

Service Name

Amazon Elastic MapReduce (EMR)

Service Overview

Amazon EMR is a managed big data platform that simplifies running big data frameworks such as Apache Hadoop, Apache Spark, Apache Hive, Apache Flink, Trino, and Presto on AWS. It automates the provisioning, configuration, and scaling of clusters, allowing organizations to process and analyze vast amounts of data cost-effectively. EMR handles the complex setup of distributed computing environments while providing integration with AWS services like S3, DynamoDB, and Kinesis for comprehensive data processing workflows.

Key Use Cases

- Log processing and clickstream analysis: Processing large volumes of web logs and user behavior data for analytics and insights
- Extract, Transform, Load (ETL) operations: Moving and transforming data between various AWS data stores and preparing data for analytics
- Machine learning and predictive analytics: Running ML algorithms at scale using frameworks like Spark MLlib for model training and inference
- Interactive data analysis and business intelligence: Performing ad-hoc queries and generating reports using SQL engines like Presto and Spark SQL
- Genomics and scientific computing: Processing large-scale genomic datasets and running complex scientific simulations
- Real-time stream processing: Analyzing streaming data from Kinesis and other sources for real-time insights and decision making

Features

- **Managed Scaling:** Automatically scales clusters up and down based on workload demands with configurable policies for optimal performance and cost efficiency
- **Multiple Deployment Options:** Offers EMR on EC2, EMR Serverless, EMR on EKS, and EMR on Outposts to meet different operational requirements
- **EMR Studio:** Web-based integrated development environment for data scientists and engineers with Jupyter notebook support and collaborative features
- **Open-Source Framework Support:** Pre-configured support for Apache Spark, Hadoop, Hive, Flink, Presto, HBase, and many other popular big data frameworks
- **Instance Fleet Management:** Supports mixing On-Demand and Spot instances across multiple instance types and Availability Zones for cost optimization and resilience

- **Security Integration:** Built-in integration with IAM, VPC, encryption at rest and in transit, and AWS Lake Formation for fine-grained access control
- **Data Store Flexibility:** Native integration with S3, HDFS, DynamoDB, and other data stores with optimized connectors for efficient data access

Value Proposition

Amazon EMR provides unmatched ease of use by eliminating the complexity of setting up and managing big data infrastructure. Organizations benefit from significant cost savings through per-second billing, Spot instance integration, and automatic scaling capabilities:

- EMR offers superior performance with AWS-optimized distributions of open-source frameworks and native integration with AWS services
- The platform provides enterprise-grade security, reliability, and compliance features while maintaining the flexibility to use preferred open-source tools
- EMR's serverless option further reduces operational overhead, allowing teams to focus on data insights rather than infrastructure management

Service Integration

Amazon EMR deeply integrates with the broader AWS ecosystem for comprehensive data workflows. S3 serves as the primary data lake with optimized connectors (EMRFS) for high-performance data access:

- Integration with AWS Glue provides centralized metadata management and ETL capabilities
- Lake Formation enables fine-grained access control and data governance
- Kinesis integration supports real-time data processing scenarios
- EMR connects with DynamoDB for NoSQL data processing, CloudWatch for monitoring and alerting, IAM for security, and VPC for network isolation
- Additional integrations include AWS Data Pipeline for workflow orchestration, Step Functions for complex workflows, and SageMaker for machine learning model deployment

Onboarding and Offboarding

Getting started with Amazon EMR requires completing AWS account setup prerequisites including IAM user creation and EC2 key pair generation. Customers can launch their first cluster using the EMR console's quick launch option or follow the comprehensive getting started tutorial:

- The process involves creating an S3 bucket for data storage, uploading sample data and scripts, then launching a cluster with desired applications like Spark or Hadoop
- EMR provides templates and guided workflows for common use cases

- Customers can choose between different deployment options (EC2, Serverless, EKS) based on their operational preferences
- EMR Studio offers an integrated development environment for interactive development and testing

Getting Started Guide: <https://docs.aws.amazon.com/emr/latest/ManagementGuide/emr-gs.html>

Data Removal

When offboarding from Amazon EMR, customers must terminate all active clusters through the EMR console or CLI to stop compute charges. Data stored in HDFS is automatically deleted when clusters terminate, while data in S3 persists and must be explicitly deleted. Customers should delete associated S3 buckets, remove CloudWatch log groups, delete security configurations, and clean up any EMR-created IAM roles or policies. For EMR Serverless, applications must be stopped and deleted. Complete data removal requires deleting all associated AWS resources across all regions where EMR was used.

Backup/Restore and Disaster Recovery

Amazon EMR provides backup and disaster recovery through integration with AWS services rather than built-in capabilities. EMRFS enables data storage in S3 with cross-region replication for disaster recovery. EBS snapshots can backup persistent data on core nodes. EMR clusters can be automatically recreated using infrastructure as code templates. For disaster recovery, clusters can be launched in alternate regions with data accessed from replicated S3 buckets. EMR's stateless compute architecture allows for rapid cluster recreation and recovery.

Backup Capabilities

Amazon EMR does not include native backup capabilities but integrates with AWS Backup for EBS volumes attached to cluster nodes. Primary data protection relies on storing data in Amazon S3, which provides 99.999999999% durability and supports versioning and cross-region replication:

- For applications requiring persistent storage, customers can use EBS snapshots or configure automated backup policies
- EMR's integration with S3 as the primary data store provides inherent data protection and backup capabilities through S3's durability and replication features

Disaster Recovery

Amazon EMR supports disaster recovery strategies through AWS infrastructure rather than integration with AWS Elastic Disaster Recovery service. EMR clusters can be quickly recreated in alternate regions using infrastructure templates and configuration scripts:

- Data recovery relies on S3 cross-region replication and multi-AZ storage
- EMR's architecture supports backup and restore, pilot light, and warm standby disaster recovery patterns
- The service's ability to launch clusters rapidly and process data from S3 enables effective disaster recovery implementations with customizable RTO and RPO objectives

Recovery Objectives

Amazon EMR helps customers achieve recovery objectives through rapid cluster provisioning and S3 integration. RPO objectives are met through S3's continuous replication and versioning capabilities, with near-zero data loss potential. RTO objectives can be optimized by pre-staging cluster configurations, using automation scripts for rapid deployment, and maintaining warm standby clusters in alternate regions. EMR's managed scaling and instance fleet capabilities ensure quick capacity acquisition during recovery scenarios, supporting RTO requirements from minutes to hours based on cluster size and configuration complexity.

Service Constraints

Service Quotas: <https://docs.aws.amazon.com/emr/latest/ManagementGuide/emr-service-limits-what-are.html>

FAQ: <https://aws.amazon.com/emr/faqs/>

Technical Requirements

Service Documentation:

<https://docs.aws.amazon.com/emr/latest/ManagementGuide/emr-what-is-emr.html>

User Guide: <https://docs.aws.amazon.com/emr/latest/ManagementGuide/emr-what-is-emr.html>

API Reference: <https://docs.aws.amazon.com/emr/latest/APIReference/Welcome.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/emr/pricing/>

Cost Optimization

Amazon EMR offers several unique cost optimization features including per-second billing with one-minute minimum charges, eliminating waste from hourly billing models. Spot

instance integration can provide up to 90% cost savings with intelligent instance fleet management across multiple pools:

- Managed scaling automatically adjusts cluster size based on workload demands, preventing over-provisioning
- EMR Serverless eliminates idle time charges by automatically starting and stopping resources
- Reserved Instances and Savings Plans provide predictable cost savings for steady workloads
- S3 Intelligent-Tiering integration optimizes storage costs, and AWS Graviton instances offer additional price-performance benefits

Savings Considerations

Primary cost savings opportunities include leveraging Spot instances for fault-tolerant workloads, which can reduce compute costs by up to 90%. Right-sizing clusters through instance type optimization and managed scaling prevents over-provisioning:

- Using S3 as primary storage instead of HDFS reduces costs and eliminates storage redundancy expenses
- EMR Serverless eliminates charges for idle cluster time and management overhead
- Implementing auto-termination policies prevents forgotten clusters from accruing charges
- Reserved Instances and Savings Plans provide significant savings for predictable workloads
- Cost allocation tagging and detailed monitoring help identify optimization opportunities and control spending across different projects and teams