Aurora Extended Content

Amazon Aurora

Amazon Aurora: Aurora Extended Content

Table of Contents

Amazon Aurora features

For Aurora PostgreSQL and Aurora MySQL

Get started with Aurora

Connect with an Aurora specialist

Amazon Aurora is a fully managed relational database service that combines the performance and availability of high-end commercial databases with the simplicity and cost-effectiveness of open source databases. Aurora is comprised of three engines: PostgreSQL, MySQL, and DSQL. Both Aurora PostgreSQL and Aurora MySQL offer full open source compatibility, allowing existing applications and tools to run without requiring modification.

New AWS customers can get started with Aurora PostgreSQL serverless at no cost through the AWS Free Tier, which provides up to $200 in credits to explore Aurora and other eligible AWS services.

Performance

Topics

Up to 6x the throughput of PostgreSQL and MySQL

Testing on standard benchmarks such as SysBench has shown an increase in throughput of up to 6x over stock PostgreSQL and stock MySQL on similar hardware. Aurora uses a variety of software and hardware techniques to ensure the database engine is able to fully use available compute, memory, and networking. I/O operations use distributed systems techniques, such as quorums, to improve performance consistency.

Optimized reads

Optimized reads for Aurora PostgreSQL delivers up to 30% cost savings compared to instances without it. It is ideal for applications with large datasets that exceed the memory capacity of a database instance.

Optimized reads instances use local NVMe-based SSD block-level storage, available on Graviton-based R6gd and R8gd and Intel-based R6id instances. Performance enhancements include tiered caching and temporary objects to enable you to make the most of your database instances. With up to 8x improved query latency, you can effectively run read-heavy, I/O-intensive workloads such as operational dashboards, anomaly detection, and vector search.

When you use optimized reads instances with pgvector, it increases queries per second for vector search by up to 9x.

Diagnose and resolve performance bottlenecks

Amazon DevOps Guru for RDS uses ML-powered insights to help easily detect and diagnose performance-related database issues and resolve them in minutes rather than days. You can use it to automatically identify the root cause of performance issues and get intelligent recommendations to help address the issue, without needing help from database experts.

To get started, enable CloudWatch Database Insights in the Amazon RDS Management Console, and then enable DevOps Guru for RDS for your Aurora resources or your entire account.

Availability and durability

Topics

- Instance monitoring and repair
- Multi-AZ failover
- Amazon Aurora Global Database
- Fault-tolerant and self-healing storage
- Automatic, continuous, incremental backups and point-in-time restore
- Database snapshots

Instance monitoring and repair

Aurora continuously monitors the health of your databases. In the rare event of a failure, Aurora automatically restarts your database and associated processes. Unlike other databases, Aurora does not require crash recovery replay of database redo logs, which greatly reduces restart times. Aurora also isolates the database buffer cache from database processes, which allows the cache to survive a database restart without brownouts.

Multi-AZ failover

On instance failure, Aurora uses RDS Multi-AZ technology to automate failover to one of up to 15 Aurora read replicas you created in any three Availability Zones (AZs). If no read replicas were provisioned, Aurora will automatically attempt to create a new Aurora DB instance for you. You can reduce failover times further by using open source and drop-in compatible AWS JDBC Driver for PostgreSQL and AWS JDBC Driver for MySQL, or utilize RDS Proxy to decrease failover times by up to 66% while preserving application connections.

Amazon Aurora Global Database

Aurora Global Database allows a single Aurora database to span multiple AWS Regions to enable fast local reads and improve disaster recovery posture. It uses storage-based replication to replicate a database across multiple Regions with typical latency of less than one second. In the rare event of a Regional degradation or outage, a database in a secondary Region can be promoted to full read and write capabilities in less than 1 minute.

Fault-tolerant and self-healing storage

Aurora makes your data durable across 3 AZs, and its storage is fault-tolerant transparently handling the loss of up to two copies of data without affecting database write availability and up to three copies without affecting read availability. Aurora storage is also self-healing, and data blocks and disks are continuously scanned for errors and replaced automatically.

Automatic, continuous, incremental backups and point-in-time restore

Aurora backups are automatic, incremental, and continuous and have no impact on database performance. The backup capability of Aurora enables point-in-time recovery for your instance, and you can restore your database to any second during your retention period, up to the last 5 minutes. Your automatic backup retention period can be configured up to 35 days. Automated backups are stored in Amazon S3, which is designed for 99.999999999% durability.

Database snapshots

You can create user-initiated backups of your Aurora instance at any time, and snapshots are stored in Amazon S3 and retained until you explicitly delete them. Aurora uses automated incremental snapshots to reduce the time and storage required. You can create a new instance from a DB snapshot whenever you desire.

Scalability

Topics

Serverless

Aurora serverless is an on-demand, auto-scaling configuration that automatically starts up, shuts down, and vertically scales capacity up or down based on your application's needs, so you never have to manage database capacity manually. You specify a capacity range in Aurora Capacity Units (ACUs), and Aurora scales within that range. When your database is idle, Aurora will automatically scale down to zero, so you only pay when your database is in use. It's especially well-suited for agentic AI applications, which typically have bursts of activity, long idle windows, and unpredictable patterns. You can also use Aurora serverless alongside with provisioned instances in your existing or new database clusters for a mixed-configuration approach.

Automated horizontal scaling

Aurora PostgreSQL Limitless Database horizontally scales beyond the write throughput and storage limits of the largest single instance while maintaining transactional consistency. Your database automatically scales based on the workload, and you only pay for what you use. In only a few steps in the RDS Management Console or AWS CLI, you can create a new Limitless Database cluster. Additional information is available in Aurora PostgreSQL Limitless Database requirements and considerations documentation.

Push-button compute scaling

Aurora lets you scale provisioned instances using the RDS APIs or the RDS Management Console, and compute scaling operations typically complete in a few minutes.

Storage auto scaling

Aurora automatically scales I/O to match the needs of your most demanding applications and increases the size of your database volume as your storage needs grow. Your database volume expands in increments of 10 GB up to a maximum of 256 TiB so you never need to provision excess storage in advance. When using Aurora I/O-Optimized configuration, you can save up to 40% when I/O spend exceeds 25% of your Aurora database spend. Additional information is available in Aurora storage and reliability documentation.

Low-latency read replicas

You can create up to 15 read replicas to increase read throughput without impacting performance on the primary instance. Aurora read replicas share the same underlying storage as the primary instance, lowering costs and avoiding the need to perform writes at the replica nodes. This frees up more processing power to serve read requests and reduces the replica lag time—often down to single-digit milliseconds.

Aurora provides a reader endpoint for automatic connection routing and load balancing across read replicas and supports auto scaling to add and remove replicas based on your performance metrics settings. The application can connect without having to keep track of replicas as they are added and removed. Aurora also supports cross-Region read replicas, which provide fast local reads to your users, and each Region can have an additional 15 Aurora read replicas to further scale local reads.

Custom database endpoints

Custom endpoints allow you to distribute and load balance workloads across different sets of database instances. For example, you can provision a set of Aurora read replicas to use an instance type with higher memory capacity in order to run an analytics workload while keeping other instances isolated for transactional traffic.

Security

Topics

- Network isolation
- Resource-level permissions
- Encryption

- Advanced auditing
- Threat detection

Network isolation

Aurora runs in Amazon VPC, isolating your database in your own virtual network. You can connect to your on-premises IT infrastructure using industry-standard encrypted IPsec VPNs and configure firewall settings to control network access to your DB instances.

Resource-level permissions

Aurora integrates with AWS IAM to give you fine-grained access control over the actions that your IAM users and groups can take on specific Aurora resources (for example, DB instances, DB snapshots, DB parameter groups, DB event subscriptions, DB option groups). You can tag your Aurora resources and control actions taken on groups of resources that have the same tag and tag value. Additional information is available in IAM database authentication documentation.

Encryption

Aurora encrypts your data at rest using keys you create and control through AWS KMS. When encryption is enabled, data stored at rest in the underlying storage is encrypted, as are the automated backups, snapshots, and replicas in the same cluster. Aurora uses SSL (AES-256) to secure data in transit.

Advanced auditing

Aurora logs database events with minimal impact on database performance. You can analyze logs for database management, security, governance, regulatory compliance, and other purposes. You can also monitor activity by sending audit logs to Amazon CloudWatch.

Threat detection

Amazon GuardDuty offers threat detection for Aurora to help you identify potential threats to data stored in Aurora databases. GuardDuty RDS Protection profiles and monitors login activity to existing and new databases in your account and uses tailored ML models to accurately detect suspicious logins to Aurora databases. If a potential threat is detected, GuardDuty generates a security finding that includes database details and rich contextual information on suspicious

activity. Aurora integration with GuardDuty provides direct access to database event logs without requiring you to modify your databases and has no impact on database performance.

Easy to manage

Topics

Quick create and connect

You can create and connect to an Aurora PostgreSQL database in seconds with two clicks. Pre-configured with an Aurora serverless cluster, you can get started without worrying about database configuration. You also get a new internet gateway that fully supports the Postgres wire protocol, enabling secure connections from your favorite tools and IDEs outside of AWS infrastructure — no VPN or AWS Direct Connect required.

Monitoring and metrics

Aurora provides multiple options for monitoring and optimizing database performance. CloudWatch metrics for Aurora tracks key operational metrics such as compute, memory, storage, query throughput, cache hit ratio, and active connections. You can set CloudWatch alarms for specific metrics over a specified time period, and perform actions based on customizable thresholds.

CloudWatch Database Insights consolidates logs and metrics from your applications, your databases, and the operating systems on which they run into a unified view in the console. Using its pre-built dashboards, recommended alarms, and automated telemetry collection, you can

monitor the health of your database fleets and use a guided troubleshooting experience to drill down to individual instances for root-cause analysis. Application developers can correlate application performance with database performance by drilling down from the context of their application performance view in CloudWatch Application Signals to the specific dependent database in CloudWatch Database Insights. CloudWatch Database Insights inherits all the features of RDS Performance Insights along with additional features such as fleet-level monitoring, integration with application performance monitoring and correlation of database metrics with logs and events.

Enhanced Monitoring provides metrics in real time from the operating system instance running your database. You can view all the system metrics and process information for your Aurora database instances on the console. Aurora delivers the metrics from Enhanced Monitoring to your CloudWatch Logs account. You can create metrics filters in CloudWatch from CloudWatch Logs and display the graphs on the CloudWatch dashboard.

Additionally, DevOps Guru for RDS, powered by machine learning, automatically identifies, diagnoses, and provides recommendations for database issues, including resource overuse and SQL query problems, all without requiring ML or deep database expertise. You can simply enable DevOps Guru for RDS in the RDS Console Management, Performance Insights, or DevOps Guru Console for comprehensive database monitoring.

Amazon RDS blue/green deployments

RDS blue/green deployments let you make safer, simpler, and faster database updates with zero data loss. In a few steps, blue/green deployments create a staging environment that mirrors production and keeps the two environments in sync using logical replication. You can make changes – such as major/minor version upgrades, schema modifications, and parameter changes — without impacting your production workload. When promoting your staging environment, blue/green deployments blocks write to both the blue and green environments until switchover is complete. Blue/green deployments use built-in switchover guardrails that time out promotion if it exceeds your maximum tolerable downtime, detects replication errors, checks instance health, and more.

Automatic software patching

Aurora keeps your database up to date with the latest patches automatically. You control if and when your instance is patched through DB Engine Version Management. Aurora uses zero-downtime patching when possible: if a suitable time window appears, the instance is updated in

place, application sessions are preserved and the database engine restarts while the patch is in progress, leading to only a transient (approximately five seconds) drop in throughput.

Upgrade rollout policies

AWS Organizations upgrade rollout policies let you centrally manage and stagger automatic minor version upgrades across multiple AWS accounts. You can upgrade your Aurora databases in a specified order (e.g., development before production), define upgrade sequences using account-level policies or resource tags, provide time between upgrade phases to validate changes, and monitor upgrade health notifications for each phase. To use upgrade rollout policies your AWS account must be part of an organization in AWS Organizations with upgrade rollout policy enabled. Additional information is available in managing organization policies documentation.

Event notifications

Aurora notifies you by email or SMS of important database events such as automated failovers. You can subscribe to over 40 different DB events using the RDS Management Console, RDS API, or AWS CLI.

Database cloning

Aurora supports quick, efficient cloning operations, where entire multi-terabyte database clusters can be cloned in minutes. Cloning is useful for a number of purposes including application development, testing, database updates, and running analytical queries. Immediate availability of data can significantly accelerate your software development and upgrade projects and make analytics more accurate. You can clone an Aurora database in only a few steps, and you don't incur any storage charges, except if you use additional space to store data changes.

Stop and start

You save costs by stopping your Aurora database when it is not in use, e.g., during development and test cycles. Stopping your database does not delete your data, and you can restart in a few steps. Additional information is available in the start/stop documentation.

Cost-effective

Topics

- Pay only for what you use

- AWS Free Tier
- Price predictability
- Optimize I/O costs
- Optimized reads

Pay only for what you use

Aurora requires no upfront commitment. You pay an hourly charge for each instance that you launch, and when you're finished with an Aurora DB instance, you can delete it. You do not need to overprovision storage as a safety margin, and you only pay for the storage you actually consume. Additional information is available on the Aurora pricing page.

AWS Free Tier

New AWS customers can get started with Aurora PostgreSQL serverless at no cost through the AWS Free Tier. You receive $100 in credits at sign-up with the opportunity to earn an additional $100 for a total of $200 to use across eligible AWS services, including Aurora, for up to 12 months.

The Free plan gives you access to Aurora PostgreSQL serverless instances with up to 4 ACUs and 1 GiB of storage per cluster. You can upgrade to the Paid plan at any time to scale up to 256 ACUs and 256 TiB of storage. You can also access Aurora PostgreSQL serverless directly through the Vercel Marketplace using only an email address with the same $100 in free credits applied automatically.

Price predictability

Aurora offers the flexibility to optimize your database spend by choosing between two configuration options – Aurora I/O-Optimized and Aurora Standard – based on your price-performance and price-predictability needs, regardless of the I/O consumption of your application. Neither option requires upfront I/O or storage provisioning and both can scale I/O to support your most demanding applications. You can switch between configurations using the RDS Management Console, AWS CLI, or AWS SDK.

Optimize I/O costs

Aurora was designed to eliminate unnecessary I/O operations to reduce costs and ensure resources are available for serving read/write traffic. Every database page read operation counts as one I/

O (8 KB in Aurora PostgreSQL and 16 KB in Aurora MySQL). Write I/O operations are counted in 4 KB units and are only consumed when pushing transaction log records to the storage layer for the purpose of making writes durable. For example, a transaction log record that is 1,024 bytes counts as one I/O operation. However, concurrent write operations whose transaction log is less than 4 KB can be batched together by the Aurora database engine to optimize I/O consumption. Unlike traditional database engines, Aurora never pushes modified database pages to the storage layer, resulting in further I/O consumption savings.

You can monitor how many I/O operations your Aurora instance is using "Billed read operations" and "Billed write operations" metrics in the monitoring section of the RDS Management Console.

You are charged for read and write I/O operations when you configure your database clusters to the Aurora Standard configuration. You are not charged for read and write I/O operations when you configure your database clusters to Aurora I/O-Optimized. Additional information on the pricing of I/O operations is available on Aurora pricing page.

Optimized reads

With optimized reads for Aurora PostgreSQL, you have more flexibility to grow your datasets without the need to frequently upsize their database instances to obtain larger memory capacity. Optimized reads deliver up to 8x improved query latency and up to 30% cost savings for latency-sensitive applications with large working sets, using performance enhancements such as tiered caching and temporary objects.

Tiered caching delivers up to 8x improved query latency and up to 30% cost savings for read-heavy, I/O-intensive applications such as operational dashboards, anomaly detection, and vector-based similarity searches. These benefits are realized as caching data is automatically evicted from the in-memory database buffer cache onto local storage to speed up subsequent access of that data. In addition, temporary objects achieve faster query processing by placing temporary tables generated by Aurora PostgreSQL on local storage, improving the performance of queries involving sorts, hash aggregations, high-load joins, and other data-intensive operations.

Agentic AI

Topics

- Agent memory
- Vector database

- Machine learning

Agent memory

AI agents are stateless without memory. Aurora provides long-term memory for your AI agents, giving them the ability to remember past interactions and enable more intelligent, context aware, and personalized conversations.

Vector database

With Aurora PostgreSQL, you can store, search, index, and query vector embeddings alongside your transactional data – and vector search scales to hundreds of billions of vectors. You can also use Aurora PostgreSQL as your vector database in Amazon Bedrock Knowledge Bases. With one click, you can configure Aurora as a Knowledge Base for Bedrock and connect your organization's private data sources from Aurora to LLMs available in Bedrock to enable automated Retrieval-Augmented Generation (RAG) workflows. This makes your LLMs more knowledgeable about your specific domain and organization. Additional information is available in Aurora PostgreSQL as a Knowledge Base for Amazon Bedrock in one click documentation.

Machine learning

Aurora machine learning (Aurora ML) simplifies adding generative AI model predictions to your Aurora database. Aurora ML exposes ML models as SQL functions, allowing you to use standard SQL to call ML models, pass data to them, and return predictions, text summaries, or sentiment as query results. With Aurora ML, you can make the process of adding new embeddings to your Aurora PostgreSQL database with the pgvector extension real-time via periodic calls to a SageMaker or Amazon Bedrock model, which returns the latest, up-to-date embeddings.

Developer productivity

Topics

- MCP servers
- Agentic tools
- Agent skills
- Zero-ETL integration with Amazon Redshift
- Zero-ETL integration with Amazon SageMaker

- RDS Proxy
- Data API

MCP servers

Aurora MCP server provides the flexibility and ease to build and deploy from the tools and agents of your choice. A critical component in agentic AI architectures, Aurora MCP server provide the standardized interface needed for agents to interact with developer tools and your Aurora databases. Aurora MCP server is available on Github for PostgreSQL and MySQL.

Agentic tools

Aurora integrates directly with the developer tools you already use, saving you hours or weeks of time to build, test, and deploy. AI-enabled IDEs (e.g., Kiro) and agentic development platforms (e.g., Vercel) help you build in your preferred environment and significantly shorten the path from idea to working application. Aurora is integrated with agentic frameworks (e.g., Amazon Bedrock AgentCore, Letta, LangGraph) to persist agent memories using a fully managed database.

Agent skills

Kiro power for Aurora PostgreSQL provides specialized skills to AI agents that contain specific Aurora knowledge, MCP tools, and best practices to instantly understand how to work with Aurora PostgreSQL. With one-click installation, you can start building database-backed apps in Kiro using natural language – no Aurora expertise required.

Zero-ETL integration with Amazon Redshift

Aurora zero-ETL integration with Amazon Redshift enables near real-time analytics and ML using Amazon Redshift on petabytes of transactional data from Aurora by removing the need for you to build and maintain complex data pipelines that perform extract, transform, and load (ETL) operations. Transactional data is automatically and continuously replicated within seconds of being written in Aurora and is seamlessly made available in Amazon Redshift.

Once data is available in Amazon Redshift, you can start analyzing it immediately and apply advanced features like data sharing, materialized views, and Amazon Redshift ML to get holistic and predictive insights. You can consolidate multiple tables from various Aurora database clusters and replicate your data into one Amazon Redshift data warehouse to run unified analytics across multiple applications and data sources. When using both Aurora serverless and Amazon Redshift

Serverless, you can generate near real-time analytics on transactional data without having to manage any infrastructure for data pipelines. Additional information is available in Aurora zero-ETL integrations with Amazon Redshift documentation.

Zero-ETL integration with Amazon SageMaker

Aurora zero-ETL integration with Amazon SageMaker enables near real-time access of your data in the lakehouse architecture of SageMaker to run a broad range of analytics. With zero-ETL integration, data from Aurora is automatically extracted and loaded into the lakehouse in SageMaker enabling you to derive near real-time insights from your operational data. The data synced into the lakehouse is compatible with Apache Iceberg open standards, enabling you to use your preferred analytics tools and query engines such as SQL, Apache Spark, BI, and AI/ML tools. Additional information is available in Aurora zero-ETL integration with Amazon SageMaker documentation.

RDS Proxy

Aurora works in conjunction with RDS Proxy, a fully managed, highly available database proxy that makes applications more scalable, more resilient to database failures, and more secure. RDS Proxy allows applications to pool and share connections established with the database, improving database efficiency and application scalability. It reduces failover times by automatically connecting to a new database instance while preserving application connections. It enhances security through integrations with AWS IAM and AWS Secrets Manager.

Data API

Data API is an easy-to-use, secure HTTPS API for executing SQL queries against Aurora databases. It eliminates the need for database drivers, client-side connection pools, and VPC networking configuration to securely connect to an Aurora database, which makes accessing Aurora as simple as making an API call. Data API also improves application scalability by automatically pooling and sharing database connections and is integrated with AWS IAM and AWS Secrets Manager. You can call Data API via applications built with an AWS SDK or through AWS AppSync GraphQL APIs.

Migration support

Topics

- MySQL migration

MySQL migration

Standard MySQL import and export tools (mysqldump, mysqlimport) work with Aurora. You can also create a new Aurora database from an RDS for MySQL DB snapshot, and migration operations based on DB snapshots typically complete in under an hour. Alternatively, AWS Database Migration Service (AWS DMS) offers built-in native tooling for seamless migration with no replication instances to provision or scale. You can initiate a database migration with a few simple clicks and only pay on an hourly basis for the time used. Finally, you can also set up binlog-based replication between an Aurora MySQL database and an external MySQL database running inside or outside of AWS.

PostgreSQL migration

Standard PostgreSQL import and export tools (pg_dump, pg_restore) work with Aurora. Aurora also supports snapshot import from RDS for PostgreSQL, and replication with AWS DMS.

Commercial database migrations

AWS DMS accelerates migrations from commercial database to Aurora with DMS Schema Conversion, which automatically assesses and converts schemas and source objects to be compatible with the target Aurora cluster. DMS Serverless automates provisioning, monitoring, and scaling of migration resources.

Amazon Aurora FAQ

What is Amazon Aurora?

Amazon Aurora is a fully managed relational database service designed for high performance and availability at global scale for PostgreSQL, MySQL, and DSQL. Aurora PostgreSQL and Aurora MySQL offer full open source compatibility, so you can use your existing code, applications, drivers, and tools with little or no modification.

Aurora's storage system is distributed, fault-tolerant, and self-healing — it automatically scales up to 256 TiB per database instance and replicates your data across three Availability Zones (AZs), you only pay for one copy. With Aurora PostgreSQL and MySQL, you get up to 15 low-latency read replicas, point-in-time recovery, and continuous backup to Amazon S3. Aurora automates administration tasks like hardware provisioning, patching, and backups, delivering commercial-grade reliability at the cost effectiveness of open source databases. With Aurora DSQL, its serverless distributed architecture means zero infrastructure management and virtually unlimited scalability.

Is Amazon Aurora MySQL compatible?

Yes. Amazon Aurora is drop-in compatible with MySQL and adds support for new releases regularly. You can migrate MySQL databases to and from Aurora using standard tools like mysqldump and mysqlimport, or use Amazon RDS DB Snapshot migration. Your existing MySQL code, applications, drivers, and tools work with Aurora with little or no change. See the Aurora MySQL compatibility documentation for supported versions.

Is Amazon Aurora PostgreSQL compatible?

Yes. Amazon Aurora is drop-in compatible with PostgreSQL and adds support for new releases regularly. You can migrate PostgreSQL databases to and from Aurora using pg_dump and pg_restore, or use RDS DB Snapshot migration. For SQL Server migrations, Babelfish for Aurora PostgreSQL lets your applications work without code changes. See the Aurora PostgreSQL compatibility documentation for supported versions.

Does Amazon Aurora support PostgreSQL extensions?

Yes, Amazon Aurora PostgreSQL supports all PostgreSQL extensions available with Aurora. You can enable extensions using the standard CREATE EXTENSION command, just as you would with any PostgreSQL database. If you need help with a specific extension, contact AWS Support. Customers with an active AWS Premium Support plan can reach out directly for Aurora-specific troubleshooting and guidance.

How do I get started with Amazon Aurora?

Getting started with Amazon Aurora is straightforward and takes just a few minutes — or even seconds with the latest deployment options. You can create and connect to an Aurora PostgreSQL database in seconds using express configuration in the Amazon RDS console, pre-configured with an Aurora serverless cluster. Once created, you have access to Aurora features that can be modified at any time. You also get a new internet gateway supporting the Postgres wire protocol for secure connections from your favorite tools and IDEs outside AWS — no VPN required. New AWS customers can also get started with Aurora PostgreSQL serverless through the Vercel Marketplace using only an email address.

To customize configurations, sign in to the Amazon RDS console and navigate to Amazon RDS under the Database category. Select Amazon Aurora as your engine and choose either the MySQL-compatible or PostgreSQL-compatible edition for full configuration. The console guides you through instance size, storage, networking, and security settings.

For step-by-step instructions, tutorials, and best practices, visit the Getting Started with Amazon Aurora page, which provides quickstart guides, sample code, and migration tools. If you're new to Aurora, consider starting with a development environment to familiarize yourself with Aurora's features before deploying production workloads.

In which AWS Regions is Amazon Aurora available?

Aurora is available across AWS Regions worldwide. For a complete list, see Supported features in Amazon Aurora by AWS Region and Aurora DB engine.

Does Amazon Aurora require special drivers?

No. Aurora works with standard MySQL and PostgreSQL database drivers. You can use the same drivers and connection libraries you already use with MySQL or PostgreSQL databases, with no modifications required.

Performance

Topics

How does Amazon Aurora improve MySQL performance?

Amazon Aurora delivers up to 6x the throughput of stock MySQL by tightly integrating the database engine with an SSD-based virtualized storage layer purpose-built for database workloads. This reduces writes to the storage system, minimizes lock contention, and eliminates delays created by database process threads.

How does Amazon Aurora improve PostgreSQL performance?

Amazon Aurora delivers up to 6x the throughput of stock PostgreSQL by tightly integrating the database engine with an SSD-based virtualized storage layer purpose-built for database workloads. Aurora decouples compute from storage, reducing writes to the storage system, minimizing lock contention, and eliminating delays created by database process threads.

How can I maximize throughput on Amazon Aurora MySQL?

Aurora is fully compatible with MySQL, so existing applications and tools run without modification. The area where Aurora excels beyond stock MySQL is highly concurrent workloads. To maximize throughput, design your applications to drive a large number of concurrent queries and

transactions — Aurora's storage architecture is optimized to handle high parallelism with minimal lock contention.

How can I maximize throughput on Amazon Aurora PostgreSQL?

Aurora is fully compatible with PostgreSQL, so existing applications and tools run without modification. Aurora delivers the highest performance gains over stock PostgreSQL under high concurrency. To maximize throughput, build your applications to drive a large number of concurrent queries and transactions. Aurora's decoupled compute and storage architecture reduces write amplification and I/O bottlenecks, enabling consistently higher throughput as connections scale.

What is Amazon Aurora Parallel Query?

Parallel Query for Aurora MySQL pushes down and distributes the computational load of a single query across thousands of CPUs in Aurora's storage layer, speeding up analytical queries by up to two orders of magnitude. It can push down processing of more than 200 SQL functions, equijoins, and projections — no changes to query syntax are required, as the query optimizer automatically decides whether to use Parallel Query.

Parallel Query is ideal for operational analytics — when you need fast analytical queries on fresh data in large tables. It is not a data warehouse replacement; for exabyte-scale analytics, consider Amazon Redshift. Parallel Query is compatible with Aurora serverless and Backtrack, runs on R* instance family types, and is included at no additional charge beyond standard instance, I/O, and storage pricing.

What is optimized reads for Aurora PostgreSQL?

Optimized reads for Aurora PostgreSQL delivers up to 8x improved query latency and up to 30% cost savings compared to instances without it. It is ideal for applications with large datasets that exceed the memory capacity of a database instance. Optimized Reads is available on Intel-based R6id and Graviton-based R6gd and R8gd instances (not available on Aurora serverless).

Optimized reads uses local NVMe-based SSD storage and includes two key features. Tiered caching automatically caches data evicted from the in-memory buffer cache onto local storage, speeding up subsequent accesses (available with Aurora I/O-Optimized). Temporary objects places temporary tables on local storage, improving performance of queries involving sorts, hash aggregations, and high-load joins. Approximately 90% of local storage is available for these

features. If local storage fails, Aurora automatically performs a host replacement and triggers an in-region failover.

Can I use optimized reads for Aurora PostgreSQL with Aurora Standard and Aurora I/O-Optimized configurations?

Yes, Amazon Aurora optimized reads is available with both configurations. On both configurations, optimized reads-enabled instances automatically map temporary tables to the NVMe-based local storage to improve the performance of analytical queries and index rebuilds.

For I/O intensive workloads which are read heavy, optimized reads-enabled instances on Aurora PostgreSQL configured to use Aurora I/O-Optimized automatically cache data evicted from memory on NVMe-based local storage to deliver up to 8x improved query latency and up to 30% cost savings compared to instances without it, for applications with large datasets that exceed the memory capacity of a database instance.

How do I get started with optimized reads for Aurora PostgreSQL?

Customers can get started with Amazon Aurora Optimized Reads through the AWS Management Console, CLI, and SDK. Optimized reads is available on all R6id and R6gd instances by default. To use this capability, customers can simply modify their existing Aurora database clusters to include R6id and R6gd instances or create new database clusters using these instances. See the Amazon Aurora optimized reads documentation to get started.

Availability and Durability

Topics

- Do I need to update my application code if my primary Region becomes unavailable?
- Can I set up cross-Region replicas?
- Can I add replicas to a cross-Region Aurora cluster?
- Can I promote a cross-Region replica to primary?
- What is the replication lag for replicas in Aurora?

How does Amazon Aurora store data for high availability?

Amazon Aurora automatically divides your database volume into 10 GiB segments spread across many disks. Each segment is replicated six ways across three Availability Zones (AZs). Aurora is designed to transparently handle:

- Loss of up to two copies of data without affecting database write availability
- Loss of up to three copies without affecting read availability

Aurora storage is also self-healing — data blocks and disks are continuously scanned for errors and repaired automatically.

How does Amazon Aurora recover after a database crash?

Unlike other databases, Aurora does not need to replay the redo log from the last checkpoint after a crash. This reduces database restart times to less than 60 seconds in most cases. Aurora also moves the buffer cache out of the database process and makes it available immediately at restart, preventing brownouts while the cache repopulates.

Are automated backups enabled on Amazon Aurora?

Yes, automated continuous backups are always enabled on Amazon Aurora DB Instances with no database performance impact. Aurora automatically makes your data durable across three AZs and will attempt to recover your database in a healthy AZ with no data loss. You can also perform point-in-time restore operations (up to five minutes in the past) or restore from DB snapshots. Can I take manual snapshotsof my Aurora database?

Yes, and there is no performance impact when taking snapshots. Note that restoring data from snapshots requires the creation of a new DB Instance.

If my database fails, what is my recovery path?

Amazon Aurora automatically makes your data durable across three AZs in a Region and will automatically attempt to recover your database in a healthy AZ with no data loss. In the unlikely event your data is unavailable within Amazon Aurora storage, you can restore from a DB snapshot or perform a point-in-time restore operation to a new instance. Note that the latest restorable time for a point-in-time restore operation can be up to five minutes in the past.

What happens to my data if I delete my Aurora DB instance?

You can choose to create a final DB snapshot when deleting your DB Instance. You can use this snapshot to restore the deleted DB Instance at a later date. Amazon Aurora retains this final user-created DB Snapshot along with all other manually created DB Snapshots after the DB Instance is deleted. Automated backups created for point-in-time restore are not kept after deletion.

Can I share my snapshots with another AWS account?

Yes. You can share a snapshot with up to 20 AWS account IDs or make them public. Shared snapshots are accessible only within the same AWS Region. Encrypted Aurora snapshots can also be shared.

You can use this feature to share data between your various environments (production, dev/test, staging, etc.) that have different AWS accounts, as well as keep backups of all your data secure in a separate account in case your main AWS account is ever compromised.

Can I automatically share snapshots?

No. Automatic sharing of DB snapshots is not supported. To share a snapshot, you must manually create a copy of the snapshot and then share the copy.

Will I be billed for shared snapshots?

There is no charge for sharing snapshots between accounts. However, you may be charged for the snapshots themselves and any databases restored from them. Learn more about Aurora pricing.

How does failover work in Amazon Aurora?

Failover is handled automatically. The process depends on your configuration.

With replicas, Aurora flips the canonical name record (CNAME) to point at the healthy replica, which is promoted to new primary. Failover typically completes within 30 seconds. For faster failovers, use Amazon RDS Proxy which automatically connect to the failover instance while preserving application connections – reducing failover times by up to 66%.

Aurora serverless works like provisioned for failover and high availability.

For single instance (no replicas, Aurora attempts to create a new DB Instance in the same AZ as the original instance on a best-effort basis. For example, it may not succeed if there is an issue that is broadly affecting the AZ.

Your application should retry database connections in the event of connection loss. Disaster recovery across Regions is a manual process, where you promote a secondary Region to take read/write workloads.

You can assign priority tiers to replicas to control which one gets promoted first. To increase availability, create 1–15 replicas across any of three AZs.

If I have a primary database and a replica actively taking read traffic in Aurora and a failover occurs, what happens?

Amazon Aurora will automatically detect a problem with your primary instance and trigger a failover. If you are using the Cluster Endpoint, your replica will be promoted to the primary. Read traffic that your replicas were serving will be briefly interrupted and directed to the newly promoted Aurora Replica until the old primary node is recovered as a replica.

Can I control which replica gets promoted during failover?

Yes. You can assign a promotion priority tier to each instance on your cluster. When the primary instance fails, Amazon RDS promotes the replica with the highest priority. If two or more replicas share the same priority, Amazon RDS promotes the largest one. You can modify priority tiers at any time without triggering a failover. For more information, read the Amazon Aurora User Guide.

Can I prevent a specific replica from being promoted?

You can assign lower priority tiers to replicas you don't want promoted to primary. However, if the higher priority replicas on the cluster are unhealthy or unavailable, Amazon RDS will promote the lower priority replica.

How can I increase database availability?

You can add replicas. Any replica can be promoted to primary without any data loss, enhancing fault tolerance. Simply create 1 to 15 replicas in any of three AZs, and Amazon Aurora will automatically include them in failover primary selection. You can use Amazon Aurora Global

Database if you want your database to span multiple AWS Regions for disaster recovery from region-wide outages.

What is Amazon Aurora Global Database?

Amazon Aurora Global Database allows a single Aurora database to span multiple AWS Regions. It replicates your data with no impact on database performance, enables fast local reads with typical latency of under one second, and provides disaster recovery from region-wide outages.

Key capabilities:

- Promote a secondary region to full read/write in under one minute
- Create up to 10 secondary Regions
- Supports mixed provisioned/serverless instance types between regions
- Available for both Aurora MySQL and Aurora PostgreSQL

How do I create an Amazon Aurora Global Database?

To get started, you can use the Amazon RDS console, AWS SDK, or AWS CLI to create an Aurora Global Database. You can use a mixed configuration of provisioned or serverless instance class types between your primary and secondary Regions. You can also configure your primary Region as the Aurora I/O-Optimized cluster configuration and your secondary Regions as Aurora Standard or the reverse. To learn more, visit Creating an Amazon Aurora Global Database.

Can I use database activity streams with Aurora Global Database?

Yes. If your goal is to analyze database activity, consider using Aurora advanced auditing, general logs, and slow query logs instead, to avoid impacting the performance of your database.

Do I need to update my application code if my primary Region becomes unavailable?

No. If your primary Region becomes unavailable, you can use the managed cross-Region Aurora Global Database failover operation to promote a secondary Region. You can also use the Aurora Global Database writer endpoint to avoid the need to make application code changes. To learn more, visit Connecting to an Amazon Aurora Global Database.

What types of replicas does Aurora support?

Aurora supports multiple replication options:

For cross-Region replication, use Aurora Global Database for physical replication with sub-second latency, or native logical replication (binlog for MySQL, replication slots for PostgreSQL).

Can I set up cross-Region replicas?

Yes, you can set up cross-Region replicas using either physical or logical replication. Physical replication, called Amazon Aurora Global Database, uses dedicated infrastructure and can replicate up to five secondary Regions with typical latency of under a second. It is available for both Aurora MySQL and Aurora PostgreSQL. For low-latency global reads and disaster recovery, we recommend using Amazon Aurora Global Database.

Can I add replicas to a cross-Region Aurora cluster?

Yes, you can add up to 15 replicas on each cross-Region Aurora cluster. They share the same underlying storage as the cross-Region replica and typically lag behind the primary by tens of milliseconds.

Can I promote a cross-Region replica to primary?

Yes. For logical (binlog) replication, the promotion process typically takes a few minutes depending on your workload. With Amazon Aurora Global Database, you can promote a secondary Region to take full read/write workloads in under a minute.

What is the replication lag for replicas in Aurora?

Since replicas share the same data volume as the primary instance in the same AWS Region, there is virtually no replication lag. We typically observe lag times in the tens of milliseconds. For cross-Region replication, binlog-based logical replication lag can grow indefinitely based on change/apply rate and network delays, but under typical conditions, under a minute of replication lag is common. Cross-Region replicas using Aurora Global Database's physical replication will have a typical lag of under a second.

Can I set up binlog replication with an external MySQL database?

Yes, you can set up binlog replication between an Aurora MySQL instance and an external MySQL database. The other database can run on Amazon RDS, as a self-managed database on AWS, or completely outside of AWS. If you're running Aurora MySQL 5.7, consider setting up GTID-based binlog replication for complete consistency.

Scalability

How does Amazon Aurora scale?

Topics

Aurora storage starts at a minimum of 10 GiB and automatically grows up to 256 TiB in 10 GiB increments with no impact to database performance. There is no need to provision storage in advance. For workloads that need to scale beyond 256 TiB, Aurora PostgreSQL Limitless Database provides automated horizontal scaling.

What are the ways to scale compute resources in Amazon Aurora?

There are three ways to scale compute resources:

- Aurora serverless: An on-demand, autoscaling configuration that adjusts database compute based on application demand. You specify a capacity range and Aurora scales automatically. Learn more in the Aurora serverless user guide.

- Aurora PostgreSQL Limitless Database: Automatically scales compute horizontally to support high-scale applications that need more write throughput or storage than a single instance provides. Learn more in the Limitless Database User Guide.
- Manual scaling: Select the desired DB instance type in the Amazon RDS console. Changes apply during your maintenance window or immediately with the Apply Immediately flag.

What is Amazon Aurora serverless?

Aurora serverless is an on-demand, autoscaling configuration for Amazon Aurora. It automatically adjusts database compute capacity based on your application's needs — you specify a desired capacity range in Aurora Capacity Units (ACUs) and Aurora scales within that range in fine-grained increments. It's especially well-suited for agentic AI applications, which typically have bursts of activity, long idle windows, and unpredictable patterns. You pay per-second for ACU usage.

Aurora serverless scales down to zero ACUs when your database has no active connections, automatically pausing the instance to eliminate compute costs during idle periods. When a new connection arrives, the instance automatically resumes — typically faster than starting a stopped cluster. This automatic pause and resume capability is available for both Aurora PostgreSQL and Aurora MySQL, and is especially useful for development, test, and SaaS multi-tenant environments where databases may have extended periods of inactivity.

Aurora serverless supports all features of provisioned Aurora, including read replica, Multi-AZ configuration, Aurora Global Database, RDS Proxy, and CloudWatch Database Insights. If capacity cannot scale fast enough for a sudden workload change, you can set it explicitly via the console, CLI, or API.

How do I connect to Aurora serverless?

You access Aurora serverless from within a client application running in the same VPC. You can't give a public IP address to Aurora serverless.

Can I use Aurora serverless with my existing Aurora cluster?

Yes, you can add Aurora serverless instances to your existing Aurora DB cluster , creating a mixed-configuration cluster with any combination of provisioned and serverless instances. To test it, add a serverless reader, validate it with read-only workloads, then initiate a failover to use it for both reads and writes with minimal downtime. You can also restore snapshots between provisioned and serverless clusters in either direction.

How is Aurora serverless priced?

In Aurora serverless, database capacity is measured in Aurora Capacity Units (ACUs). You pay a flat rate per second of ACU usage. Compute costs depend on the database cluster configuration you choose: Aurora Standard or Aurora I/O-Optimized. Visit the Aurora pricing page for information about pricing and Regional availability.

What is Aurora PostgreSQL Limitless Database?

Aurora PostgreSQL Limitless Database provides automated horizontal scaling while maintaining the simplicity of operating inside a single database. It splits data across database instances using customer-specified shard keys and automatically scales compute and storage as your workload grows. Your application connects using the standard cluster endpoint — no special drivers required.

Limitless Database supports three table types: sharded tables distributed across shards for scaling your largest tables, reference tables copied in full on every shard for faster joins, and standard tables placed on a single shard. You may need to adjust your schema to add shard keys to tables and queries. For high availability, set compute redundancy greater than zero to achieve 99.99% availability with automatic failover. Available with Aurora I/O-Optimized (see documentation for supported versions and pricing).

When should I use Aurora PostgreSQL Limitless Database?

You should use Limitless Database for applications that need to scale horizontally and require more write throughput or data storage capacity than a single Aurora database instance supports. For example, an accounting application can be horizontally partitioned by user since each user's accounting data is independent from the others.

How does Limitless Database differ from Aurora serverless and replicas?

Aurora offers three scaling mechanisms, each designed for different needs:

How does Aurora PostgreSQL Limitless Database work?

Aurora PostgreSQL Limitless Database is a distributed, horizontally scalable PostgreSQL database that automatically splits data across multiple serverless compute instances based on customer-defined shard key.

Architecture:

Shard key — A designated table column (e.g., User-ID) that determines how data is distributed across instances.

Routers — Serverless nodes that analyze incoming queries and route them to the correct shard or orchestrate cross-shard queries.

Shards — Serverless nodes that each store a subset of data, enabling parallel processing for high write throughput.

Auto-scaling:

Aurora automatically scales up individual instances (compute and storage) first, then scales out by adding shards to handle growing workloads. Each shard key value is owned by a single serverless instance at any point in time.

Transaction semantics:

Multiple compute instances serve requests simultaneously while maintaining the same ACID transaction guarantees as single-writer Aurora PostgreSQL — no application-level transaction domain management required.

What table types does Aurora PostgreSQL Limitless Database support?

Aurora PostgreSQL Limitless Database supports three table types: sharded, reference, and standard. Sharded tables: Distributed across multiple shards based on the values of designated columns in the table, called shard keys. Use these for scaling out the largest, most I/O-intensive tables in your application. Reference tables: Fully replicated to every shard.. Ideal for infrequently modified lookup data, such as product catalogs and zip codes. Standard tables: Standard Aurora PostgreSQL tables stored together on a single shard. You can create sharded and reference tables from standard tables.

Are there any PostgreSQL compatibility considerations when using Aurora PostgreSQL Limitless Database?

To learn more about PostgreSQL compatibility considerations, visit

How do I get started and connected to Aurora PostgreSQL Limitless Database?

You can create a new Aurora PostgreSQL cluster with the supported engine version using the Amazon RDS console or Amazon APIs. Then connect to the cluster endpoint — the same way you connect to any standard Aurora PostgreSQL cluster.

To learn more, visit Aurora PostgreSQL Limitless Database User Guide.

Do I need to change my existing database schema or application to use Aurora PostgreSQL Limitless Database?

Yes, you might need to adjust your database schema to use Aurora PostgreSQL Limitless Database.

Schema changes:

- All sharded tables must include the shard key column. If a table doesn't naturally contain it, you'll need to add and backfill it.
- Column names are flexible, but the shard key column definition must match across collocated tables.
- For example, an accounting application sharded by User-ID would need to add a User-ID column to the invoice line items to collocate related items.

Query changes:

Include the shard key in application queries – which may require adjusting your queries and transactions for optimal performance.

For example, a query specifying both Invoice-ID and User-ID routes to a single shard; omitting User-ID forces execution across all shards, increasing latency.

PostgreSQL compatibility:

Some PostgreSQL features have specific requirements or limitations in Limitless Database. For more details, go to Aurora PostgreSQL Limitless Database requirements and considerations.

Does Aurora PostgreSQL Limitless Database have support for high availability?

Yes. Aurora PostgreSQL Limitless Database supports up to 99.99% availability with configurable compute redundancy. Set compute redundancy to be greater than zero to enable high availability.. Each compute instance that stores and accesses data from your Aurora PostgreSQL Limitless Database can have one or two standbys that can take over requests if the primary is unavailable. The routers will automatically redirect the traffic for minimal impact on your application.

What versions does Aurora PostgreSQL Limitless Database support?

Aurora PostgreSQL Limitless Database is available for the Aurora I/O-Optimized cluster configuration starting with PostgreSQL 16.4 compatibility. Additional information is available in the Aurora release notes.

How is Aurora PostgreSQL Limitless Database priced?

In Aurora PostgreSQL Limitless Database, database capacity is measured in ACUs. You pay a flat rate per second of ACU usage. Aurora I/O-Optimized configuration storage rates apply. For more information, visit the Aurora pricing page.

Security

Topics

Does Amazon Aurora run in Amazon Virtual Private Cloud (Amazon VPC)?

Yes, Aurora DB instances can be created in a VPC. This gives you complete control over who can access your Aurora databases, including defining virtual network topologies that closely resemble traditional on-premises networks.

For streamlined setup, you can create an Aurora serverless instance without a VPC using express configuration.

Does Amazon Aurora support dynamic data masking?

Yes. Amazon Aurora PostgreSQL supports native dynamic data masking though the pg_columnmask extension. Dynamic data masking lets you obscure sensitive column values at query time — such as personally identifiable information (PII), payment data, and health records — without altering the data stored on disk, helping you meet regulatory requirements like GDPR, HIPAA, and PCI DSS.

Does Amazon Aurora support encryption?

Yes. Amazon Aurora uses SSL (AES-256) to secure connections between the database instance and your application. You can encrypt databases at rest using keys you manage through AWS Key Management Service (AWS KMS).

When encryption is enabled, data stored at rest in the underlying storage is encrypted, along with automated backups, snapshots, and replicas in the same cluster. Encryption and decryption are handled seamlessly. For more information, see the Amazon RDS User's Guide.

Note: Encrypting an existing unencrypted Aurora instance is not currently supported. To encrypt an existing database, create a new encrypted DB instance and migrate your data into it.

How do I connect to my Aurora database securely?

Aurora databases must be accessed through the database port entered on database creation. This provides an additional layer of security for your data. Step-by-step instructions on how to connect to your Amazon Aurora database are provided in the Amazon Aurora Connectivity Guide.

Is Amazon Aurora HIPAA eligible?

Yes, both the MySQL- and PostgreSQL-compatible editions of Aurora are HIPAA eligible. You can use them to build HIPAA-compliant applications and store protected health information (PHI) under an executed Business Associate Addendum (BAA) with AWS. If you have already entered into a BAA with AWS, no further action is necessary to begin using these services in the account(s) covered by your BAA. More information about using AWS to build compliant applications is available at Healthcare Providers.

How does Amazon Aurora integrate with Amazon GuardDuty?

Aurora is integrated with Amazon GuardDuty to help identify potential threats to data stored in Aurora databases. GuardDuty RDS Protection profiles and monitors login activity using tailored ML models to detect suspicious logins. For more information, see Monitoring threats with GuardDuty RDS Protection and the GuardDuty RDS Protection User Guide.

Where can I find Aurora security updates?

You can find a current list of CVEs at Amazon Aurora Security Updates.

Billing

Topics

How much does Amazon Aurora cost?

See the Aurora pricing page for current pricing information. Aurora offers two cluster configurations — Aurora Standard and Aurora I/O-Optimized — so you can optimize costs based on your workload.

Is there a free tier for Amazon Aurora?

Yes. Amazon Aurora is available with the AWS Free Tier for Aurora PostgreSQL serverless and Aurora DSQL. New customers will receive up to $100 in credits at sign-up and can earn up to an additional $100 in credits, totaling $200, to explore eligible AWS services, including Aurora, on both Free and Paid plans. In addition, the Aurora free tier offers the following benefits:

Aurora PostgreSQL provides 4 ACUs and 1 GiB of storage per cluster on the Free plan.

Aurora DSQL provides the first 100k DPUs and 1 GiB of storage every month on the Free and Paid plans.

Check the Aurora pricing page for current Free Tier details and eligibility.

Can I save money with a Database Savings Plan?

Yes, you can purchase a Database Savings Plan for your Aurora usage and reduce costs by up to 35% when you commit to a consistent amount of usage over a 1-year term.

Is there a charge for Aurora replication?

No, Aurora replication is bundled into the price. You are charged based on the storage your database consumes at the database layer, not the storage consumed in the virtualized storage layer.

What are I/O operations in Aurora and how are they calculated?

I/O operations are performed by the Aurora database engine against its SSD-based virtualized storage layer. Every database page read operation counts as one I/O.

Key details about I/O billing:

- Read I/Os: If your query traffic can be served entirely from memory or cache, you will not be charged for data page retrieval. Pages not in memory incur read I/O charges (16 KB per page in Aurora MySQL, 8 KB in Aurora PostgreSQL).

- Write I/Os: Counted in 4 KB units. Only redo log records (Aurora MySQL) or write-ahead log records (Aurora PostgreSQL) are written — Aurora never flushes dirty data pages to storage.
- Monitoring: Check "VolumeReadIOPs" and "VolumeWriteIOPs" metrics in the Amazon RDS console.

With Aurora Standard, you pay per-request for I/O. With Aurora I/O-Optimized, read and write I/O operations are included at no additional charge.

What is the difference between Aurora Standard and Aurora I/O-Optimized?

Aurora offers two configurations to optimize your database spend: Aurora Standard and Aurora I/O-Optimized.

You can switch between configurations using the Amazon RDS console, AWS CLI, or AWS SDK. You can switch to I/O-Optimized once every 30 days, and switch back to Standard at any time. Aurora I/O-Optimized works with existing Aurora Reserved Instances.

When should I choose Aurora I/O-Optimized?

Aurora I/O-Optimized is the ideal choice when you need predictable costs for any application. It delivers improved price performance for I/O-intensive applications that require high write throughput or run analytical queries processing large amounts of data. For customers with an I/O spend exceeding 25% of their Aurora bill, you can save up to 40% on costs.

How do I switch to Aurora I/O-Optimized?

You can use the one-click experience in the Amazon RDS console to change the storage type of your existing database clusters. You can also use the AWS CLI or AWS SDK. You can switch to I/O-Optimized once every 30 days and switch back to Standard at any time.

Does Aurora I/O-Optimized work with Reserved Instances?

Yes, Aurora I/O-Optimized works with existing Aurora Reserved Instances. Aurora automatically accounts for the price difference between Aurora Standard and Aurora I/O-Optimized with Reserved Instances, providing even more savings on your I/O spend.

Are there changes to backup or snapshot pricing with I/O-Optimized?

No. There are no changes to the price of backtrack, snapshot, export, or continuous backup with Aurora I/O-Optimized.

Do cross-Region replication I/O charges still apply with I/O-Optimized?

Yes. The charges for I/O operations required to replicate data across Regions continue to apply. Aurora I/O-Optimized does not charge for read and write I/O operations, which is different from data replication.

Are there additional charges for optimized reads for Aurora PostgreSQL?

No. There are no additional charges for optimized reads beyond the price of Intel-based R6id and Graviton-based R6gd and R8gd instances. For more information, visit the Aurora pricing page.

Agentic AI

Topics

What is pgvector and how does it work with Amazon Aurora?

pgvector is an open-source extension for PostgreSQL supported by Aurora PostgreSQL. You can use pgvector to store, search, index, and query hundreds of billions of vector embeddings generated from AI and ML models — such as those from Amazon Bedrock or Amazon SageMaker.

Vector embeddings are numerical representations of semantic meaning for content like text, images, and video. With pgvector, you can perform efficient semantic similarity searches combined with traditional tabular data in Aurora, enabling applications like:

- Personalized recommendations
- Chatbots and customer service agents
- Candidate matching
- Next-best-action recommendations

For Aurora PostgreSQL, using optimized reads with pgvector increases queries per second for vector search by up to 9x for workloads exceeding available instance memory. Read our blog on vector database capabilities.

Can I use Aurora machine learning to keep vector embeddings up to date?

Yes. Aurora machine learning (ML) exposes ML models as SQL functions, allowing you to call models, pass data, and return predictions using standard SQL. While pgvector requires embeddings to be stored in the database, Aurora ML can make this a real-time process by making periodic calls to Amazon Bedrock or Amazon SageMaker to return the most recent embeddings from your model.

How does Amazon Aurora integrate with Amazon Bedrock?

There are two methods to integrate Aurora with Amazon Bedrock for agentic AI applications:

- Aurora ML: Access foundation models in Amazon Bedrock directly through SQL for both Aurora MySQL and Aurora PostgreSQL.
- Amazon Bedrock Knowledge Bases: Configure Aurora PostgreSQL as your vector store in Amazon Bedrock Knowledge Bases in one click for Retrieval-Augmented Generation (RAG) use cases.

Read our blog and documentation on Using Aurora PostgreSQL as a knowledge base for Amazon Bedrock.

How does Amazon Aurora integrate with Amazon Bedrock AgentCore?

Aurora integrates with Amazon Bedrock AgentCore to help you bring AI agents to production faster. You can persist agent memories within Aurora — a trusted, fully managed database — enabling agentic AI architectures where agents maintain context across sessions. Aurora also works with agentic frameworks such as Letta and LangGraph, as well as AI-coding assistants such as v0 by Vercel.

How does optimized reads for Aurora PostgreSQL improve vector search performance?

Optimized reads with pgvector increases queries per second for vector search by up to 9x in workloads that exceed available instance memory. This is possible due to the tiered caching capability that automatically caches data evicted from the in-memory database buffer cache onto local storage. Read our blog and documentation on how to improve query performance for Aurora PostgreSQL with Aurora Optimized Reads.

Developer Tools

Topics

What are Aurora MCP servers?

Aurora MCP servers provide the flexibility and ease to build and deploy from the tools and agents of your choice. A critical component in agentic AI architectures, MCP servers provide the standardized interface needed for agents to interact with developer tools and your Aurora databases.

Aurora integrates directly with the developer tools you already use, including AI-enabled IDEs such as Kiro and cloud platforms such as Vercel. Aurora also works with agentic frameworks such as Amazon Bedrock AgentCore, Letta, and LangGraph.

What are Kiro Powers for Aurora PostgreSQL?

Kiro is Amazon's AI-powered development environment that can connect directly to Aurora PostgreSQL databases. Developers can use Kiro Powers — guided, reusable workflows — to scaffold full-stack applications with Aurora PostgreSQL as the backend, or to build agentic AI applications that use Aurora as a vector store for Amazon Bedrock Knowledge Bases. Combined with the Vercel v0 marketplace integration, developers can go from idea to deployed application with Aurora PostgreSQL in minutes.

What is the Vercel v0 integration with Aurora?

Through the Vercel Marketplace, developers can access Aurora PostgreSQL Serverless directly using only an email address. New AWS customers receive $100 in free credits applied automatically. v0 by Vercel is an AI-coding assistant that can generate full-stack applications with Aurora PostgreSQL as the database backend, enabling rapid prototyping and deployment without manual database configuration.

What are Amazon Aurora zero-ETL integrations?

Aurora zero-ETL integrations remove the need to build and maintain complex data pipelines. You can consolidate data from multiple Aurora database clusters and run near real-time analytics and ML on operational data. Aurora supports zero-ETL integrations with two targets: Amazon Redshift for near real-time analytics on transactional data, and Amazon SageMaker Lakehouse for building an open lakehouse on operational data.

Both integrations are compatible with Aurora serverless. When combined with Amazon Redshift Serverless, you can generate near real-time analytics without managing any infrastructure. Get started in the Amazon RDS console by specifying the Aurora source and target. Ongoing processing of data changes is offered at no additional charge. See the Aurora zero-ETL documentation for supported versions and Region availability.

When should I use Aurora zero-ETL integration with Amazon Redshift?

Use this integration when you need near real-time access to transactional data for analytics. It allows you to take advantage of Amazon Redshift ML with straightforward SQL commands, materialized views, and data sharing.

When should I use Aurora zero-ETL integration with Amazon SageMaker?

Use this integration to bring data from your operational databases into your lakehouse in near real-time. With the lakehouse architecture of SageMaker, you can build an open lakehouse on your existing data investments without changing your data architecture, and use your preferred analytics tools including SQL, Apache Spark, BI, and AI/ML tools.

How do I get started with Aurora zero-ETL integrations?

Use the Amazon RDS console to create the zero-ETL integration by specifying the Aurora source and target. Once created, the Aurora database will be replicated to the target and you can start querying the data once initial seeding is completed. You can also manage and automate the configuration and deployment using AWS CloudFormation.

How are Aurora zero-ETL integrations priced?

Ongoing processing of data changes by zero-ETL integration is offered at no additional charge. You pay for existing resources used to create and process the change data, which may include additional I/O and storage for enhanced binlog, snapshot export costs for initial seeding, additional storage and compute for processing data replication, and cross-AZ data transfer costs. For more information, visit the Aurora pricing page.

What are Amazon RDS blue/green deployments?

Amazon RDS blue/green deployments allow you to make safer, simpler, and faster database changes. The blue environment is your current production; the green environment is your staging environment that becomes the new production after switchover. Blue/green deployments are ideal for major or minor version upgrades, operating system updates, schema changes, and parameter changes.

During switchover, writes are blocked on both environments until the green catches up, ensuring zero data loss. Guardrails include health checks, replication monitoring, long-running transaction

detection, and configurable maximum downtime (as low as 30 seconds). Your old production environment is preserved after switchover for validation — standard billing applies until you delete it. Supports Aurora Global Database and RDS Proxy. Does not currently support cross-Region read replicas or rollback. See the blue/green deployments documentation for supported versions. For minor version upgrades only, consider Aurora Zero Downtime Patching (ZDP).

Can I use blue/green deployments when I have a blue database as a subscriber/publisher for a self-managed logical replica?

Switchover will be blocked if your blue environment is a self-managed logical replica, or subscriber. We recommend that you first stop replication to the blue environment, proceed with the switchover, and then resume replication. In contrast, if your blue environment is a source for a self-managed logical replica, or publisher, you can continue to switchover. However, you will need to update the self-managed replica to replicate from the green environment post switchover.

Do blue/green deployments support Aurora Global Database?

Yes. Amazon RDS blue/green deployments support Aurora Global Database. To learn more, read the blue/green deployments for Amazon Aurora Global Database User Guide.

How much do blue/green deployments cost?

You pay the same price for running workloads on green instances as you do for blue instances. Effectively, you are paying approximately 2x the cost of running workloads on db.instances for the lifespan of the blue-green deployment. Costs include current standard pricing for db.instances, storage, read/write I/Os, and any enabled features.

What are Trusted Language Extensions (TLE) for PostgreSQL?

Trusted Language Extensions (TLE) enables developers to build high-performance PostgreSQL extensions and run them safely on Amazon Aurora. TLE improves time to market by removing the need for database administrators to certify custom and third-party code before production use. For example, with TLE, independent software vendors (ISVs) can provide new PostgreSQL extensions to customers running on Aurora. You can build functions such as bitmap compression and differential privacy using JavaScript, PL/pgSQL, Perl, and SQL.

TLE offers multiple layers of protection: it limits access to system resources, isolates extension defects to a single database connection, and provides fine-grained permission controls via the rds_superuser role. Deploy extensions using the standard CREATE EXTENSION command from any PostgreSQL client. TLE extensions have access to your PostgreSQL database through the TLE API.

TLE is available on supported Aurora PostgreSQL 14.5 or higher versions at no additional cost in all AWS Regions (excluding China) and GovCloud. To learn more, visit the TLE documentation and TLE GitHub page.

How is TLE for PostgreSQL different from extensions available on Amazon Aurora and Amazon RDS today?

TLE for PostgreSQL extension is included in the set of over 85 PostgreSQL extensions supported by Aurora and RDS. While AWS manages the security risks for each of these extensions under the AWS shared responsibility model, the extensions you write or obtain from third-party sources and install in TLE are considered part of your application code – you are responsible for their security.

Monitoring and Metrics

Topics

- What is CloudWatch Database Insights?
- What is Amazon DevOps Guru for RDS?
- What is the difference between CloudWatch Database Insights and DevOps Guru?
- How does CloudWatch Database Insights collect telemetry?

What is CloudWatch Database Insights?

CloudWatch Database Insights is a monitoring solution that simplifies database troubleshooting by automating telemetry collection — metrics, logs, and traces — without manual setup. Key benefits include:

- Pre-built dashboards and alarms for monitoring database performance
- AI/ML anomaly detection to reduce manual troubleshooting
- Fleet and instance-level views for both high-level monitoring and root cause analysis
- Application context monitoring to correlate database and application performance
- Integration with Amazon CloudWatch Application Signals and AWS X-Ray

What is Amazon DevOps Guru for RDS?

Amazon DevOps Guru for RDS is an ML-powered capability that automatically detects and diagnoses database performance and operational issues, enabling resolution in minutes rather

than days. It analyzes telemetry from CloudWatch Database Insights to identify issues like lock pile-ups, connection storms, SQL regressions, CPU/I/O contention, and memory issues — making database administration more accessible to non-experts.

How can I get started with Amazon DevOps Guru for RDS?

To get started with DevOps Guru for RDS, enable CloudWatch Database Insights through the RDS console, then enable DevOps Guru for your Aurora databases. You can scope analysis to your entire AWS account, specific AWS CloudFormation stacks, or use AWS tags. DevOps Guru publishes insights to the AWS/DevOps-Guru namespace in CloudWatch, so you can monitor anomalies alongside your database metrics in a single dashboard.

What is the difference between CloudWatch Database Insights and DevOps Guru?

CloudWatch Database Insights monitors Aurora resources in real time with customizable dashboards, pre-built alarms, and fleet-level monitoring. It inherits all RDS Performance Insights capabilities and adds application performance integration and correlation of database metrics with logs and events. DevOps Guru uses ML to analyze metrics over time, detect anomalies, and offer insights and recommendations. You can monitor DevOps Guru insights via the AWS/DevOps-Guru namespace in CloudWatch for a single-pane-of-glass view.

How does CloudWatch Database Insights collect telemetry?

CloudWatch Database Insights automates telemetry collection, including metrics, logs, and traces, eliminating the need for manual setup and configuration. By consolidating this telemetry into Amazon CloudWatch, it provides a unified view of database performance and health.

Connectivity

Topics

When should I use the Data API vs. database drivers?

You should use the Data API for new modern applications, particularly those built with AWS Lambda that need to access Aurora in a request/response model. The Data API is an HTTP-based API that eliminates the need to deploy database drivers, manage connection pools, or configure VPC networking.

Database drivers should be used when an existing application is highly coupled with database drivers, when there are long-running queries, or when the developer wants to take advantage of database features such as temporary tables or use session variables.

What benefits does Data API provide?

The Data API simplifies and accelerates modern application development by eliminating database driver deployment, client-side connection pool management, and complex VPC networking between application and database. It automatically pools and shares database connections for better scalability and supports Aurora Global Database writer instances. See the Data API documentation for Region and version availability.

How is the Data API secured?

Users can invoke Data API operations only if they are authorized to do so. Administrators can give a user permission to use the Data API by attaching an AWS Identity and Access Management (IAM) policy that defines their privileges. When you call the Data API, you can pass credentials for the Aurora DB cluster by using a secret in AWS Secrets Manager.

How is the Data API priced?

The Data API includes a free tier of 1 million requests per month (aggregated across all Regions) for the first year. After that, pricing is based on API request volume — see the Aurora pricing page. Data API uses AWS Secrets Manager for credentials (separate charges apply) and logs activity via AWS CloudTrail data events.

Migration

Topics

- How do I migrate a MySQL database to Amazon Aurora (and back)?
- How do I migrate a PostgreSQL database to Amazon Aurora (and back)?

How do I migrate a MySQL database to Amazon Aurora (and back)?

You can migrate between MySQL and Amazon Aurora using several reliable methods. The most straightforward approach is using standard MySQL utilities: mysqldump to export data from MySQL and mysqlimport to import data into Aurora, with the same tools working in reverse for migration back to MySQL.

Alternatively, you can use the Amazon RDS DB snapshot migration feature to migrate an Amazon RDS for MySQL DB Snapshot directly to Aurora through the Amazon RDS console. For large-scale migrations with minimal downtime, AWS Database Migration Service (AWS DMS) supports continuous replication while your source database remains operational. Most customers complete their Aurora migration in under an hour, though migration duration depends on your database format and dataset size. For comprehensive migration strategies, review the Best Practices for Migrating MySQL Databases to Amazon Aurora documentation.

How do I migrate a PostgreSQL database to Amazon Aurora (and back)?

You can migrate between PostgreSQL and Amazon Aurora using several proven methods. The most common approach is using standard PostgreSQL utilities: pg_dump to export data from PostgreSQL and pg_restore to import data into Aurora, with the same tools working in reverse for migration back to PostgreSQL.

Alternatively, you can use the RDS DB snapshot migration feature to migrate an Amazon RDS for PostgreSQL DB snapshot directly to Aurora through the Amazon RDS console. For organizations migrating SQL Server databases to Aurora PostgreSQL, AWS Transform for SQL Server is an AI-powered service that modernizes your SQL Server workloads to Aurora PostgreSQL. Most customers complete their Aurora migration in under an hour, though migration duration depends on your database format and dataset size. For detailed guidance, consult the Amazon Aurora User Guide.

Why customers choose Amazon Aurora

Amazon Aurora customers – from startups to global enterprises – rely on Aurora PostgreSQL, MySQL, and DSQL to power high-performance, data-driven and agentic AI applications. Built to eliminate the operational burden of managing database fleets at Amazon.com scale, Aurora delivers availability and scalability that Intuit and Netflix depend on to serve customers worldwide. Now, Bio-Rad Laboratories, LeadSquared, CORTO, and SurveySparrow are using Aurora to drive hybrid search and agentic AI workloads to ground agents in their operational data, enabling agents to reason over truth. It also provides memory for agents to recall past interactions and deliver more context aware and personalized conversations.

Get started with Aurora

Connect with an Aurora specialist

Aurora PostgreSQL

Amazon.com

"Implementing and scaling hundreds of on-premises instances across the database fleet consumed all database team member resources for many weeks and was a relatively low value-add task for database administrators. In the Oracle world, a seemingly simple change such as scaling from a medium to a large database instance required provisioning hardware, standing up primary and standby databases, and managing failover during transitions, which could take a full day for each instance. Not to mention the fact that we were using specialized hardware that had to be ordered months in advance. After migrating to Amazon Aurora, provisioning additional capacity is achieved through a few simple mouse clicks or API calls."

Brent Bigonger, Senior Database Administrator - Amazon Fulfillment Tech

Watch the video >>

Amdocs

"Prepping and getting a database cluster up and running took three weeks for installing, confirming network, testing for latency. On Amazon Aurora we're now able to effectively do it in a day."

Jay Deen, CTO - Amdocs Media

Read the case study >>

Best Western International, Inc.

"The hotel industry is rapidly changing, as more customers expect the ease and convenience of mobile computing. Moving to AWS brings our organization to the forefront of innovation and allows us to give our guests fast, reliable and secure data processing so they can organize their trips, change their reservations, and book their stay with us."

David Kong, President & CEO - Best Western

Bio-Rad Laboratories

"Our Unity Next Peer Quality Control (QC) solution provides labs with instant, centralized access to peer reporting through an intuitive platform. By simplifying access to QC insights, we empower labs to refine their processes, mitigate errors, ensure regulatory compliance, and prioritize patient safety with confidence. To address key challenges—driving product adoption, educating users on features, and supporting multiple languages— Bio-Rad enables customers to engage with a chatbot that references product and user manuals to deliver answers directly from these documents, complete with feature-specific image references. We wanted to build a generative AI solution aimed at enhancing our support services and speeding up product feature education and adoption.

We turned to AWS and built our Retrieval-Augmented Generation (RAG) solution leveraging Amazon Aurora PostgreSQL-Compatible Edition with the pgvector extension, integrated with Amazon Bedrock using the Claude V3 Haiku LLM, Amazon Titan Embeddings text models, and AWS Translate to facilitate multi-language support for 14 different languages in Unity Next Peer QC. Unlike traditional support methods, our chatbot leverages product and user manuals to deliver precise answers, complemented by feature-specific visual references. This approach reduces reliance on support teams while accelerating user education. With this innovation, we see a 20% reduction in customer call volume and improved feature adoption, driving better outcomes for labs worldwide."

Jasmina Desai, Associate Director, Software Architecture and Development - Bio-Rad Laboratories

BMC

"The idea of having a company like Amazon Web Services stand behind an open source product and take care of important attributes like high availability, scaling, and overall data management is huge. Amazon Aurora provides the open source database management layer we've been seeking."

Raj Cheruvu, AVP R&D, Helix ITSM - BMC

Read the case study »

BMLL Technologies Ltd

"Amazon Aurora PostgreSQL is helping us solve performance and scalability challenges to provide deeper and faster data analysis to our customers. In addition, the built-in security, durability, and high availability capabilities of Aurora PostgreSQL help to simplify and automate most of our DBA requirements, lowering our costs while increasing reliability."

Dr. David Robinson, CTO - BMLL Technologies

British Airways

"Thanks to Amazon Aurora serverless and Data API, we can now parallelise our data analysis and processing within AWS Lambda. Instead of executing our data science workloads sequentially, we can run them concurrently. Lambda can seamlessly scale from zero to 200 or 300 functions running concurrently, and the Data API efficiently manages connection spikes. Consequently, data analysis for a single flight is now typically completed in less than 30 seconds, a significant improvement from the tens of minutes it used to take."

Nils Mohr, Senior Flight Data Software Engineer - British Airways

Cloudability

"Amazon Aurora has become the database of choice for all our MySQL workloads. Our True Cost Engine delivers cost efficiency to customers, using a new predictive model based on petabytes of customer cost optimization data. To support this kind of analysis, we need a database that is performant, scalable, cost-effective, and easy to maintain and tune. Aurora solves our biggest issues with managing large, production databases with strict SLAs. Now that PostgreSQL is supported, we expect all our PostgreSQL workloads to move to Aurora as well."

Matt Finlayson, Vice President of Engineering - Cloudability

CloudZero

"Amazon Aurora serverless and Data API enable us to avoid managing complex VPC's and network access controls which significantly reduces our operational complexity and security burden. This

means being able to focus on hiring more software engineers and less operations folks. We think of this as living the "Serverless lifestyle" where we rely on AWS for ensuring our uptime and availability which allows us to move faster with less headcount costs."

Erik Peterson, CTO/CISO & Founder - CloudZero

CORTO

"We began using Amazon Aurora PostgreSQL-Compatible Edition with the pgvector extension in 2023 as part of CORTO's AI-powered legal technology platform, which helps lawyers achieve faster and more effective results by automating legal tasks. With pgvector and HNSW indexing, we can store, index, and query high-dimensional embeddings to power semantic search across more than 2.5 billion documents. These capabilities allow us to deliver highly relevant insights in milliseconds while keeping our entire data stack within AWS, simplifying operations, strengthening compliance, and improving performance. The scalability with Aurora has allowed us to seamlessly manage fast-growing datasets, including 46 terabytes of data in our APAC cluster and 7.6 billion vectors in our primary embeddings table. By consolidating both relational and vector workloads in a single high-performance environment, we've significantly reduced infrastructure complexity and accelerated product development."

Anisa Dean, Senior DevOps Engineer - CORTO

DriveWealth

"After DriveWealth successfully migrated the production workload from CockroachDB to Aurora PostgreSQL, we improved our read/write throughput up to 5X, enhanced data consistency, and achieved an 80% cost reduction. We were impressed with the AWS team's ability to resolve our challenges from working with previous vendors – they earned our immediate trust by recommending Aurora."

Venkat Vadlamudi, Engineering, Data & Analytics Leader - DriveWealth

Decisiv

"Aurora PostgreSQL is instrumental for Decisiv as we upgraded our infrastructure to support the growing scale, performance and reliability requirements that our customers expect. We've consolidated multiple SQL Server instances into a single database that boasts higher performance at a better cost. Thanks to Aurora, our migration from SQL Server to PostgreSQL went more smoothly than we anticipated, and we are now well-positioned to rapidly scale."

Satish Joshi, CTO - Decisiv

Easygo

"The highly variable nature of these sport-seasons workloads meant Easygo saw a rapid scale increase on top of its usual traffic patterns. We wanted the engineering team to spend less time managing database scaling requirements, and a database solution that could seamlessly scale to handle sporting events worldwide in different time zones. Easygo migrated roughly 50 databases over to Amazon Aurora serverless, enabling the team to focus on building bigger and better games, while spending less time investigating bottlenecks."

Director of Engineering - Easygo

Emma

"At Emma, we're focused on thoughtful product design and friendly expert services to help marketers do their very best work. We manage several multi-terabyte OLTP databases that process nearly 300 million transactions per day. Aurora PostgreSQL provides us with a highly performant, secure, and more scalable database cluster than we are able to affordably create in a traditional datacenter. Aurora PostgreSQL also allows us to be much more responsive when meeting expected business growth demands for the foreseeable future, all while reducing the time needed to manage, maintain, and scale the database systems."

Marc Powell, Director of Infrastructure - Emma

FireEye

"With Amazon Aurora, we were able to bring Detection On Demand to the market in a matter of months thanks to its serverless architecture and fully managed database services."

Martin Holste, CTO Cloud - FireEye

Read the case study and watch the video >>

Guidewire Software

"Compared to commercial databases, we observed similar or better performance at the 90th percentile. But at the 99th percentile, we saw even better response times."

Kevin De Yager, Senior Product Manager for InsuranceSuite Cloud - Guidewire

Read the case study and watch the video >>

Globe Telecom, Inc.

"The shift out of Oracle will save us an estimated $1 million over the next three years in licensing fees—funds that Globe Telecom can use for new digital transformation projects. Performance has been solid—exceeding the latency target of 40 milliseconds."

Melissa Banzon, Head of ISG Transformation Office - Globe Telecom

Goldsky

"Goldsky delivers blockchain data to leading stablecoin issuers and fintechs requiring sub-second freshness for payments, treasury, and compliance. Our real-time platform ingests 20+ million events per second from 120+ networks into Amazon Aurora PostgreSQL, the backbone of our streaming pipeline. Aurora handles our heavy write velocity — over 100,000 commits per second at peak and ~50,000 sustained 24/7 — with bulk inserts of stablecoin transfers and on-chain transactions driving 1M+ write IOPS and 1.5M+ read IOPS at peak. With Aurora read replicas, we can serve low-latency queries to hundreds of customer applications across ~8,000 concurrent writer connections, while transactional semantics handle chain reorganizations atomically, ensuring fintechs never see stale or inconsistent ledger data.

Aurora auto scaling means we never pre-provision capacity as our customers expand into new networks or payment corridors. With Aurora I/O-Optimized, we get cost predictability as our transaction volume grows, meeting high-TPS workloads in a way no alternative we evaluated matched. We now run our streaming engine at billions of messages per month, scale that would be unmanageable on self-hosted PostgreSQL, delivered on Aurora PostgreSQL with operational excellence that stablecoin issuers and regulated fintechs require. With Aurora PostgreSQL our small engineering team operates at massive scale without dedicated database administrators, so we can focus on building the data infrastructure that powers the next generation of stablecoin and financial applications, not on managing infrastructure ourselves."

Jeff Ling, CTO - Goldsky

INRIX

"From raw GPS points, INRIX generates large-scale vehicle movement data and ingests the data into sharded Amazon RDS for PostgreSQL instances. We are hitting the storage and performance limits per shard and looking for a more scalable solution. With Amazon Aurora's compatibility

with PostgreSQL, we've seen three times performance improvements in our benchmarks. We love Amazon Aurora's ability to scale storage independently of computing resources at better price points."

Trang Nguyen, Senior Software Engineer - INRIX

Intuit

"Looking to reduce operational overhead and improve our database costs, we decided to utilize Amazon Aurora serverless. Mixed-configuration clusters are perfect for our use case, which allows us to use a combination of provisioned Amazon Aurora writer and Amazon Aurora serverless readers within the same cluster. With Aurora serverless, we get the benefits of automatic scaling without compromising on our requirement for high availability and disaster recovery. We use Aurora serverless in both our production and non-production environments, and it has helped us save approximately 55% on our database provisioning costs. With Aurora serverless, our developers can focus more on building features and less on managing capacity."

Rajesh Saluja, Principal Data Engineer - Intuit

LeadSquared

"As our customers started demanding faster onboarding of our chatbot, we wanted our chatbot solution to offer an easy setup, provide a personalized experience (based on customer specific data and intent), and automate repetitive tasks better. The integration of Retrieval-Augmented Generation (RAG) capabilities using Amazon Aurora PostgreSQL with the pgvector extension and LLMs available in Amazon Bedrock has empowered our chatbots to deliver natural language responses to out-of-domain inquiries, enhanced dialogue management, and reduced our manual efforts. With RAG, we can retrieve data from outside the LLM, for example from the website, knowledge base, or help documentation, and augment the prompts by adding the relevant retrieved data in context. RAG capabilities have made our chatbot better by allowing the system to provide natural language answers to questions that aren't in or a variation of the intent list. Consequently, we have observed a 20% improvement in customer onboarding times."

Prashant Singh, COO & Cofounder - LeadSquared

MoeGo

"MoeGo is a SaaS provider offering all-in-one solutions that include business management, customer communication, and digital payments. As such, we use a multi-tenant architecture and

face significant traffic fluctuations. We have adopted Amazon Aurora serverless across our core business scenarios, including appointment management, financial processing, and CRM to address our business need of 10x traffic variation between peak and off-peak hours. We leverage Aurora serverless for its exceptional elastic scaling capabilities to achieve intelligent adaptive allocation of database resources, significantly reducing our manual operational overhead and enabling our team to focus on product innovation and business development. Aurora serverless has a pay-as-you-go model, which has delivered 30% cost savings compared to traditional peak-capacity fixed instances, providing both a flexible technical foundation for global expansion and continuous delivery of high-quality digital services to our customers."

Zihao Yin, Platform Engineering Lead - MoeGo

New Innovations

"Thanks to AWS and Amazon Aurora PostgreSQL, our company has been able to build an infrastructure that scales to meet our customers' demands. We found that Aurora PostgreSQL is a drop-in replacement for Amazon RDS PostgreSQL, with a few very important improvements: write throughput and automatically-expanding storage. We migrated 700+ instances of Microsoft SQL Server, and LOVE the simplicity of management that Aurora PostgreSQL provides. Gone are the days of dealing with tuning and tweaking configuration files for optimal performance."

Stephen Sciarini, IT Manager - New Innovations

Nielsen

"In our testing of Amazon Aurora PostgreSQL in the preview, we have seen very good performance upwards of 7-11 times that of RDS PostgreSQL, for both write and read/write workloads. We are also excited about the expected scalability and reliability, giving us great confidence that Amazon Aurora PostgreSQL will meet our requirements as we move some of our core database workloads to AWS."

Todd Lightbody, Watch Architecture Leader - Nielsen

ProQuest

"We've gotten so many benefits from migrating our database from Oracle to Amazon Aurora such as high availability and easier scaling for reads. All of these benefits come out of the box and at a lower cost and with far less licensing complexity. For data migration, we used the AWS

Database Migration Service (DMS). It only took 24 hours to migrate 1.25 Terabytes of data with our first attempt. We worked with the AWS team who helped us experiment with parallelization and organizing the migration by breaking it down to multiple tasks to get the final production data migration down to four and a half hours!"

Suresh Karri, Director of Technology - ProQuest

RavenPack

"We are enhancing our products by incorporating new heavily structured data such as market consensus estimates, products relationships, and supply chain information, to name a few. Many of our analytics apps have to access this data in real-time and maintain point-in-time sensitivity, which requires a fast performing and heavily relational database, resilient to failures, and distributed across multiple AZ's. In our testing, Aurora PostgreSQL has performed significantly faster than standard PostgreSQL, and has shown high compatibility with standard PostgreSQL. Given that we already have a deep usage of Amazon Redshift, and we are used to PostgreSQL interfaces, Aurora PostgreSQL will perfectly fit into our system."

Jason Cornez, CTO - RavenPack

ResultsCX

"We moved from SQL Server to using Amazon Aurora PostgreSQL as an Enterprise Data Store (EDS) and have continued to expand its reach within the business. Building our EDS in Aurora PostgreSQL allowed us to reduce complexity through automation and consolidation, while producing more accurate and stable reports for business users. It also allowed us to substantially reduce our licensing costs as Aurora PostgreSQL is an open-source solution. Finally, the high availability and disaster recovery capabilities of Aurora PostgreSQL help us to deliver on the SLAs that our business users expect.

The ability to retrieve data at a high speed for developing ad-hoc analysis for near real-time issues has increased our speed to customer, vastly. We can now identify valuable insights over larger data sets, instead of selected sampling, utilizing Aurora PostgreSQL. We have been able to reduce our total number of reports from 3,000 to 1,000, decrease time spent managing the legacy reporting platform by 30 percent, and, most importantly, resource time spent on manual reporting historically is now being utilized to develop valuable skills in other areas that provide higher value for career development."

Dr. Jim Sullivan, Vice President Information Systems & Enterprise Applications - ResultsCX

RocketReach

"Our team has enjoyed the scalability benefits of Aurora PostgreSQL, which has allowed us to quickly ingest significant amounts of data without negatively impacting customers. The addition of the Amazon Aurora I/O-Optimized feature has stabilized our daily costs at RocketReach, reducing our total monthly Aurora costs by 60% and allowing our team to focus on business priorities rather than cost management efforts. We are now able to reap the scalability benefits of Aurora without having to worry about cost increases as our traffic scales.

Aurora I/O-Optimized was a critical addition for our team at RocketReach, not only did it allow us to leverage the advantages of Aurora more economically, it also helped prevent unplanned work across our teams. We will absolutely continue to leverage Aurora I/O-Optimized when the use case aligns with its capabilities. At RocketReach, we have many workloads that require substantial amounts of data to be processed efficiently while maintaining performance for our customers. For these use cases, Aurora I/O-Optimized allows us to balance cost and performance, while rapidly delivering value to our customers."

Jeremy Livingston, CTO - RocketReach

Shippo

"Our service is 24/7 and our transactional database is highly critical such that any unavailability will result in service downtime. We wanted to upgrade our PostgreSQL database to Aurora, but the traditional upgrade approach required a two-step process, with each step resulting in a multi-hour outage - far too much impact to our business and customers. Looking to the unconventional, we explored using AWS DMS for the upgrade. After completing a very positive POC with outstanding support from AWS DMS team, we decided to go ahead with DMS and were able to skip the intermediate PostgreSQL versions and migrate directly to Aurora. DMS was stress-free and reduced our upgrade downtime to minutes. Post-upgrade, Aurora has proven to have better availability, performance, and scalability which gives us the comfort that we can continue to support Shippo's rapid growth."

Calvin Xu, Data Architect - Shippo

SBI Sumishin Net Bank

"We are an internet-only bank based in Japan. Our databases are critical to our ability to provide excellent personal banking and investment services. Since we migrated from Oracle RAC to

Amazon Aurora, we have cut our database management costs by 83% while also getting 50% better performance. We've benefitted from improved speed, availability and scalability."

Shinichi Aikawa, Director of System Development 2 - Sumishin SBI Net Bank

SRA OSS

"In our performance testing of Amazon Aurora's PostgreSQL compatibility, we found that the performance was three times better than standard PostgreSQL. Our testing also showed that Amazon Aurora is fully compatible with PostgreSQL 9.6, and we believe customers will be able to move large enterprise workloads from on-premises commercial databases to Amazon Aurora because of its high performance, high availability, and PostgreSQL compatibility. SRA OSS will add support for Amazon Aurora to the next version of pgpool-II, which provides clustering management middleware for PostgreSQL."

Tatsuo Ishii, Japan President - SRA OSS, Inc.

SurveySparrow

"At SurveySparrow, we operate five business units powered by agentic AI integrated throughout our product stack. Our newly launched agentic AI-driven customer experience capabilities for customer support and sales demanded high-performance vector search capabilities combined with seamless transactional data access—making the database layer mission-critical. We migrated from standalone vector databases like Pinecone to Amazon Aurora PostgreSQL with pgvector, consolidating both vector embeddings and relational data into a unified database platform. This architectural shift eliminated the complexity of managing separate systems while delivering the performance and scalability our AI agents require. It gave us immediate wins: no cross-service hops, simpler architecture, and deeper integration with AWS services. With Aurora serverless, we have cut costs by nearly 50% and improved query latency by an estimated 30% for our agentic AI use cases. Storing embeddings and metadata in the same tables has made our design cleaner, and auto-scaling handles our spiky AI workloads without any manual effort. The best part is our teams can focus on building better AI, not juggling multiple databases, while our customers benefit from faster, more cost-efficient experiences."

Jagadeesh Rajarajan, Head of AI - SurveySparrow

Tenable

"Tenable relies on AWS to provide the flexibility and scalability needed to run its large-scale Tenable.io vulnerability management platform. Thanks to Aurora, Tenable can more easily handle

large and complex unscheduled queries from customers in seconds without breaking the bank, unlike with standard PostgreSQL instances. Aurora's cost model for IO and storage means Tenable's usage and costs are correlated. Plus, what previously took 10+ minutes on a similarly sized RDS instance takes a few seconds on Aurora, and in cybersecurity, every second counts. Tenable uses Aurora for workloads with large-scale unpredictable queries, and ultimately, Aurora enables Tenable to better serve our customers."

Scott Hirleman, Cloud Infrastructure Cost Manager - Tenable

TIBCO

"We are an early adopter of Amazon Aurora PostgreSQL and used the AWS Database Migration Service to transition TIBCO Cloud Live Apps to Amazon Aurora seamlessly, while it was in production, without our customers noticing any interruption in our service. Amazon Aurora's reliability, security, and fast failover will continue to help us scale Live Apps, giving customers constant access to our service so they can build and run apps quickly and with high availability."

Matt Quinn, Chief Operating Officer - TIBCO

Tokio Marine & Nichido Fire Insurance

"We applaud AWS for providing the extended support for Amazon Aurora and Amazon RDS. AWS is a company that listens to the voice of user companies and makes them come true. This is one of the biggest reasons we use AWS. We want to continue to strengthen our partnership for mutual growth and development."

Hiroki Yamashita, Manager - Tokio Marine & Nichido Fire Insurance Co., Ltd.

Verisk Analytics, Inc.

"We moved our Oracle and SQL databases to Amazon Aurora to improve the speed, latency, and processing times of our databases. The databases migration to Aurora PostgreSQL took less time and resources than we anticipated. Verisk 3E has compliance and performance requirements from our customers to run 3E Insight in various regions globally. Thanks to Amazon Aurora, we can now deliver a highly responsive, scalable, and highly available intelligent compliance solution to our worldwide customers. The migration gave us a significant cost reduction and improved our ability to deliver value to our customers."

Ashish Verma, Sr. Director of Software Engineering - Verisk 3E

Verizon

"Verizon is helping our customers build a better, more connected life. As part of this journey, we are undergoing a major transformation in our database management approach, moving away from expensive, legacy commercial database solutions to more efficient and cost-effective options. Testing of Amazon Aurora PostgreSQL showed better performance over standard PostgreSQL residing on Amazon EC2 instances, and the AWS Database Migration Service and Schema Conversion Tool were found effective at identifying areas for data-conversion that required special attention during migration."

Shashidhar Sureban, Associate Director, Database Engineering - Verizon

Vindicia

"Beyond defraying the costs, the move to AWS makes the tedious on-premises jobs we worry about go away. We look forward to using the capabilities of Amazon Aurora to access data more efficiently."

Steven Azar, Senior Manager of Data Programs - Vindicia

Read the case study »

Wappa

"To help our customers reduce corporate travel expenses, our platform needs to find rides quickly and accelerate the budgeting, payment, and reporting processes. Since migrating our Oracle database to Amazon Aurora, our user validation process has become 60 percent faster, reporting time per user has dropped 75 percent, and the payment process is 70 percent faster. We're clearly seeing the results in our user growth numbers and user satisfaction ratings of our application."

Cesar Matias, Chief Technology Officer - Wappa

Xata

"Xata has customers that require tenant isolation and want to build AI-driven applications that deliver low latency, highly relevant vector similarity searches to their users. In one particular case, Xata migrated a customer from Pinecone to using Amazon Aurora PostgreSQL-Compatible Edition with pgvector and realized a 65% cost reduction while meeting our query latency targets and

providing additional functionality by being able to store the metadata in the same tables as the vector data. This resulted in a significant simplification of their application architecture."

Monica Sarbu, CEO & Founder - Xata

Aurora MySQL

8x8

"Amazon Aurora allowed me to do something innovative. We didn't have to wait for infrastructure to be procured and provisioned, and we didn't have to free up DBAs."

Lance Johnson, Director of Engineering Cloud R&D - 8X8

Read the case study »

Alert Logic

"It's a very rapid process from "let's do it," to the proof of concept, to getting it done on Amazon Aurora. We have one DBA working on it and another person helping him, and we expect to have a production workload running at scale in under two months."

Paul Fisher, Chief Architect - Alert Logic

Read the case study »

Agora

"We provide developers with simple, flexible, and powerful APIs to embed real-time voice, video, interactive streaming, chat, and AI capabilities into applications. We created the Conversational AI Engine—a solution that makes human-like conversations with AI possible. Our SD-RTN™ (Software Defined Real-Time Network) has network nodes covering over 200 countries and regions globally. Amazon Aurora attracts us with its high performance, innovative features, and advanced architecture. Using Aurora Global Database with Amazon Aurora MySQL-Compatible Edition we have achieved 99.99% service availability through multi-Region deployment, with user login success rates exceeding 99%. In addition, using Aurora Global Database we have implemented proximity-based routing and service delivery for users located in different countries and achieved reduced latency. With blue/green deployments support for Aurora Global Database, we have

also effectively simplified our maintenance operations to just 3 clicks for upgrades and reduced downtime to within 5 seconds."

Tony Wang, Co-founder & CRO - Agora

EPSi

"Aurora has provided operational value, speed to market, cost, and ultimately customer value. AWS constantly provides us with new innovation that you can't build internally...EPSi has done a great job using technologies like AWS to meet clients where they are and take that next step as a thought leader... The only way we can do that is with robust back-end and analytics and performance that can only be built on AWS to meet those demands."

Grant Gordon, CTO - EPSi

Watch the video »

Akatsuki Inc.

"Amazon Aurora's fast failover capabilities and storage durability have minimized technical obstacles for our online gaming service. After launching Romancing SaGa Re;univerSe, which achieved more than 10M downloads in the first 3 weeks, Aurora's high durability brought us continuous and stable service operation without any maintenance emergencies. As the backend database for our gaming service, Aurora has provided invaluable reliability and stability."

Yuto Komai, Server Side Engineer - Akatsuki Inc.

Alfresco

"Amazon Aurora was able to satisfy all of our scale requirements with no degradation in performance. With Alfresco on Amazon Aurora we scaled to 1 billion documents with a throughput of 3 million per hour, which is 10 times faster than our MySQL environment. It just works!"

John Newton, Founder & CTO - Alfresco

Arizona State University

"ASU continues to leverage the AWS cloud as a means to improve our agility and capacity in meeting our customer needs. One such AWS service we're bullish on is Amazon Aurora. We're

moving much of our on premise database workload, currently MySQL, Sybase, Oracle, and Microsoft SQL Server to Aurora or RDS. Aurora also plays an integral role in our data lake strategy, along with Redshift and S3, as we build out our new advanced analytic platform. Data behind our ASU Alexa skill and ASU mobile app benefits from Amazon Aurora's scalability and high performance as enormous concurrent student users during the school year creates new peak loads each season. And, we see Amazon Aurora serverless as a next step in our cloud maturity to help us improve development agility while reducing costs on infrequently used systems, to further optimize our overall infrastructure operations."

John Rome, Deputy Chief Information Officer - Arizona State University

Autodesk

"Our Account Control Management (ACM) application is a central component for many Autodesk applications, and usage has grown this year. ACM requires a high performance database with low-lag replication, automatic scaling, and MySQL compatibility. Amazon Aurora allowed us to increase our database connections and greatly improve our scalability, response times, and CPU utilization. Our customers' experience improved in a short time so we're looking into Aurora for additional applications."

Brian Mathews, Platform Engineering Vice President - Autodesk

Read the blog and watch the video »

Beachbody

"To support the transition from DVD sales to our Beachbody on-demand streaming service, we needed a cloud service that provides agility, scalability, and high availability. Amazon Aurora provides the foundation for our digital services, supporting over a million users every month with peaks of over 250,000 users in a single day. Aurora also makes live events possible, as it allows us to scale up quickly for popular live workouts and then scale down, so we keep our costs low while our customers enjoy fast, reliable service."

Grant Leathers, Vice President, Technology Operations - Beachbody

Capital One

"At Capital One, our cloud-first approach led us to start testing the preview of Amazon Aurora's PostgreSQL compatibility in November 2016. We anticipated performance benefits, as well as the

high availability and fast failover capabilities ideal for serving our customers. In our testing during the preview, we've been impressed with the performance and high availability offered by Amazon Aurora."

John Andrukonis, Chief Architect - Capital One

California Polytechnic State University

"Our Learning Management System (LMS) is central to the student experience at Cal Poly, and we have high standards for performance, scalability and high availability. Amazon Aurora meets our high standards. Given LMS usage patterns, with peaks during the first week of class, mid-terms, and finals, we expect to economize by paying only for the resources we need when we need them and eliminating the need for on-premises hardware. We also capitalize on the flexibility of the AWS environment that allows us to automatically shut down development and test instances when not in use, and spin them up quickly when needed."

Alison Robinson, Associate Vice President - Cal Poly Information Technology Services

Choice Hotels International, Inc.

"At Choice Hotels, we process millions of requests every day for guests looking to stay at our properties. Transaction volume has grown over 300 percent in the last three years, and sets new daily records every summer. To accommodate this growth, we undertook a significant challenge and developed the first new global reservation system from a hotel company in over 30 years, and built it on AWS. Leveraging Amazon Aurora and other AWS services allowed us to both exceed our performance requirements and lower our costs."

Brian Kirkland, Chief Technology Officer - Choice Hotels

The Canadian Football League (CFL)

"We migrated from Microsoft SQL Server to Amazon Aurora to take advantage of Aurora's scaling capabilities, which couldn't be better suited to meet the spikes in traffic we get during the football season while also allowing us to minimize costs. We get all of the capacity we need, but don't pointlessly run at high capacity on non-game days where our needs are much, much smaller. We also pair Aurora with Elastic Beanstalk, which lets us mirror that optimization of performance and cost on the server side, and has helped us become more productive, spending less time on DevOps and more time writing code and creating value for the league."

Sully Syed, Senior Director of Technology - Canadian Football League

Domo

"Amazon Aurora made my life easier. I was dealing with three possible system degradation alerts each night after hours, up to 15 or 20 some nights, that all had to be investigated. Now I am typically alerted once or twice a week."

Bret Bills, Director of Engineering - Domo, Inc.

Read the case study »

Dow Jones

"Dow Jones' first legacy on-prem database to Aurora migration was a high-profile workload that plays a critical role in engaging and retaining our customers. We were an early adopter of Aurora and the AWS Database Migration Service, which we're utilizing to move data from our legacy on-prem environment to Aurora. The migration allowed us to replace a legacy platform with performance challenges that required 400k in DBA staff to manage, with $1M in licensing costs annually, to a cloud-based, highly scalable and resilient solution. It uses a 1TB Aurora cluster with 200 transactions per second, along with replication to another region for high availability and disaster recovery. By moving much of the operational overhead to AWS, and eliminating the need to manage storage completely, Aurora frees up funds for innovation."

Ramin Beheshti, Chief Product & Technology Officer - Dow Jones

DHI Group, Inc.

"AWS gave us the opportunity to build an enterprise-class, cost-effective, open source-based database strategy with Amazon Aurora. We achieved the database freedom our team needs to deliver specialized insights and relevant connections to our customers."

Brian Hostetter, Senior Director - DHI Group

Eduphoria

"We were very excited after testing our workloads on the initial Amazon Aurora beta because of its performance potential. When we moved our production systems from Amazon RDS to Aurora, we found that we were able to double the number of customers per database cluster while increasing performance at the same time. Aurora's flexibility with adding read slaves on demand as well

as automatically managing all underlying storage lets our engineers focus more on customer experience and less on how to speed up data access layers."

Aaron Dulaney, Director of Infrastructure Operations - Eduphoria

FirstCry

"At FirstCry, a smooth customer experience is key for us to build trust and strong relationships with our customers. Using Amazon Aurora has helped us increase our agility, scalability, and resilience as we scale to meet our customers' needs. As our application is I/O-intensive, we were looking to eliminate variability in our I/O spend. With the adoption of the Aurora I/O-Optimized feature, we have realized 50% cost savings per month, achieving price predictability and improved price performance for our I/O heavy workloads.

Based on the great experience we have had using the fully managed service, we are migrating more of our self-managed databases to Aurora. We have also made both Aurora MySQL- and PostgreSQL-compatible editions core pillars of our data strategy, along with Amazon Redshift and Amazon S3. We look to further enhance our Analytics and AI/ML capabilities in the organization utilizing these services. We are very pleased with the performance that Aurora has delivered and it will continue to be a trusted service we utilize as we scale our partnership with AWS."

Sumit Kumar, Sr. Director Cloud Engineering - FirstCry

FirstFuel

"Amazon Aurora has immediately become a key component to meeting FirstFuel's uptime and disaster recovery Service Level Agreements (SLAs) with some of the world's most demanding enterprise clients. Key functionality such as restore to point in time, reader nodes, cross-region replication, and automatic node failover have dramatically reduced or eliminated our data loss and downtime risks. Aurora has also reduced the time we spend on database management, leading to real productivity improvements."

Siddharth Kundalkar, Director of Engineering - FirstFuel

Funny Or Die

"Our first tests of Aurora were difficult to believe because the performance increase was substantial. Aurora has greatly improved the responsiveness of our web site and our data warehouse analytics. After migrating to Aurora, we were able to rapidly add more read slaves to our site as needed, while maintaining great performance. Overall the performance and throughput

have been excellent and Aurora's performance metrics are very detailed and helpful, simplifying database management and scalability. Aurora made our migration from traditional colocation to AWS easier because the storage was fully managed and replication was extremely fast."

Mark Smallcombe, CTO - Funny Or Die

GungHo Online Entertainment

"Amazon Aurora improves the backend database performance of our online gaming services and reduces the operational cost. For instance, Update query performance runs 8 times faster and Select query performance runs about 10 times faster on Princess Punt Sweets, a long-running gaming title from 2012. In addition, backup time was shortened by 80%, and we achieved a 40% reduction in our AWS cost by reducing the number of instances. After the migration, Aurora continues to achieve high reliability and stability as our backend database."

Satomi Sakashita, Engineer of Development Division - GungHo Online Entertainment, Inc.

HungryPanda

"We began using AWS in April 2024 and chose Aurora, a MySQL-compatible and fully managed database service, to support our core business - food delivery platform. From the initial deployment, we opted for Graviton2 (R6g) instances due to competitive pricing, 12% lower compared to Intel x86 instances. As we expanded to more countries, we started to hit the limits of our current r6g instance class. While there was always an option to scale vertically, this would almost double costs. The launch of Graviton3 (R7g) instances offered a solution. Switching to Graviton3 (R7g) instances brought significant improvements: 22% lower CPU utilization, 27% reduced latency, and fewer reader instances(5 to 4). This transition allowed us to accommodate our growing business needs while optimizing costs and performance."

Li Chong, Vice President of Technology - HungryPanda

Intuit

"Intuit recently migrated their commerce platform to Amazon Aurora MySQL to support increasing global demand. All direct purchases of Intuit's software will go through the Intuit's Commerce Platform running on Aurora, with TurboTax already live to meet traffic demands during tax season. A large portion of our workload involves low-latency, read-only access to data. An example is pricing information, which is infrequently updated but needs to be readily available for reads from coast to coast.

Aurora Global Database, with sub-second global replication, enables us to address this business requirement without performance or latency constraints. As a financial services company, we also care deeply about business continuity even in the face of large-scale events. Aurora Global Database allows us to maintain a strong disaster recovery posture by distributing data across AWS regions with failover typically taking under a minute to complete."

Krishna Vaishnav, Engineering Manager E-commerce & Cloud Platform Engineering - Intuit

ISCS

"We project that the size and throughput requirements of our relational databases will more than double year-over-year for the next several years. We need a scaling strategy that delivers reliable performance with growth, yet is simple to operate. Amazon Aurora's ability to auto-grow database sizes all the way up to 64 TB, as well as provide consistent throughput and latency, is an extremely promising approach to reducing our operational risk."

Doug Moore, CTO & VP of Consumer Experience - ISCS

Ivanti

"We're evaluating migrating our relational datastore from SQL Server to Aurora. The most important thing to us is the flexibility...whereas with SQL Server...it's very cumbersome. The goal is to introduce a modern relational data store that allows us to scale at will as our customer onboard."

Jared Neilsen, Sr. Director Cloud Ops & Devops - Ivanti

Jobvite

"We were fortunate to have had the opportunity to partner with AWS during the development/rollout of the DMS migration service to migrate from Microsoft SQL Server to Aurora MySQL. In performing this migration, Jobvite realized cost reduction of 40%, improved responsiveness to customer requests by up to 40%, eliminated monthly database maintenance that took the application down for 3 hours, and reduced refresh time from 20 hours to 2 hours. All of these benefits have allowed Jobvite to invest efforts into other aspects of the business to improve customer satisfaction."

Chaitanya Konduri, DevOps Manager - Jobvite

Read the blog »

The Mainichi Newspapers

"We migrated all systems of the Mainichi Shimbun News website to AWS, and celebrated the first anniversary in December 2016. We use Amazon Aurora for the core database of our system, and it shows excellent performance. The database has been running with 100% availability in the last 16 months, and we spent almost no time on maintenance and administration. By having such a wonderful database service, engineers could focus on developing better applications, allowing us to provide better service to our customers. As a result, annual page views on our website increased by 20% from the previous year."

Yuzi Mori, Solutions Architect, Digital Media Division - The Mainichi Newspapers Co. Ltd

Netflix

"We were able to test Aurora's parallel query feature and the performance gains were very good. To be specific, for queries doing full table scan or fetching fat indexes with billions of rows, we noticed the query time reduced from 32 minutes to 3 minutes. We were able to reduce the instance type from r3.8xlarge to r3.2xlarge. For this use-case, parallel query was a great win for us."

Jyoti Shandil, Cloud Data Architect - Netflix

Packagecloud

"Using Amazon Aurora MySQL-Compatible Edition, Packagecloud has scaled our databases to hundreds of millions of rows. Previously, performing in-place major version upgrades led to unpredictable downtime and the risk of breaking our Service Level Agreements (SLAs) to customers. Using Amazon RDS Blue/Green Deployments for Aurora, we were able to automate and simplify the upgrade process. With Blue/Green Deployments we were able to upgrade from MySQL version 5.6 to version 5.7 on Aurora in about two minutes. This enabled us to preserve our customer SLAs while saving developer time."

Soon Hin Khor, Chief Technical Officer - Packagecloud

Pagely

"We are excited by the flexibility that Amazon Aurora serverless offers in our current managed WordPress hosting stack, and the new serverless WordPress application hosting model that it enables us to pursue. Many of our large-scale WordPress sites have unpredictable database usage. Amazon Aurora serverless gives us the ability to lower customer costs for these fluctuating workloads and free internal resources by eliminating the need for manual or monitored database

resizing. Ultimately this is about saving time, money, and serving our customers better, and Amazon Aurora serverless will help us do that."

Joshua Strebel, Founder & CEO - Pagely

PayNearMe

"At PayNearMe, we power over $50B in annual payments as a leading fintech. We identified a critical bottleneck during proactive load testing due to write concurrency limitations in our primary database cluster. As our business scaled, this constraint risked limiting our ability to serve customers during high-volume periods. With the use of the Amazon Aurora I/O-Optimized configuration for our Aurora MySQL database, we resolved the write concurrency bottleneck, immediately doubling our transaction capacity — a 100% increase in throughput — while providing ample room for future growth. Beyond throughput gains, we achieved substantial performance improvements across our entire system: Average Active Sessions (AAS) and commit latency decreased by approximately 87%, and one core subsystem now runs about 50% faster, no longer constrained by write blocking. These results directly enhance our customer experience and operational efficiency. Working closely with AWS, we adopted Aurora I/O Optimized for our I/O-intensive application to address the specific write concurrency challenges we were encountering without having to rearchitect our application. With a single API call we changed the storage type of our existing Aurora database clusters to be I/O-Optimized;, the implementation was quick and straightforward, delivering immediate and measurable performance gains."

Derek Douglas-Hecker, Performance Architect - PayNearMe

Pearson

"The cloud plays a major role in Pearson's transformation into a digital education company. Using AWS, the IT team created a platform product teams use to create next-generation apps for education. AWS enables our teams to release features and update them globally at any time, lower costs to internal developer partners, transforming work with security teams, and delivering value to students. Amazon Aurora has been a core part of our AWS infrastructure to ensure we have best performance, high availability, and scalability with our increasing users."

Ian Wright, VP of Infrastructure & Operations - Pearson

Personal Capital

"Adopting Amazon Aurora is a game-changer because we offload management, security, and backup to AWS. When we managed our own databases, it could take days or even weeks to spin

up a new instance, replicate data, load test, patch, and so on. With Amazon Aurora, it takes a few clicks or a few lines of script and a new instance is running in minutes. If you multiply those productivity gains across the organization, it's huge."

Fritz Robbins, Chief Technology Officer - Personal Capital

RecoChoku

"If you work with databases on AWS, there is no reason not to choose Amazon Aurora. At peak periods the membership system is accessed via API more than 250 times per second, but there have been no issues at all even under this kind of processing load. Also, because it is a fully managed service, we can focus our resources on service development."

Shuhei Sakai, Systems Architect Group, Business Systems Promotion Department - RecoChoku Co., Ltd.

S&P Dow Jones Indices

S&P Dow Jones Indices (S&P DJI) has nearly 200 databases supporting its core applications, with sizes ranging from 500GB to 15TB.

"We began our migration journey with Amazon Web Services (AWS) migrating from on-premises to MySQL on Amazon EC2 and then directly to Amazon Aurora MySQL-Compatible Edition, utilizing AWS Database Migration Service (AWS DMS). We recently adopted Aurora serverless for a new application that includes machine learning capabilities using AWS SageMaker for sector wide asset classification. Setting up AWS Glue jobs to ingest data from Twitter feeds led to a very dynamic workload for this application. Hence we adopted Aurora serverless, which can scale the compute resources dynamically to support processing all the AWS Glue jobs and AWS Lambda functions, while being able to ingest massive volumes of data. We have also enabled write forwarding to keep our application active in more than one region, with Amazon Aurora Global Database."

Shivakumar Bangalore, Sr. Director of Database Engineering - S&P Global Inc.

Shoplazza

"In 2024, we introduced Amazon Aurora serverless into our microservices architecture, to support critical workloads such as marketing, ordering, and payments. During peak periods Aurora serverless seamlessly scales capacity up and then scales down automatically after the surge.

With this autoscaling configuration, we have cut costs by over 40% while also reducing manual database operations work by 50%. Aurora serverless has empowered our teams to focus less on infrastructure management and more on application innovation that drives our business forward."

Bing Xia, Cofounder & CTO - Shoplazza

ShortMax

"Our business requires real-time resource expansion without downtime to handle sudden traffic spikes. Traditional scaling methods often necessitate minutes of downtime, which is unsuitable for our needs. To address this challenge, we adopted Amazon Aurora serverless database to support our short drama platform, ShortMax, including multiple core modules such as payment and user management. Aurora serverless can scale within seconds, reducing business interruption from 10+ minutes when using traditional scaling methods to zero seconds during the expansion period. It also supports scaling capacity down to zero Aurora Capacity Units (ACUs) with no charges during periods of database inactivity. This solution allows us to efficiently manage our database resources while maintaining maximum business continuity for our 5,500,000 daily active users."

Gao Zhensheng, CTO - ShortMax

Safe Software

"With support for 325+ data formats and limitless transformations, our products' automated daily testing scenarios are demanding: 15,000 x 4 operating systems x 3 products, running 24/7. Add our developers and QA team hitting the servers, and they were tipping over. We migrated our 200 tables and 10 million rows to Aurora with the click of a button. Now we can build better, faster - which means we can deliver a higher quality product. We have no more maintenance headaches, Aurora is scalable without hassles, and we're saving 40% in annual costs."

Iain McCarthy, Product Release Manager - Safe Software

Sophos

"Amazon Aurora has become the default choice for any kind of relational data for us because of its ease of use."

John Peterson, Vice President of Engineering - Sophos

Read the case study »

SugarCRM

"Moving to Amazon Aurora allows us to reclaim 15% of our operations team's time."

Zac Sprackett, SVP Product Management - SugarCRM

Read the case study »

Sumo Logic

"Sumo Logic processes hundreds of petabytes of data every day. Any disruption, delay, or downtime becomes orders of magnitudes worse given that customers use our solution to deal with similar issues of their own. We migrated to Amazon Aurora for a fully-managed, highly-available, durable, MySQL-compatible, VPC-enabled database, and were able to migrate about a hundred database instances with hundreds of databases within them, with zero downtime from our customers' perspective. Our systems are now more robust and future proof, not just in terms of technology but also in terms of database management processes and readiness for future migrations."

Aditya Kelkar, Backend Engineer - Sumo Logic

Sumzap, Inc.

"The migration of Sengoku Enbu -KIZNA- from our on-premises data center to AWS was completed very successfully in February 2019. We are now using several Amazon Aurora clusters (MySQL-compatible edition), with our application servers running on Amazon EC2 instances in the same availability zone for low latency. During gaming events, when we have access peaks three times a day, we use auto-scaling and EC2 Spot for our application servers, and have found that both Aurora and EC2 handle these peaks without any issues whatsoever."

Hiroyuki Ishihara, SRE Team Manager - Sumzap, Inc.

SysAid

"At SysAid, we describe Aurora as the solution to everything! We were looking for a database that could withstand the needs of our established customer base, who need all of their historical IT data at their fingertips. Aurora allowed us to not compromise the level of service that's expected of us. Furthermore, we were looking for a stable and scalable cluster technology and were amazed by how quickly and easily Aurora performed intensive ALTER TABLE operations. We're now free to move SysAid forward without worrying about either performance or stability."

Rafi Rainshtein, Vice President of R&D - SysAid

Threat Stack

"We chose Amazon Aurora because we wanted a data store to support availability cross-region, so in the instance of a downed datacenter our customers can still authenticate."

Jennifer Kim, Lead Software Engineer - Threat Stack

Read the case study »

Trend Micro

"We went from being a bottleneck in our customer's lifecycle to being invisible and even an enhancement in their process due to Aurora."

Steve Quane, Executive Vice President, Network Defense & Hybrid Cloud Security - Trend Micro

Read the case study »

TalentBin

"TalentBin by Monster made the move to Aurora so as to reduce operational over-head and management of MySQL, which in turn allowed our development team to focus on innovation. Aurora offered significantly faster replication, providing larger write operations that wouldn't impact any downstream applications. Plus, Aurora's tools eliminated the need to allocate excessive storage to account for usage and growth demands, which adds even more value and savings. Aurora made it possible for our team to consolidate various databases, reducing our database instance count by roughly 40%. Other gains were earned through automatic snapshots and point-in-time restoration, providing true operational improvements. All of these features made migrating to Aurora an easy decision for us."

Travis Theune, Sr. Site Reliability Engineer - TalentBin

TransNexus

"We tested Aurora's parallel query feature with analytics applications within our ClearIP software product hosted in AWS. We've been excited to find that larger, more intensive queries perform up to 20x faster with Parallel Query turned on."

Alec Fenichel, Software Developer - TransNexus

Tuya Smart

"We selected Amazon Aurora as our core database engine for its unparalleled performance and availability at global scale. Tuya currently manages billions of real-time online devices and can keep cloud message processing response times under 10 milliseconds. However, billions online devices provide a challenge. During holidays, there will be peak traffic volume, with tens of millions of devices going online and offline virtually simultaneously. Tuya used Amazon Aurora to construct a data storage solution to solve the main problem of rapid increase in short-term traffic, and to fully utilize resources. Aurora's design, which separates compute and storage and low-latency replication functionality, improves system throughput by enhancing the effect of read-write separation. Aurora provides up to 15 read replicas, setting the groundwork for Tuya's read flexibility development. At the same time, Tuya has integrated Aurora serverless, which includes seconds-level elastic expansion and contraction, allowing Tuya to handle extremely heavy business traffic smoothly."

Eva Na, Vice President of Marketing & Strategic Cooperation and CMO - Tuya Smart

The United Nations

"At the UN, we operate multiple websites with global reach that require mission-critical reliability and consistent performance. We were able to achieve superb performance even with Amazon Aurora's smallest database engine. Amazon Aurora's new user-friendly monitoring interface made it easy to diagnose and address issues. Its performance, reliability and monitoring really shows Amazon Aurora is an enterprise-grade AWS database."

Mohamad Reza, Information Systems Officer - United Nations

Whale TV

"We use Amazon Aurora MySQL-Compatible Edition for our TV operating system to achieve high performance, availability, and scalability. Using AWS Graviton4-based Amazon EC2 R8g instances for Aurora, our testing revealed an over 40% increase in performance and, most impressively over 20% reduction in business latency over Intel predecessors. The combination of Aurora and Graviton4 instances delivers the best performance for our workloads."

Kenvin Zhou, SRE Director - Whale TV

X.D. Network

Ragnarok Online, an MMORPG (Massively Multiplayer Online Role-Playing Game) mobile game we launched in the Korea region in March 2018, needs to constantly update game statistics and player status. The burden of massive write operations can reach 25,000 queries per second easily, which exceeds the limit of traditional MySQL databases. Thanks to the Amazon Aurora database, we can accomplish this without changing any of the code, and even keep latency to less than 35 ms to ensure we can continue to expand our user base. Thanks to the high performance and availability that Amazon Aurora provides, Ragnarok Online reached number one in the free and top-grossing mobile game categories in Korea when it launched.

ZipRecruiter

"We migrated our MySQL databases to Amazon Aurora in order to handle our incredibly fast growth. Scalability and high availability can be challenging with large MySQL databases. Amazon Aurora allows us to easily scale our reads through near-zero-lag read replicas, even under a very heavy write load. That combined with reliable, out-of-the-box cross-availability zone failover means we favor Aurora for all of our live data serving needs."

Craig Ogg, Chief Technology Officer - ZipRecruiter

Zixun

"We have been using AWS since 2017 and chose Amazon Aurora, a fully MySQL-compatible managed database service, to support our core business—the Ziniao browser. AWS Graviton processors and Aurora Global Database were key factors in our decision to choose Aurora. We adopted Graviton2 (R6g) instances, achieving a 12% cost advantage compared to Intel x86 instances. As our business expanded to new regions, costs increased, and we needed to further optimize our architecture to improve cost efficiency. The launch of Aurora Graviton3 (R7g) instances provided us with a solution. After switching to R7g instances, we achieved significant improvements: nearly 50% better price-performance ratio, 25% overall performance improvement, 30% reduction in CPU utilization, and 27% latency reduction. These optimizations enabled us to achieve optimal balance between cost and performance while meeting growing business demands. Additionally, by deploying Aurora Global Database, we successfully built a cross-region disaster recovery system and effectively supported regional business expansion needs. Even when facing Region-level failures that are more severe than Availability Zone failures, we can switch our core database operations to other regions in less than 1 minute to provide continuous service, ensuring that our end-user experience is virtually unaffected.."

Wensheng Wang, CTO - Fujian Zixun

Zumba

"Our existing MySQL databases perform millions of transactions per day and we expect them to continue to grow. Amazon Aurora will give us better performance and scalability than MySQL, as well as lower latency read replicas, and we see an opportunity to use Amazon Aurora to improve the latency of our website while also reducing the number of instances required to run it. Best of all, Amazon Aurora's MySQL-compatibility means that we can use it without making changes to our existing applications."

Douglas Jarquin, Director of DevOps - Zumba

Aurora DSQL

Shenzhen Hexinzhilian Technology Development Co., Ltd.

"Scalability and high availability are essential to support our rapid user growth and ensure a high-quality user experience. Aurora DSQL has been a game-changer for our schedule management application. Its serverless architecture and PostgreSQL compatibility meant we didn't have to worry about provisioning or maintenance windows and could use the same PostgreSQL that we're familiar with, delivering a 60% improvement in development efficiency – without needing a dedicated DBA team. Aurora DSQL automatically scales to handle our millions of users with consistent millisecond response times, even under high concurrency. With 99.99% single-Region availability and zero infrastructure management, we can focus entirely on enhancing our product and user experience."

Li Zehuan, CEO of Shenzhen Hexinzhilian Technology Development Co., Ltd