

# Service Name

Amazon MQ

## Service Overview

Amazon MQ is a managed message broker service for Apache ActiveMQ Classic and RabbitMQ that enables software applications to communicate using various programming languages, operating systems, and messaging protocols. The service eliminates the operational burden of managing message broker infrastructure by automating setup, maintenance, security patches, and version upgrades. Amazon MQ supports industry-standard APIs and protocols including JMS, NMS, AMQP, STOMP, MQTT, and WebSocket, allowing applications to migrate to the cloud without rewriting messaging code. Organizations can access ActiveMQ and RabbitMQ web consoles directly while benefiting from AWS-managed infrastructure, automated failover, and integration with other AWS services.

## Key Use Cases

- **Enterprise application modernization:** Migrate existing on-premises message brokers to the cloud without rewriting messaging code
- **Microservices decoupling:** Enable loose coupling between application components using reliable message queuing and pub/sub patterns
- **Event-driven architectures:** Process real-time events and data streams through message routing and transformation workflows

## Features

- **Managed Infrastructure:** Automated provisioning, patching, maintenance, and scaling of message broker infrastructure
- **Multi-Engine Support:** Support for Apache ActiveMQ Classic and RabbitMQ with native web consoles and APIs
- **High Availability:** Active/standby deployments for ActiveMQ and cluster deployments for RabbitMQ across multiple AZs
- **Industry Standard Protocols:** Support for JMS, NMS, AMQP 0.9.1, AMQP 1.0, STOMP, MQTT, and WebSocket protocols
- **Security and Encryption:** VPC access, TLS/SSL encryption, message encryption at rest and in transit with AWS KMS
- **Storage Options:** Durability-optimized with Amazon EFS and throughput-optimized with Amazon EBS storage
- **Network of Brokers:** ActiveMQ broker mesh topology for horizontal scaling and message routing

- **Monitoring Integration:** CloudWatch metrics, logs, and alarms for broker performance monitoring

## Value Proposition

Amazon MQ eliminates the undifferentiated heavy lifting of managing message broker infrastructure while providing full compatibility with existing messaging applications. Customers benefit from automated maintenance, security patching, and version upgrades without operational overhead:

- The service offers high availability through multi-AZ deployments, integrated security with AWS services, and cost optimization through managed storage options
- Unlike self-managed solutions, Amazon MQ includes inter-node data transfer at no additional cost for RabbitMQ and provides seamless integration with AWS services like Lambda, CloudWatch, and VPC
- Organizations can migrate existing ActiveMQ or RabbitMQ applications without code changes while gaining enterprise-grade reliability and AWS security compliance

## Service Integration

Amazon MQ integrates natively with AWS Lambda for event-driven processing, allowing functions to consume messages from brokers automatically. CloudWatch provides comprehensive monitoring through metrics, logs, and alarms for broker performance tracking:

- Amazon VPC enables secure private network access with security groups and subnet configurations
- AWS Key Management Service (KMS) provides encryption key management for message encryption at rest and in transit
- Amazon EFS and EBS provide managed storage options for message persistence
- EventBridge Pipes can consume messages from Amazon MQ brokers for event routing
- AWS PrivateLink enables private API connectivity without internet exposure
- IAM provides fine-grained access control and authentication policies

## Onboarding and Offboarding

Customers begin by accessing the Amazon MQ console and selecting either Apache ActiveMQ or RabbitMQ as the broker engine. The setup process involves choosing deployment mode (single-instance, active/standby for ActiveMQ, or cluster for RabbitMQ), configuring broker instance types, setting up storage options, and creating broker credentials:

- Network configuration includes VPC settings, security groups, and subnet assignments
- Applications connect by updating endpoint URLs to point to the Amazon MQ broker endpoints
- The service provides wire-level protocol endpoints for different messaging protocols
- Most existing applications require only endpoint configuration changes without code modifications
- Getting started tutorials and migration guides are available to assist with the transition process

Getting Started Guide: <https://docs.aws.amazon.com/amazon-mq/latest/developer-guide/getting-started-activemq.html>

## Data Removal

Customers can delete Amazon MQ brokers through the AWS Management Console, API, or CLI by selecting the broker and confirming deletion. The deletion process takes approximately 5 minutes and permanently removes the broker and associated data. For cross-region data replication (CRDR) brokers, customers must first unpair and reboot both primary and replica brokers before deletion. All message data, configurations, and broker metadata are permanently removed during deletion. Customers should ensure proper data backup or migration before initiating deletion as the process is irreversible.

## Backup/Restore and Disaster Recovery

Amazon MQ provides data durability through storage replication across multiple Availability Zones using Amazon EFS for durability-optimized storage and Amazon EBS snapshots for throughput-optimized storage. Active/standby brokers for ActiveMQ and cluster deployments for RabbitMQ provide high availability and automatic failover capabilities. Cross-region data replication (CRDR) enables data replication between brokers in different AWS regions for disaster recovery scenarios. Message persistence is maintained through the underlying storage systems with built-in redundancy.

## Backup Capabilities

Amazon MQ does not directly integrate with AWS Backup service. Instead, the service provides data durability through managed storage solutions including Amazon EFS replication across multiple AZs for durability-optimized brokers and Amazon EBS snapshots for throughput-optimized brokers:

- Message persistence and broker configurations are maintained through the underlying AWS storage infrastructure
- Customers can implement custom backup strategies for broker configurations and message data using Amazon MQ APIs and storage-level backup mechanisms

## Disaster Recovery

Amazon MQ does not directly integrate with AWS Elastic Disaster Recovery service. The service provides disaster recovery capabilities through cross-region data replication (CRDR) for ActiveMQ brokers, enabling real-time data replication between primary and replica brokers in different regions:

- Multi-AZ deployments with automatic failover provide regional disaster recovery
- RabbitMQ cluster deployments distribute nodes across multiple Availability Zones for high availability within a region
- Customers can implement multi-region architectures using network of brokers or application-level replication strategies

## Recovery Objectives

Amazon MQ supports various RPO and RTO objectives through deployment configurations. Active/standby brokers provide near-zero RPO with automatic failover achieving RTO of minutes. Cross-region data replication (CRDR) enables regional disaster recovery with minimal data loss. Single-instance brokers offer basic availability with restart capabilities. Cluster deployments for RabbitMQ provide node-level failover with shared state across multiple nodes. Storage durability through Amazon EFS and EBS provides data persistence and recovery capabilities.

## Service Constraints

Service Quotas: <https://docs.aws.amazon.com/amazon-mq/latest/developer-guide/amazon-mq-limits.html>

FAQ: <https://aws.amazon.com/amazon-mq/faqs/>

## Technical Requirements

Service Documentation: <https://docs.aws.amazon.com/amazon-mq/latest/developer-guide/welcome.html>

User Guide: <https://docs.aws.amazon.com/amazon-mq/latest/developer-guide/welcome.html>

API Reference: <https://docs.aws.amazon.com/amazon-mq/latest/api-reference/resources.html>

## Pricing Overview

Please see the AWS UK G Cloud 15 Pricing Document accompanying this service in the Digital Marketplace.

Public Pricing: <https://aws.amazon.com/amazon-mq/pricing/>

## Cost Optimization

Amazon MQ offers several cost optimization strategies including rightsizing broker instances based on actual message throughput and connection requirements. Customers can choose between durability-optimized storage using Amazon EFS for high availability needs or throughput-optimized storage using Amazon EBS for performance-focused workloads:

- The service eliminates inter-node data transfer charges for RabbitMQ clusters, providing significant cost savings compared to self-managed deployments
- Single-instance brokers offer lower costs for development and testing environments
- Storage costs can be optimized by configuring appropriate message retention policies and monitoring actual storage utilization
- The pay-as-you-use pricing model eliminates upfront infrastructure costs and provides cost predictability

## Savings Considerations

Primary cost savings come from eliminating the operational overhead of managing message broker infrastructure including patching, monitoring, and maintenance activities. The fully managed service reduces staffing requirements for infrastructure management and reduces the time-to-market for messaging applications:

- Inter-node data transfer is included at no additional cost for RabbitMQ, providing significant savings for clustered deployments compared to self-managed solutions
- Storage optimization through managed EFS and EBS reduces storage management complexity and costs
- The service eliminates the need for dedicated infrastructure for broker management and monitoring
- AWS Free Tier provides up to \$200 in credits for new customers, enabling cost-effective evaluation and small-scale deployments