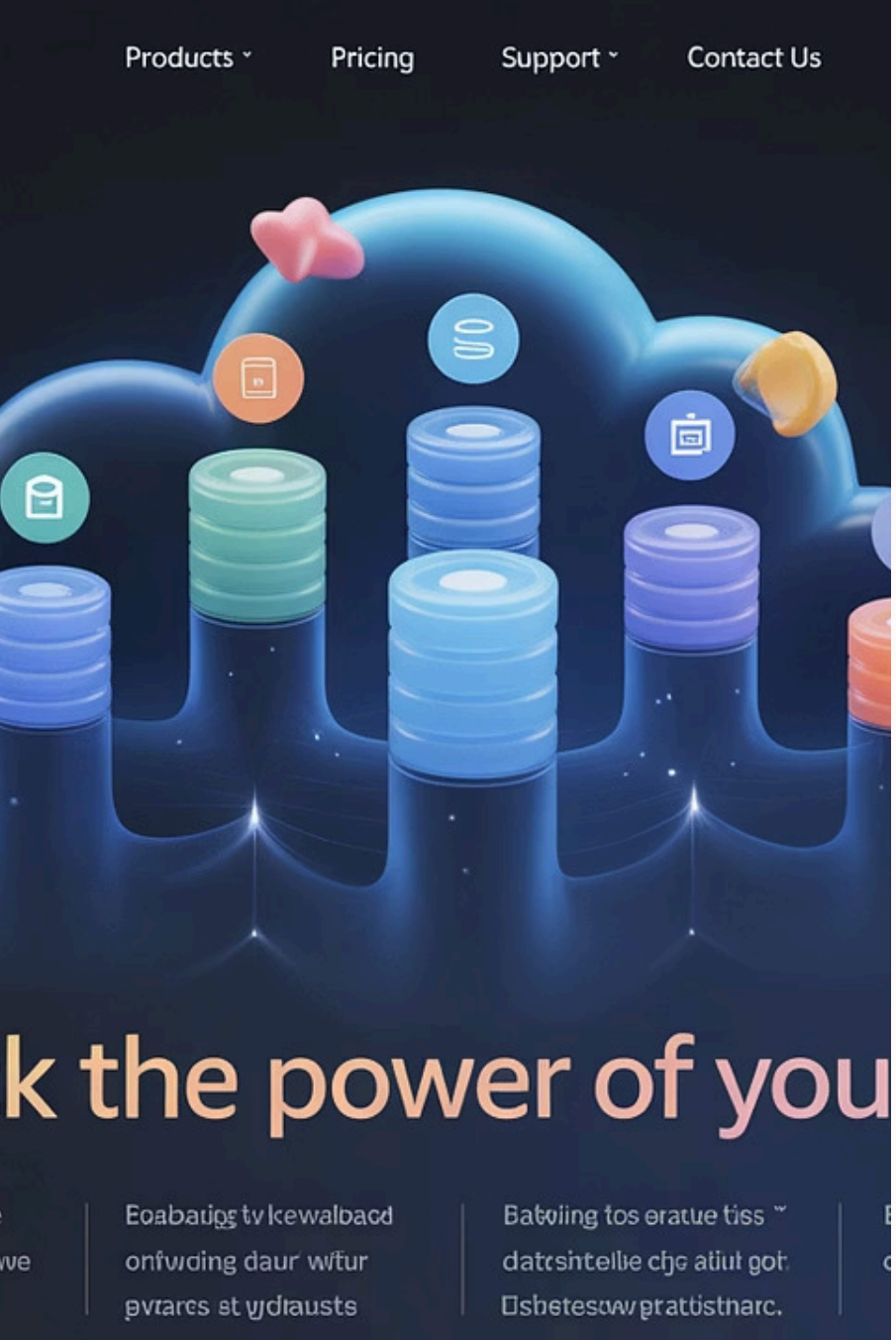



# Identifying AWS Database Services

AWS offers a comprehensive suite of purpose-built database services designed to support a wide range of applications and workloads. From traditional web applications to cutting-edge IoT solutions and advanced analytics, AWS provides specialized database options optimized for specific data models and use cases.

# AWS Database Categories

## Relational Databases

Traditional databases where data is stored in tables with rows and columns. They use structured query language (SQL) and maintain relationships between data elements. Ideal for applications with well-defined schemas and complex queries.

## Non-relational (NoSQL)

Flexible databases that accommodate various data models including key-value, document, time series, graph, and ledger. They offer high performance, seamless scalability, and are built for specific data patterns and workloads.

## Performance-Optimized

AWS database services are engineered for enterprise-grade reliability, automatic scaling to match workload demands, and consistent performance even during peak operations, with minimal management overhead.

# Amazon RDS and Aurora (Relational)

## Amazon RDS

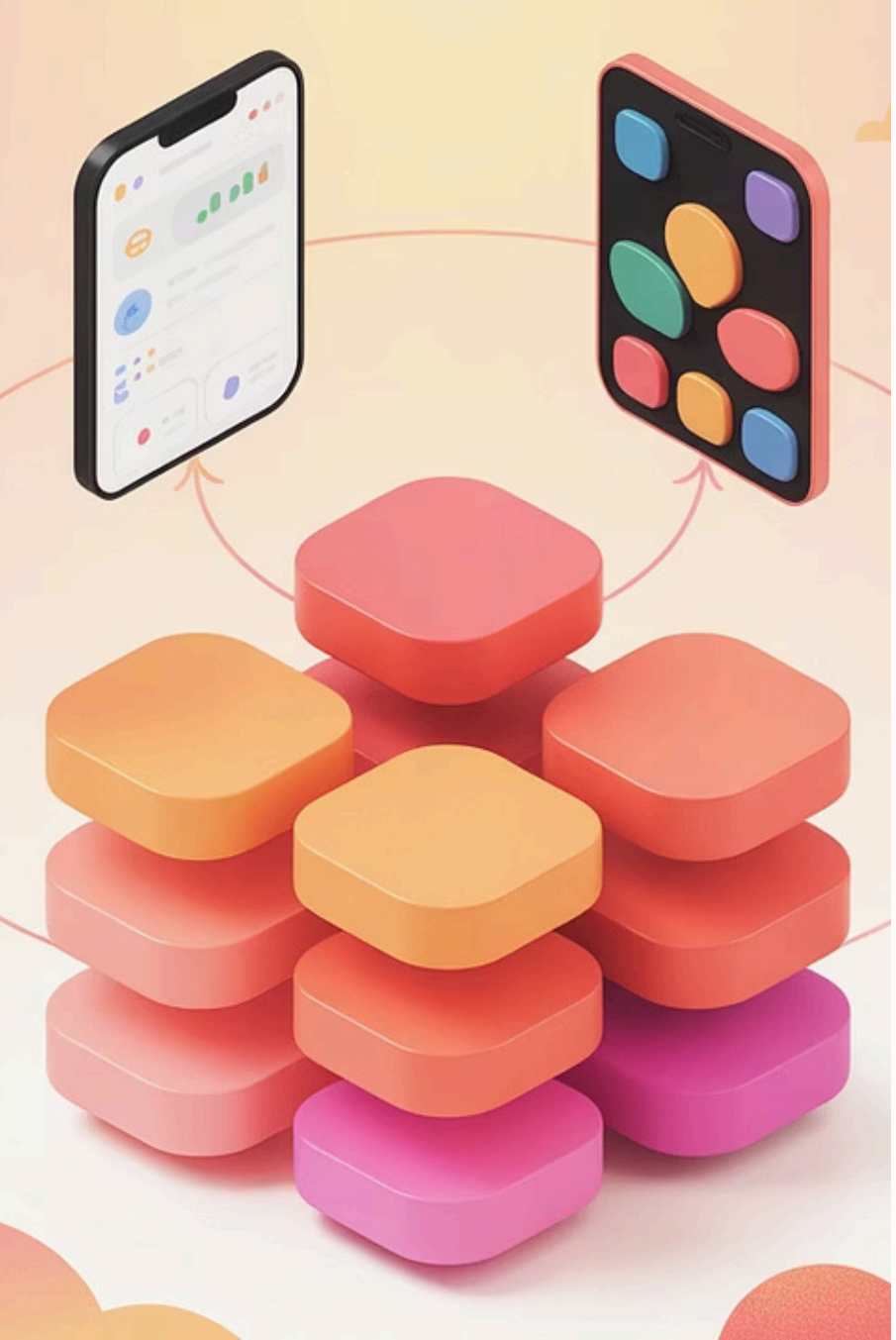
A fully managed relational database service that simplifies setup, operation, and scaling of relational databases in the cloud.

- Supports six popular database engines
- Automated backups, patching, and maintenance
- Multi-AZ deployments for high availability
- Read replicas for improved read performance

## Amazon Aurora


A MySQL and PostgreSQL-compatible relational database built for the cloud with superior performance and availability.

- 3-5x performance of standard MySQL
- Self-healing storage with six-way replication
- Continuous backup to Amazon S3



# Amazon DynamoDB (Key-Value and Document)

| <1ms | 100M+ | 99.99... |
| --- | --- | --- |
| Response Time | Requests Per Second | Availability SLA |
| Consistent single-digit millisecond latency regardless of scale or data volume | Handles extreme workloads with seamless scaling | Enterprise-grade reliability for mission-critical applications |

DynamoDB is a fully managed, serverless NoSQL database service providing seamless scalability for applications of any size. Its flexible data model supports both key-value and document data structures, making it ideal for web applications, mobile backends, gaming platforms, IoT systems, and e-commerce solutions.

# Amazon ElastiCache & MemoryDB (In-Memory)

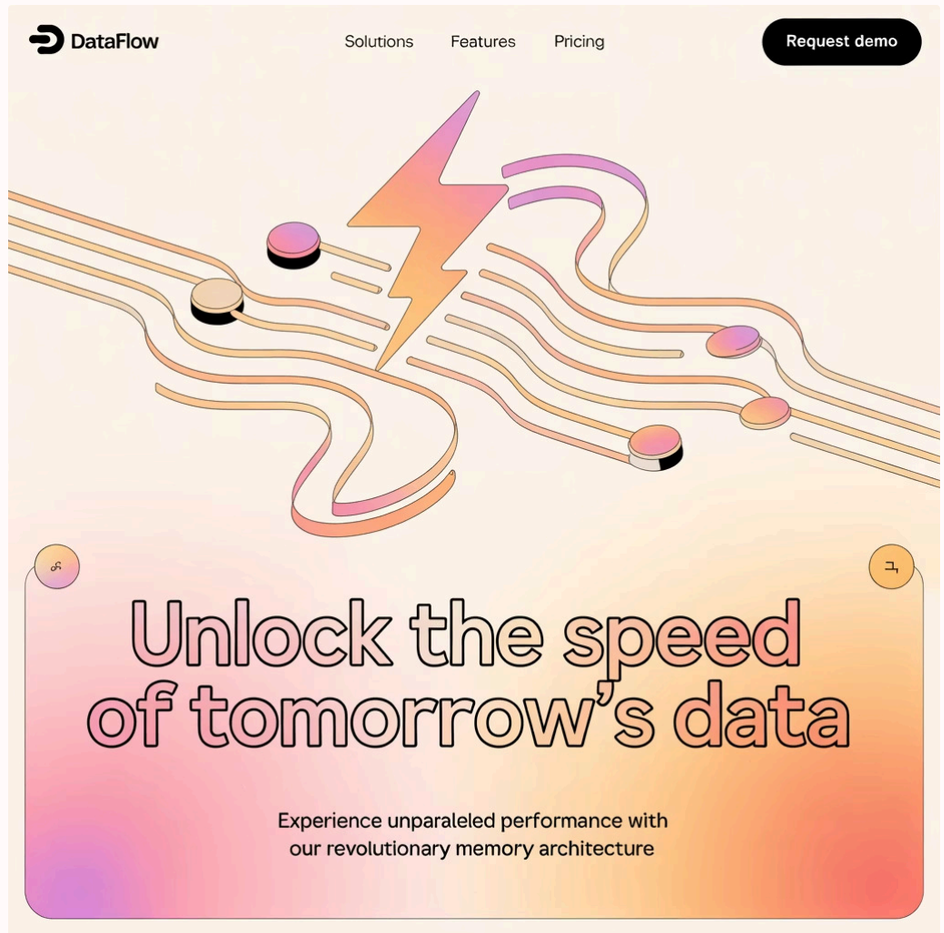
## Amazon ElastiCache

A fully managed in-memory caching service compatible with Redis and Memcached, designed to improve application performance by retrieving data from high-speed in-memory data stores.

### Key Use Cases:

- Web page caching
- Gaming leaderboards
- Real-time analytics
- Session stores

## Amazon MemoryDB




A Redis-compatible, durable in-memory database service that combines ultra-fast performance with Multi-AZ durability.

MemoryDB delivers microsecond read latency, single-digit millisecond write latency, and durability across multiple Availability Zones, making it ideal for high-performance microservices architectures.

# Time Series, Ledger, and Specialized Databases

## Amazon Timestream

Purpose-built time series database for collecting, storing, and processing time-stamped data at scale.

- IoT sensor data analytics
- DevOps monitoring metrics
- Industrial telemetry

## Amazon QLDB

Fully managed ledger database providing a transparent, immutable, and cryptographically verifiable transaction log.

- Financial transaction history
- Supply chain tracking
- Healthcare record systems

## Management Features

All AWS database services share enterprise-grade management capabilities:

- Automated backups and point-in-time recovery
- Dynamic scaling of storage and compute resources
- Comprehensive security controls and encryption
- Seamless integration with AWS monitoring tools
- Global replication options for disaster recovery

# Selecting an AWS Database Service

### Assess Data Model

Evaluate your data structure and access patterns. Relational databases (RDS, Aurora) are ideal for structured data with complex relationships. NoSQL databases like DynamoDB excel with flexible schemas and specific access patterns.

### Consider Scale Requirements

Determine your expected throughput, storage needs, and scaling patterns. Serverless options like DynamoDB provide automatic scaling with no capacity planning, while RDS offers more predictable scaling with provisioned resources.

### Leverage Managed Services

Focus on building applications rather than managing infrastructure. AWS managed database services handle time-consuming tasks like provisioning, patching, backup, recovery, and scaling, allowing your team to focus on innovation.

The AWS database portfolio allows you to mix and match different database types within a single application, choosing the right tool for each specific data workload. This purpose-built approach enables you to maximize performance while minimizing operational overhead and cost.