose from 100+ cloud regions across the 3 major cloud providers Google Cloud Platform, AWS, and Microsoft Azure.

- **Across Cloud Regions:** With multi-region clusters, cluster nodes are deployed across more than one cloud region. Cross-region replication allows the database cluster to automatically fail over to a different region during incidents affecting an entire region. As the nodes are spread out across multiple regions but within the same cloud provider, this ensures availability during a full region outage.
- **Across Cloud Providers:** With multi-cloud clusters, cluster nodes are deployed across multiple cloud provider regions. This allows automatic failover to a different cloud provider serving in the same or different geographical location. Utilizing a different cloud provider in the same geographical location preserves low-latency data access and regulatory compliance when your primary cloud provider is experiencing issues. This is the best-in-class availability option, as this ensures availability even in case of a cloud provider outage as the nodes in the replica sets are divided across multiple cloud providers such as AWS, GCP, and Azure.

While it is important to have a proper backup strategy, with a multi-region and multi-cloud deployment, Atlas will protect the customer experience with automated failover without any application downtime or recovery needed.

Prevention methods for ransomware attacks and human errors

By default, MongoDB Atlas requires Transport Layer Security (TLS) 1.2+ encryption for all client data from application client to server and intra-cluster network communications, while also having built-in encryption at rest for all cluster storage and snapshot volumes by default with every node in your cluster.

Depending on your use case, you may want to enable additional layers of security for the data stored in your cluster. On top of the default encryption provided, you can also

- Enable Encryption At Rest using Customer Key Management from the WiredTiger storage engine to encrypt the data files themselves. It can work with a cloud provider of your choice for your project: Amazon Web Services Key Management Service, Azure Key Vault, or Google Cloud Platform Key Management Service. This also means that you can utilize one cloud provider for your cluster nodes and a different cloud provider for your key management.
  - When used in conjunction with transport encryption and good security policies that protect relevant accounts, passwords, and encryption keys, Encryption at Rest can support compliance with security and privacy standards, including HIPAA, PCI-DSS, and FERPA.
- You can also enable one of MongoDB's In-Use Encryption features, either Client-Side Field Level Encryption (CSFLE) or Queryable Encryption which allows you to encrypt sensitive data in your application before you send it over the network to MongoDB databases. With CSFLE enabled, no MongoDB database or cloud provider, including employees, has access to your sensitive data in an unencrypted form. These features give you the ability to perform the following tasks which are all completed without the server having knowledge of the data it's processing:
  - Encrypt sensitive data fields from the client-side
  - Store sensitive data fields as randomly encrypted data on the database server-side.
  - Run expressive queries directly on the encrypted data, without needing to ever decrypt it server-side.
- You want to make sure that only trusted IP addresses can connect to your cluster and access to your data. This is possible through the IP access list feature. Once the IP access list is configured, Atlas only allows client connections to the database deployment from entries in the project's IP access list.
- Another important aspect is to ensure you can test your clusters and applications to ensure they can be ready for a real incident. With Test Resilience in Atlas, you will be able to test the failure of a single node up to a regional failure at the click of a button to ensure you're ready for a real-life disaster event. This type of testing across regions will be paramount for making sure your mission-critical applications remain fully functional after an unexpected outage. Testing and ensuring your cluster's resiliency is working as you expect is no longer a one-time-a-year test but now just like your CI/CD process where you can continuously test your disaster recovery throughout the year at any time.

In addition to securing the data itself, you can also secure your clusters against human errors or malicious activities. Human errors are common but development teams need to move more quickly now than ever before. They need to quickly and easily test new features, accelerate application development and ensure smooth product launches. In addition to errors, malicious attacks are more prevalent than ever and attackers can look for any way to disrupt operations which could mean deleting clusters, data, or backups. In order to protect against both of these types of threats, MongoDB Atlas offers additional safeguards depending on your business requirements and concerns.

- You can enable Termination Protection for clusters in Atlas, to ensure the prevention of accidental deletion of your production clusters and irretrievable loss of data. With this, any human error that attempts to delete a cluster via the Admin API, IaC integrations, and the UI will be blocked, and you will be prompted to first disable termination protection before deleting the cluster.
- In addition to protecting your clusters from deletion, you can also protect all of your backups as well. The Backup Compliance Policy enables organizations to further secure business-critical data by preventing all snapshots and oplogs stored in Atlas from being modified or deleted for a predefined retention period by any user, regardless of Atlas role, guaranteeing that your backups are fully WORM compliant. Only a designated, authorized point of contact can turn off this protection after completing a verification process with MongoDB support.

  – This satisfies data protection requirements while also ensuring Recovery Point Objectives (RPOs) and Recovery Time Objectives (RTOs) are achievable.

Remediation and Recovery

MongoDB offers fully managed backups of your data, including point-in-time data recovery and consistent, cluster-wide snapshots of all clusters, including sharded clusters. Atlas Cloud Backups provide backup storage using the native snapshot functionality of the cluster's cloud service provider and are stored separately from your MongoDB Atlas instances.

- Backups are snapshots that encapsulate the state of your database cluster at a given time. Backups provide a safety measure in the event of data loss or corruption. With an M10+ cluster, you can get access to Atlas Cloud Backups which provides localized backup storage using the native snapshot functionality of the cluster's cloud service provider. Some of the benefits include:

  – Strong default backup retention schedules which provide 12 month retention out of the box.

  – You have full flexibility to customize snapshot and retention schedules, including the time of day to create snapshots, the frequency at which snapshots are taken over time, and retention duration.

  – For example, you could set an hourly frequency for recovery purposes, as well as a weekly or monthly frequency for use cases such as longer term retention of your data at a specific time as regulated by some industries.

  - Another important feature is Continuous Cloud Backup which provides Point In Time (PIT) recovery. This allows you to recover back to a specific minute during the restore process by backing up the oplog, capturing data changes between snapshots. This allows you to recover your data to the exact moment (a point in time) right before any failure or event, like a cyber attack.
  - So for example, you may have set your backup schedule to take a snapshot at 11 am, with the next snapshot set to be taken at noon. With this feature, if you need to restore your data to 11:30 am, Atlas will restore the most recent snapshot from before the desired point in time and then replay the oplog changes to restore to that particular point.
  - You can set a customized restore window to dictate how many days you would like to be able to restore back to a specific point in time.
- By adding multiple region snapshot distribution with Atlas Cloud backups, you can also increase resilience by distributing backup snapshots and oplogs across geographic regions instead of just storing them in your primary region.
  - This allows you to meet compliance requirements of storing backups in different, air gapped geographical locations to ensure disaster recovery in case of regional outages.
  - This also can enable faster, optimized restores (reducing system recovery time) while maintaining the ability to still recover to a specific point.
- During a regional outage, depending on how you have configured your cluster, your cluster might not have a primary node if the regional outage affects a majority of the cluster's electable nodes. In those scenarios, you can use the Atlas UI and add additional database nodes to surviving regions to restore your Atlas multi-region cluster.
- When properly configuring, testing has shown that Atlas can quickly recover to the exact timestamp before a disaster/failure event, giving you a zero minute Recovery Point Objective (RPO) and an Recovery Time Objective (RTO) of less than 15 minutes when utilizing optimized restores, even in the event of the outage of your primary region. Recovery times can vary due to cloud provider disk warming and which point in time you are restoring to so it is important to also test this in your normal testing.

This means that regardless of your regulatory or business requirements, MongoDB Atlas allows you to configure your backups to ensure that you can meet your requirements and, most importantly, recover with precision and speed to ensure that your data loss is minimal and your recovery point objectives are met should you experience a recovery event.

How To Configure Your Data Resilience Strategy

So with a variety of ways to protect your cluster from an outage due to a failure, human error or ransomware attack, and multiple possible backup configurations to ensure minimal data loss and quick recovery time should you need to recover your data, how do you decide what is right for your company? Let's look at a hypothetical scenario with specific requirements and how you can use the features Atlas offers to meet each specification.

Example Company Requirements

Our example company is in the financial industry and has the following requirements:

1. **Ability to remain fully operational in the event of a full cloud provider region outage**
2. **Encryption of all Personally Identifiable Information (PII)**
3. **Recovery Point Objective (RPO) of 1 minute, Recovery Time Objective (RTO) of 15 minutes in the event a recovery is needed**
4. **Backups must be retained for 7 years and be immutable (unable to be edited, modified, or deleted):** Must be retained in multiple locations, including a long term archival located offsite

Example Data Resilience Configuration

1. **Ability to remain fully operational in the event of a full cloud provider region outage:** Let's start with the first requirement - Remaining fully operational in the event of a full cloud provider region outage. By default in Atlas, all clusters are at minimum a three node replica set. This means that if a primary node becomes unavailable, MongoDB will automatically perform a failover to ensure no interruption to your read and write operations. Since MongoDB is a distributed database by design, the failover will happen in seconds without need for manual intervention. In the meantime, Atlas will automatically heal the impacted data node to ensure that your cluster is not degraded for prolonged periods of time. For the most cluster protection and widespread cloud outage resilience, and continuous read and write availability, we recommend a **2-2-1 node configuration** that spreads nodes across three cloud provider regions with our Multi-Region Data Distribution. This offers the highest availability guarantees as even in the event of a complete cloud provider region outage, Atlas will automatically failover to one of the other two regions and still be fully operational with read and write availability. This configuration eliminates the need for any traditional or manual recovery due to a cloud region outage.
2. **Encryption of all Personally Identifiable Information (PII):** Let's now turn to the protection of Personally Identifiable Information (PII). Out of the box, MongoDB Atlas requires Transport Layer Security (TLS) 1.2+ encryption for all client data from application to server and intra-cluster, while also having built-in encryption at rest as previously mentioned. When working with extra sensitive data like PII, MongoDB Atlas' Client-Side Field Level Encryption (CSFLE) can be leveraged to provide among the strongest levels of data privacy and security for regulated workloads. Client-side encryption dramatically reduces the risk of unauthorized access or disclosure of sensitive data by encrypting the fields before they leave your application, protecting them everywhere: in-motion over the network, in database memory, at-rest in storage and backups, and in system logs. This makes it much easier to meet your modern privacy regulations.

3. **Recovery Point Objective (RPO) of 1 minute, Recovery Time Objective (RTO) of 15 minutes in the event a recovery is needed:** Achieving an RPO of 1 minute and a RTO of 15 minutes or less in the event of a recovery situation is possible in Atlas with our Continuous Cloud Backups. Continuous Cloud Backups delivers a customizable automated backup schedule and offers Point In Time recovery so that you can be as granular as a specific timestamp during your recovery. You have four different snapshot frequencies that you can choose from in Atlas - hourly, daily, weekly, and monthly, each with their own retention. For our example, we will utilize all four frequencies and retain the hourly snapshots for 7 days, the daily for 14 days, the weekly for 8 weeks and the monthly for 7 years to meet the 7 year retention requirement without any extra effort. You are able to specify how many days in the past you would like to be able to restore with Point In Time granularity so we will set up our restore window for 7 days giving Atlas up to a zero minute RPO for this entire period.

   Next, to ensure that Atlas performs an optimized restore, we need to also store copies of the backups located in each region of the cluster. As the Atlas cluster is located in 3 regions, we can utilize multi-region Snapshot Distribution to copy our backups to the two secondary regions. Now when Atlas performs a restore, it will identify the backup copies and intelligently use the copy in each region to restore the cluster as fast as 15 minutes or less. With multi-region Snapshot Distribution, you can also specify which types of snapshot frequencies you would like copied, so we will only copy the hourly and daily snapshots, along with the oplog for Point In Time recovery, to help optimize our backup cost but still be able to recover with any hourly or daily snapshot quickly, should a cloud provider region experience an outage. Once your data is restored, all you need to do is point your application to your new cluster and your application will once again be fully functional with full read and write capabilities. Recovery times can vary due to cloud provider disk warming, the point in time you are restoring to, and how quickly you begin the restore process so it is important to ensure your team understands and tests the restore process.

4. **Backups must be retained for 7 years and be immutable (unable to be edited, modified, or deleted). Must also be retained in multiple locations, including a long term archival located offsite:** Now, let's address the remaining requirements. The backups in Atlas need to be immutable, retained for 7 years, and have multiple copies, including a long term archival located offsite. We have already guaranteed the 7 year retention by retaining the monthly snapshots for 7 years in our automated Continuous Cloud Backup schedule. We also already have backups copied to both of the cluster's secondary regions through our multi-region Snapshot Distribution. Achieving immutable backups may seem complex but in Atlas we have introduced our Backup Compliance Policy to guarantee compliance with a pre-set policy engine that is completely automated at all times. With the Backup Compliance Policy, once a snapshot is taken, no user, regardless of role, can delete a backup or modify the retention period. The policy is set in a similar way as the cluster's backup schedule but this is done at the project level. You can think of this pre-set policy as a minimum backup schedule that all of the clusters in your project must adhere to.

For our example project, we will configure the policy to again have all four frequencies and retain the hourly snapshots for 7 days, the daily for 14 days, the weekly for 8 weeks and the monthly for 7 years. Once configured, the Backup Compliance Policy can not be disabled by any user regardless of role. The only way to disable it is to complete a verification process with MongoDB support. Now that our policy is set on a project level, All backups are automatically protected from deletion or modification as soon as they are taken protecting from accidental human errors or cyber attacks. All new clusters created in the future will also automatically adhere to this pre-set project policy so there is no need to manage or worry about new teams not achieving immutable backups with new clusters and failing to meet the requirements.

Lastly, we need to provide a long term archival copy of the backup offsite. To satisfy this final requirement, we can use Atlas' ability to download your snapshots if your offsite location is on-premise, or if you would like to store in S3 Atlas can export them to your own S3 bucket. The export is provided in a JSON format so that it is future-proof and independent of MongoDB entirely and stored in your own S3 bucket of your choice. At any time, you can import the JSON backup data back into MongoDB if needed. By downloading or exporting your snapshots to S3, you can not only achieve long-term archival of your backups but also you can meet requirements like storing your backups in a completely isolated and air gapped on-premise environment that you control and/or having an exit strategy in the event that a third-party provider is no longer operational.

Atlas Features Needed to Meet Example Company Requirements

So let's recap the Atlas features that we needed to meet the example company's requirements.

- Multi-Region Data Distribution
- Client-Side Field Level Encryption (CSFLE)
- Continuous Cloud Backups
- Multi-Region Snapshot Distribution
- Backup Compliance Policy
- Export Snapshots to S3

Understanding your requirements is key to a strong data resilience strategy. Once we knew what requirements we needed to meet, we just needed to configure Atlas appropriately and Atlas automatically takes care of the rest.

Conclusion

While regulatory and business requirements are constantly changing and cyber attacks are evolving in sophistication and diversity, formulating and enabling a data resilience strategy that provides the protection your company seeks does not have to be complex or tedious.

MongoDB Atlas provides common protection safeguards out of the box that offer a strong data resilience for your database layer that will ensure you can prevent incidents as well as remediate quickly with minimal data loss should an incident occur. Additional advanced data resilience features are easy to set up and manage, as well as automated by the pre-set policy engine that works at any scale. This ensures compliance without manual actions, all in the MongoDB Atlas platform, giving you the confidence that your requirements are met day after day, and year after year. Whether your requirements are driven by NIST, Appendix J, DORA, or industry specific frameworks, your data resilience strategy can be driven all by Atlas without having to manage additional platforms that are separate from where your data resides.

Most of the features mentioned above are available as part of the M10+ tiers of MongoDB Atlas. While some features such as queryable encryption, field-level encryption, and termination protection are available across all tiers.

Learn more

- Multi-cloud data distribution
- Backups
- Encryption
- Snapshot distribution
- Cluster termination protection
- Backup Compliance Policy
- Resilience and High Availability With MongoDB Atlas

Authors:
Darshana Paithankar
Evin Roesle