

Amazon RDS



Amazon Relational Database Service (RDS)

1.  Managed relational database service

2. Simplifies setup, operation, scaling

3. Cloud-based

4. Reduces administrative overhead

5. Supports multiple database engines

6. Automates provisioning, patching, and backups

7. Allows developers to focus on application development

8. Network isolation using Amazon VPC

9. Encryption at rest using AWS KMS

10. Encryption in transit using SSL/TLS

11. Pay only for resources consumed

12. Benefit from AWS Cloud cost-effectiveness and scalability

2.  Handles increased traffic and storage needs

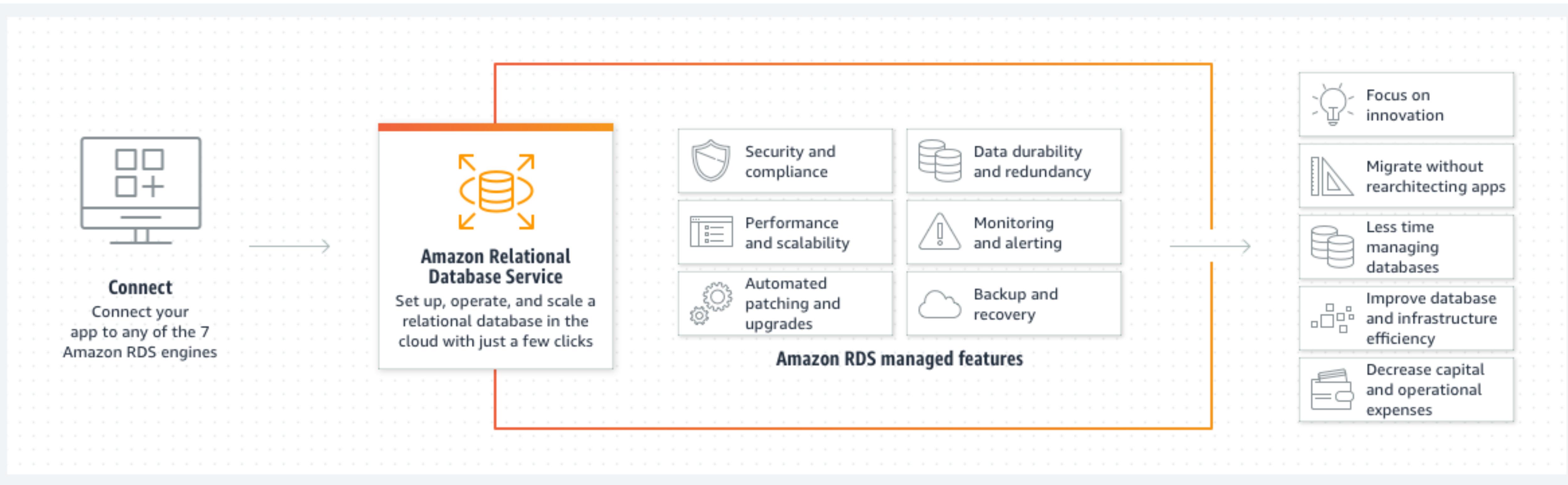
3. Vertical scaling: change instance size

4. Horizontal scaling: add read replicas

5. Handles increased traffic and storage needs

6. Monitor and track metrics

7. CPU utilization, memory usage, I/O operations



1. Db2

4. MySQL

7. Amazon Aurora PostgreSQL Compatible Amazon Aurora MySQL Compatible

2. Microsoft SQL Server

5. PostgreSQL

3. MariaDB

6. Oracle

Comparison of On Prem Database Vs. Database on EC2

Feature	On-premises management	Amazon EC2 management
Application optimization	Customer	Customer
Scaling	Customer	Customer
High availability	Customer	Customer
Database backups	Customer	Customer
Database software patching	Customer	Customer
Database software install	Customer	Customer
Operating system (OS) patching	Customer	Customer
OS installation	Customer	Customer
Server maintenance	Customer	AWS
Hardware lifecycle	Customer	AWS
Power, network, and cooling	Customer	AWS

Comparison of Database on EC2 Vs. RDS

Feature	Amazon EC2 management	Amazon RDS management
Application optimization	Customer	Customer
Scaling	Customer	AWS
High availability	Customer	AWS
Database backups	Customer	AWS
Database software patching	Customer	AWS
Database software install	Customer	AWS
OS patching	Customer	AWS
OS installation	Customer	AWS
Server maintenance	AWS	AWS
Hardware lifecycle	AWS	AWS
Power, network, and cooling	AWS	AWS



Amazon RDS Shared Responsibility Model



1. Amazon RDS: Hosts DB instances and clusters

Hosts software components

Ensures availability and performance

2. Customer: Responsible for query tuning

Adjusts SQL queries for performance

Optimizes database efficiency

3. Query performance factors

Database design

Data size and distribution

Application workload and query patterns

4. Monitoring and tuning: Customer-owned processes

Regular assessment and optimization

Individualized for each RDS database

5. Amazon RDS Performance Insights

Identifies problematic queries

Provides insights for informed optimization decisions



Amazon RDS DB Instances



1. Isolated database environments

Running in the cloud

Multiple databases supported

Secure access

2. Preconfigured parameters and security

Optimized for instance class

Security settings

Preconfigured parameters

3. User-created databases stored

Created using RDS API, AWS CLI, or Management Console

Stored within DB instances

4. Connect using database-specific protocols

Applications connect to DB instance

Protocols: MySQL, PostgreSQL, etc.

5. Capacity defined by instance class

Computation capacity

Memory capacity

Range: small to extra large

6. Run in VPC for enhanced security

Amazon Virtual Private Cloud (VPC)

Enhanced control over virtual networking

7. Monitor using CloudWatch and Performance Insights

Track metrics

Identify performance bottlenecks



Understanding DB Instance Classes

1. Determines computation and memory capacity
2. Selection based on processing power and memory needs
3. Consists of instance class type and size
4. Example: db.r6g.2xlarge (memory-optimized, 2xlarge size)



DB Instance Class Types for Various Use Cases

1. General-purpose instances for balanced workloads
2. Memory-optimized instances for memory-intensive apps
3. Burstable-performance instances with baseline and burst capabilities
4. Optimized Reads instances for read-heavy workloads

⭐ General-Purpose DB Instance Classes 🎯

1. db.m7g – Powered by AWS Graviton3 processors
2. db.m6g – Powered by AWS Graviton2 processors
3. db.m6i – Powered by 3rd Gen Intel Xeon Scalable processors
4. db.m5 – Balanced compute, memory, and network resources
5. db.m4 – More computing capacity than db.m3
6. db.m3 – More computing capacity than db.m1

Memory-Optimized DB Instance Classes .

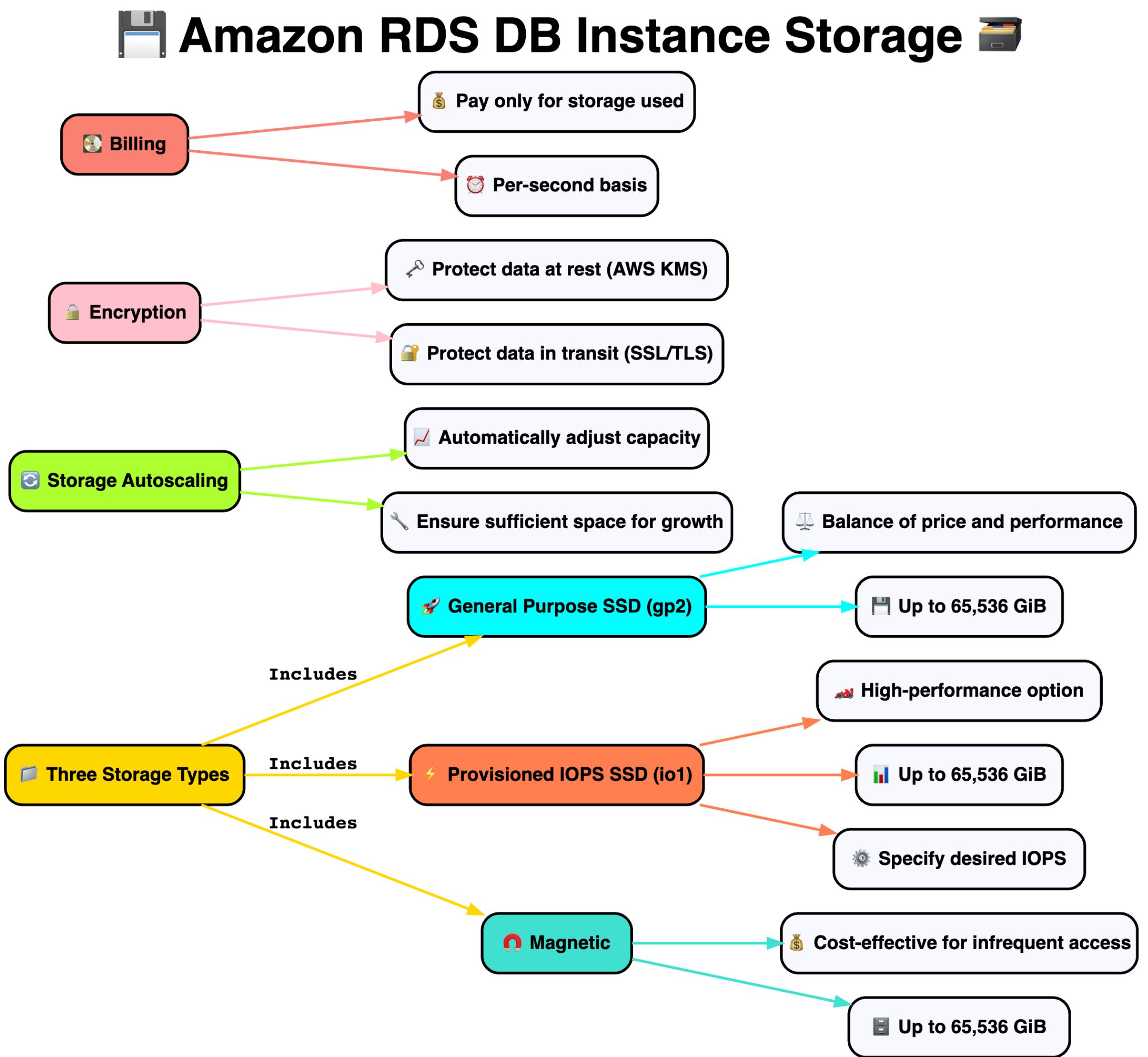
1. db.z1d – High compute and memory, up to 4.0 GHz
2. db.x2g – Powered by AWS Graviton2, low cost per GiB
3. db.x2i – Up to 3.8 TB local NVMe SSD, 100 Gbps network
4. db.x1 – Lowest price per GiB RAM, up to 3,904 GiB
5. db.r7g – Powered by AWS Graviton3, for memory-intensive workloads
6. db.r6g – Powered by AWS Graviton2, for memory-intensive workloads
7. db.r6i – Powered by 3rd Gen Intel Xeon, SAP-Certified
8. db.r5b – Memory-optimized for throughput-intensive apps
9. db.r5d – Low latency, high random I/O, high sequential read
10. db.r5 – Improved networking, powered by AWS Nitro System
11. db.r4 – Improved networking over db.r3
12. db.r3 – Memory optimization (end-of-life for some engines)



Burstable-Performance and Optimized Reads DB Instance Classes



1. db.t4g – Powered by AWS Graviton2, better price-performance
2. db.t3 – Baseline performance, burst to full CPU
3. db.t2 – Baseline performance, burst to full CPU (for dev/test)
4. db.r6gd – Powered by AWS Graviton2, local NVMe SSD storage
5. db.r6id – Powered by 3rd Gen Intel Xeon, SAP-Certified, up to 7.6 TB NVMe SSD



1. **Three storage types:** 🚀 General Purpose SSD (gp2), ⚡ Provisioned IOPS SSD (io1), 🔍 Magnetic
2. **General Purpose SSD (gp2):** ⚖️ Balance of price and performance, 💾 Up to 65,536 GiB
3. **Provisioned IOPS SSD (io1):** 🏁 High-performance option, 📊 Up to 65,536 GiB, 🛡️ Specify desired IOPS
4. **Magnetic:** 💰 Cost-effective for infrequent access, 💾 Up to 65,536 GiB
5. **Storage autoscaling:** 📈 Automatically adjust capacity, 🛠️ Ensure sufficient space for growth
6. **Encryption:** 🔑 Protect data at rest (AWS KMS), 🔒 Protect data in transit (SSL/TLS)
7. **Billing:** 💰 Pay only for storage used, 🕒 Per-second basis

RDS Features - Part 1

1 Blue/Green Deployments 

2 Cross-Region automated backups

3 Cross-Region read replicas

4 Database activity streams

5 Dual-stack mode

6 Export snapshots to S3

7 IAM database authentication

8 Kerberos authentication

9 Multi-AZ DB clusters 

10 Performance Insights

RDS Features - Part 2

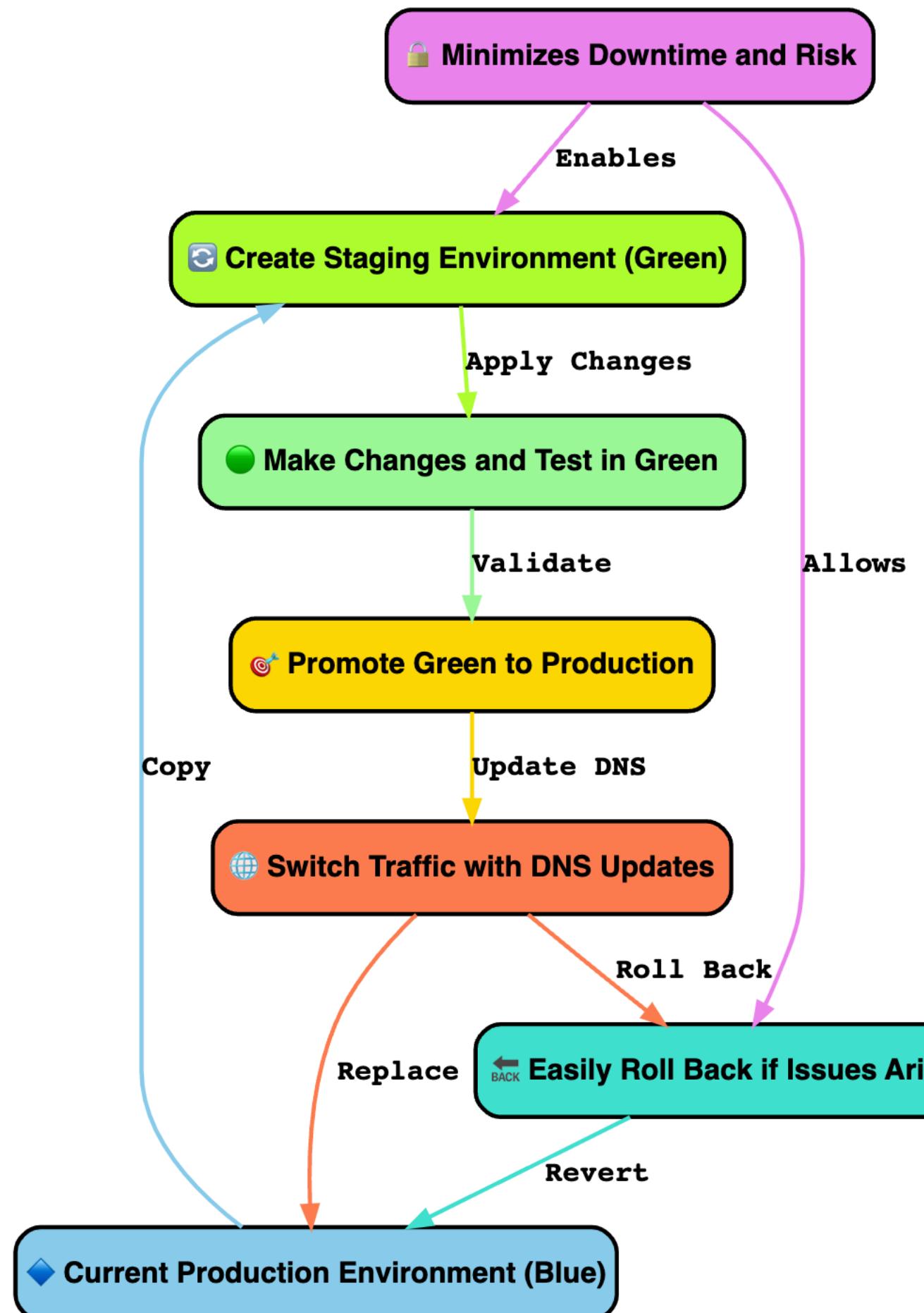
11 RDS Custom

12 Amazon RDS Proxy 

13 Secrets Manager integration 

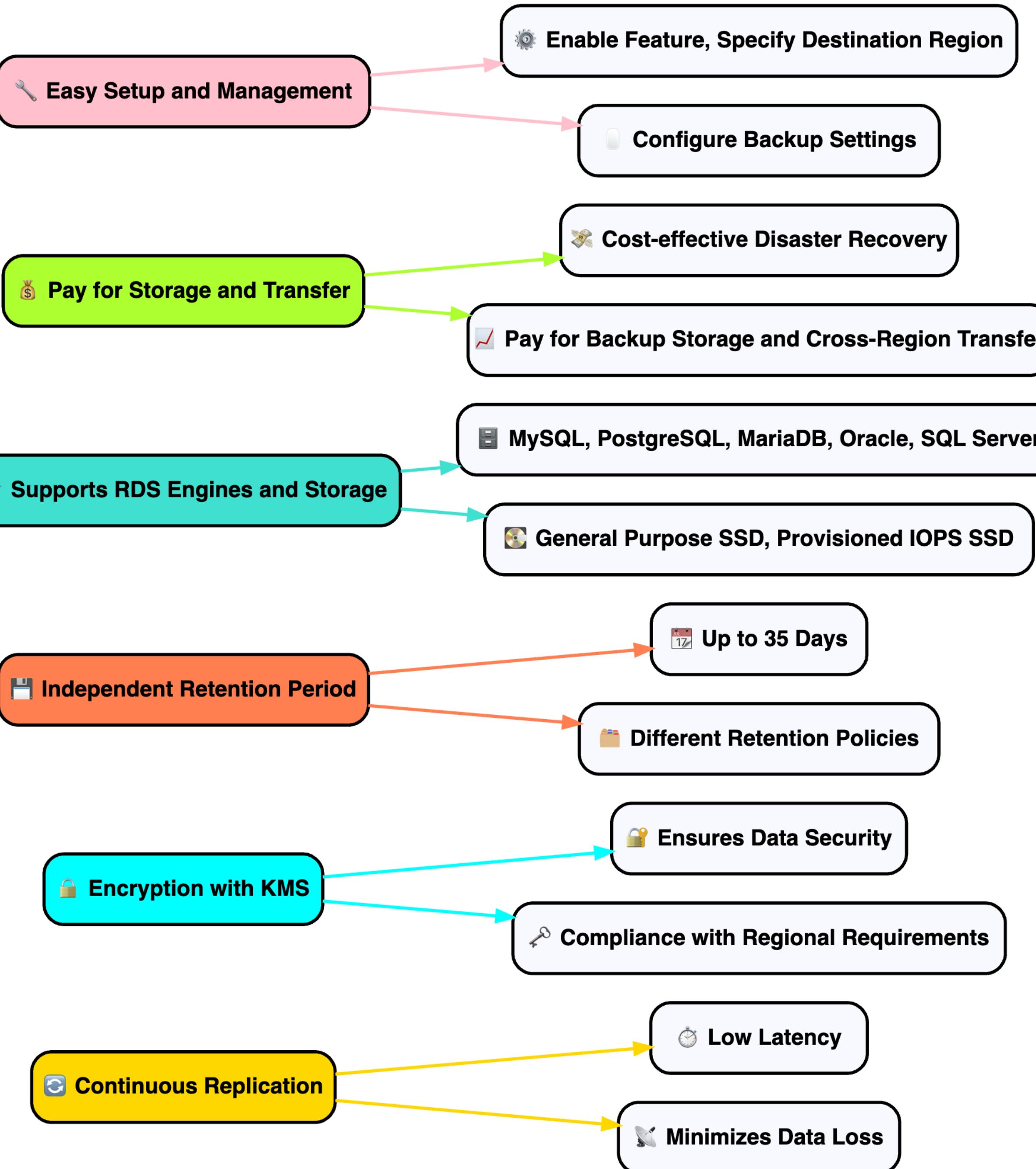
14 Zero-ETL integrations with Amazon Redshift 

Amazon RDS Blue/Green Deployments



1. **Create staging environment (green):** 📁 Copy of current production database, ✨ Known as the "green" environment
2. **Current production environment (blue):** 🌐 Serves live traffic, 👤 While changes are made in green environment
3. **Make changes and test in green environment:** 🔧 Apply database schema changes, upgrades, modifications, 🧪 Thoroughly test changes, ✓ Without impacting production environment
4. **Promote green environment to production:** 💯 Replace blue environment, ✨ After validating changes in green environment
5. **Switch traffic with DNS updates:** 🔄 Update DNS records to point to green environment, 🔗 Ensures smooth transition for applications
6. **Easily roll back if issues arise:** ⏪ Quickly roll back to blue environment, 🔧 By updating DNS records again
7. **Minimizes downtime and risk during updates:** 🛡️ Provides a staged approach, 🕒 Ability to roll back if needed

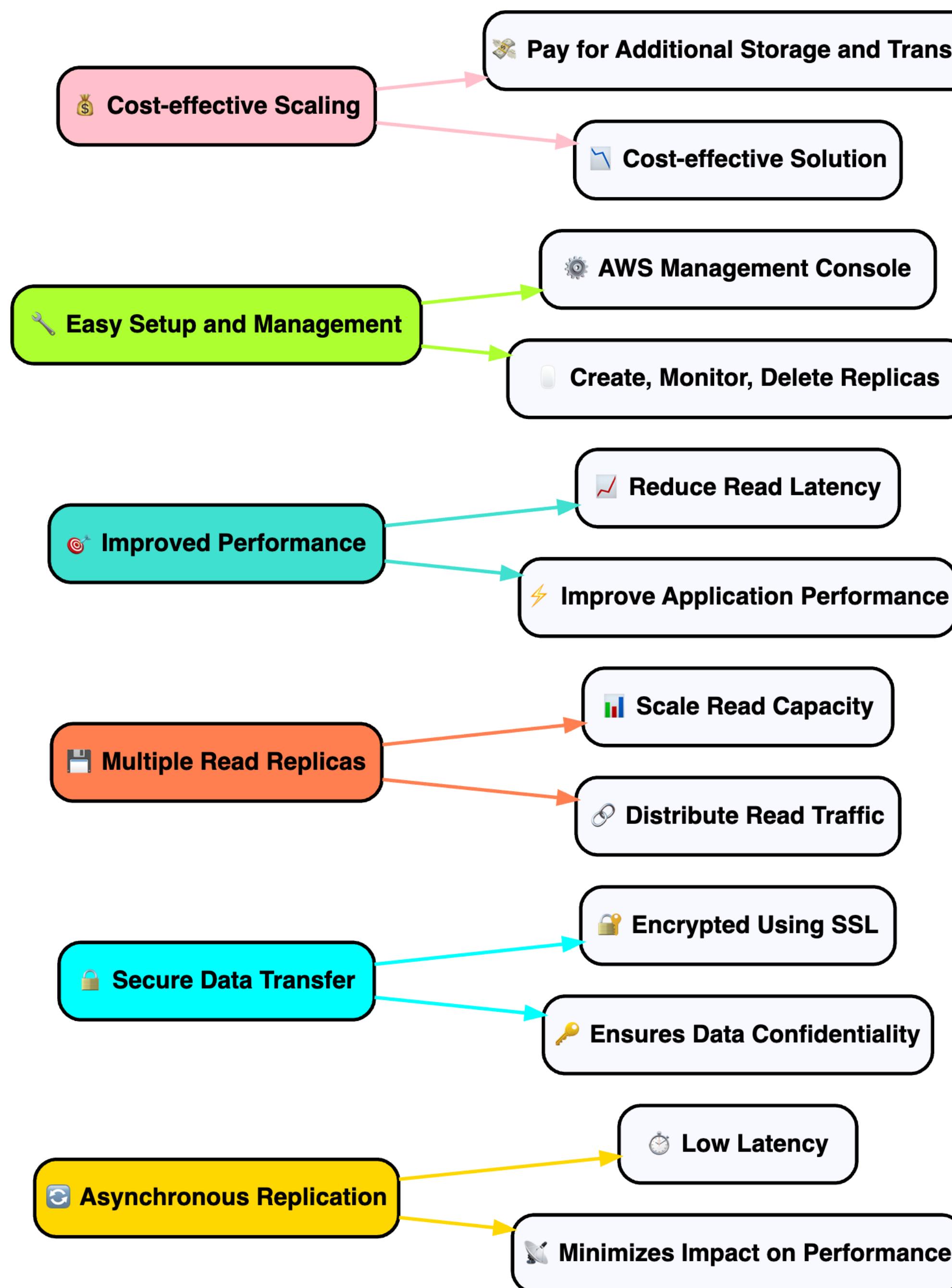
Cross-Region Automated Backups



1. **Automated backup replication:** To another AWS region, Additional protection and disaster recovery
2. **Continuous replication:** Low latency, Minimizes potential data loss
3. **Encryption:** Using KMS key of destination region, Ensures data security and compliance
4. **Independent backup retention period:** Up to 35 days, Allows different retention policies
5. **Supports most RDS engines and storage types:** MySQL, PostgreSQL, MariaDB, Oracle, SQL Server, General Purpose SSD, Provisioned IOPS SSD
6. **Pay only for backup storage and cross-region data transfer:** Cost-effective disaster recovery solution, Charges for storage and transfer fees
7. **Easy setup and management:** Through AWS Management Console, Enable feature, specify destination, configure settings

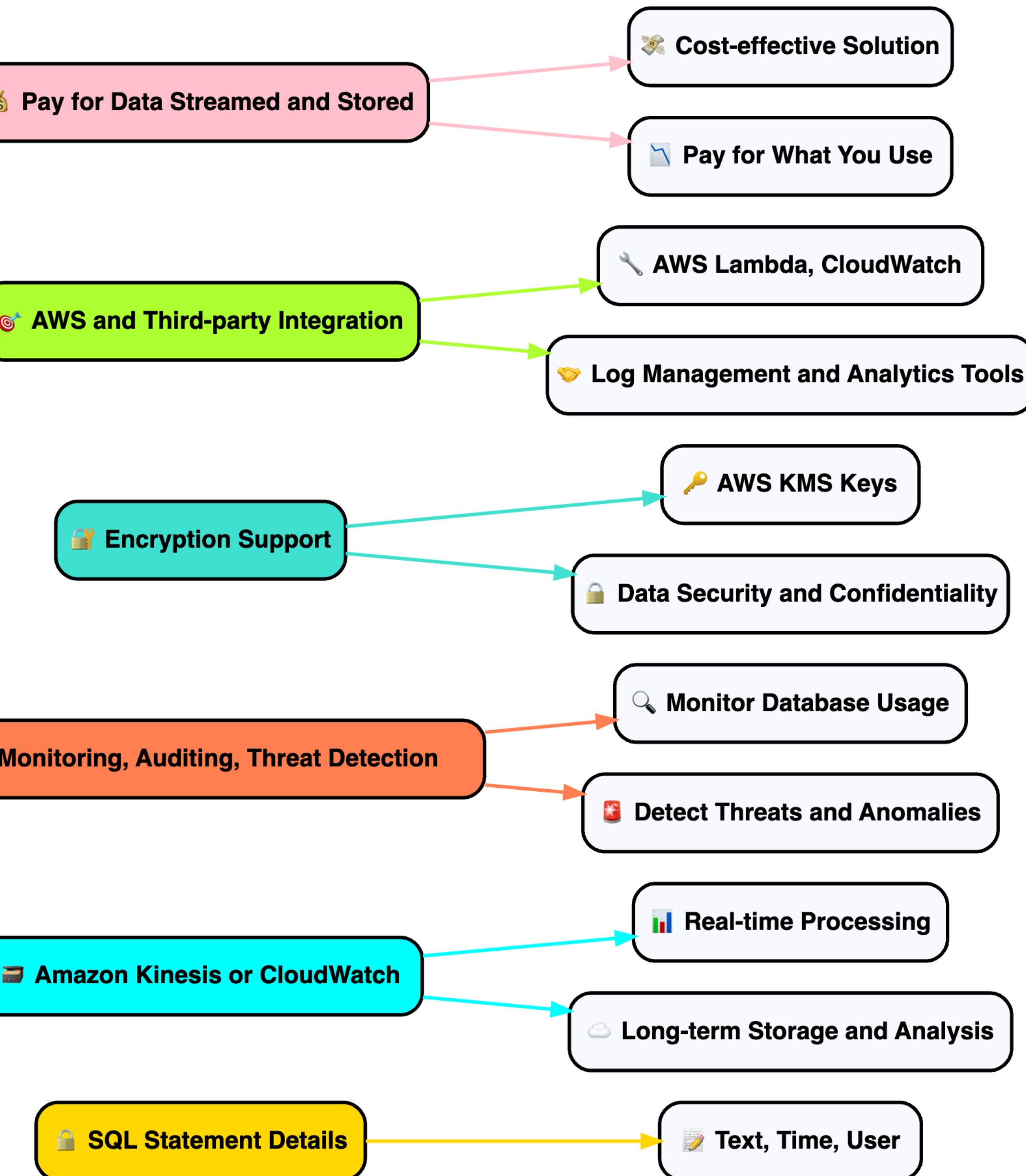


Cross-Region Read Replicas



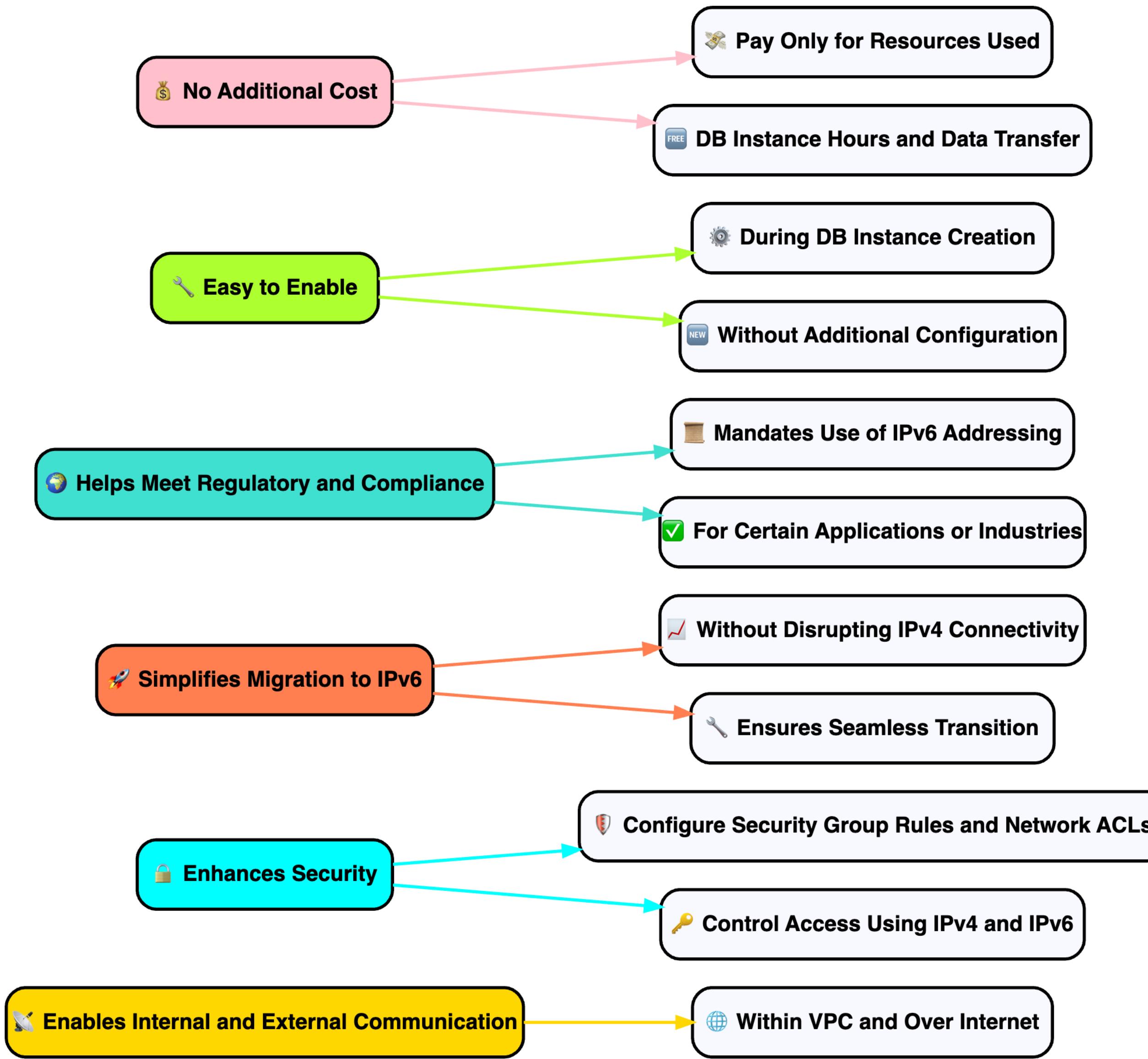
1. **Replicate databases across regions:** Geographical distribution, Improved disaster recovery
2. **Asynchronous replication:** Low latency, Minimal performance impact
3. **Secure data transfer:** Encrypted using SSL, Ensures data confidentiality
4. **Multiple read replicas per primary DB:** Scale read capacity, Distribute read traffic
5. **Improves performance:** Reduces read latency, Enhances application performance
6. **Easy setup and management:** AWS Management Console, Create, monitor, delete replicas
7. **Cost-effective scaling:** Pay for additional storage and transfer, Economical solution

Database Activity Streams



1. **Near real-time stream:** 🕒 Monitor database usage patterns, 🔎 Detect potential security threats
2. **Captures SQL statement details:** 🛡️ Text of SQL statement, 🕒 Time of execution, 💤 User who executed
3. **Stores activity:** 📈 Amazon Kinesis for real-time processing, ☁️ CloudWatch Logs for long-term storage and analysis
4. **Enables monitoring, auditing, threat detection:** 🔎 Monitor database usage, 🔎 Audit access patterns, ⚡ Detect threats and anomalies in real-time
5. **Supports encryption:** 🔑 AWS KMS keys, 🛡️ Ensures data security and confidentiality
6. **Integrates with AWS and third-party tools:** AWS Lambda, CloudWatch, 🤝 Log management and analytics tools
7. **Pay only for data streamed and stored:** 💰 Cost-effective solution, 💸 Pay for what you use

Dual-Stack Mode: IPv4 and IPv6 Support



1. **IPv4, IPv6 addressing:** Supports both Internet Protocol version 4 and version 6.
2. **Internal, external communication:** Allows communication with both VPC and internet resources.
3. **Network access control:** Enhances security with group rules and ACLs.
4. **IPv6 migration:** Enables smooth transition without disruption.
5. **Regulatory compliance:** Meets requirements that mandate IPv6 use.
6. **Easy to enable:** Can be enabled during instance creation with no extra configuration.
7. **No additional cost:** Free to enable, pay only for the resources used.



Export Amazon RDS Snapshots to Amazon S3



1. Export manual or automated DB snapshots to S3

3. Encryption using AWS Key Management Service (KMS)

5. Store snapshots in Apache Parquet format

7. Pay only for S3 storage and data transfer costs

Export to S3

Store snapshot data externally

Use for various purposes

Ensure data security

Protect data at rest

Columnar storage format

Efficient compression and encoding

Improved query performance

Pay for S3 storage consumed

Pay for applicable data transfer costs

No additional charges for exporting

2. Supports one-time or recurring exports

4. Export to S3 bucket in the same or different AWS Region

6. Query exported data using Amazon Athena, Redshift, or EMR

Set up one-time exports

Configure recurring exports

Define export schedule

Same AWS Region

Different AWS Region

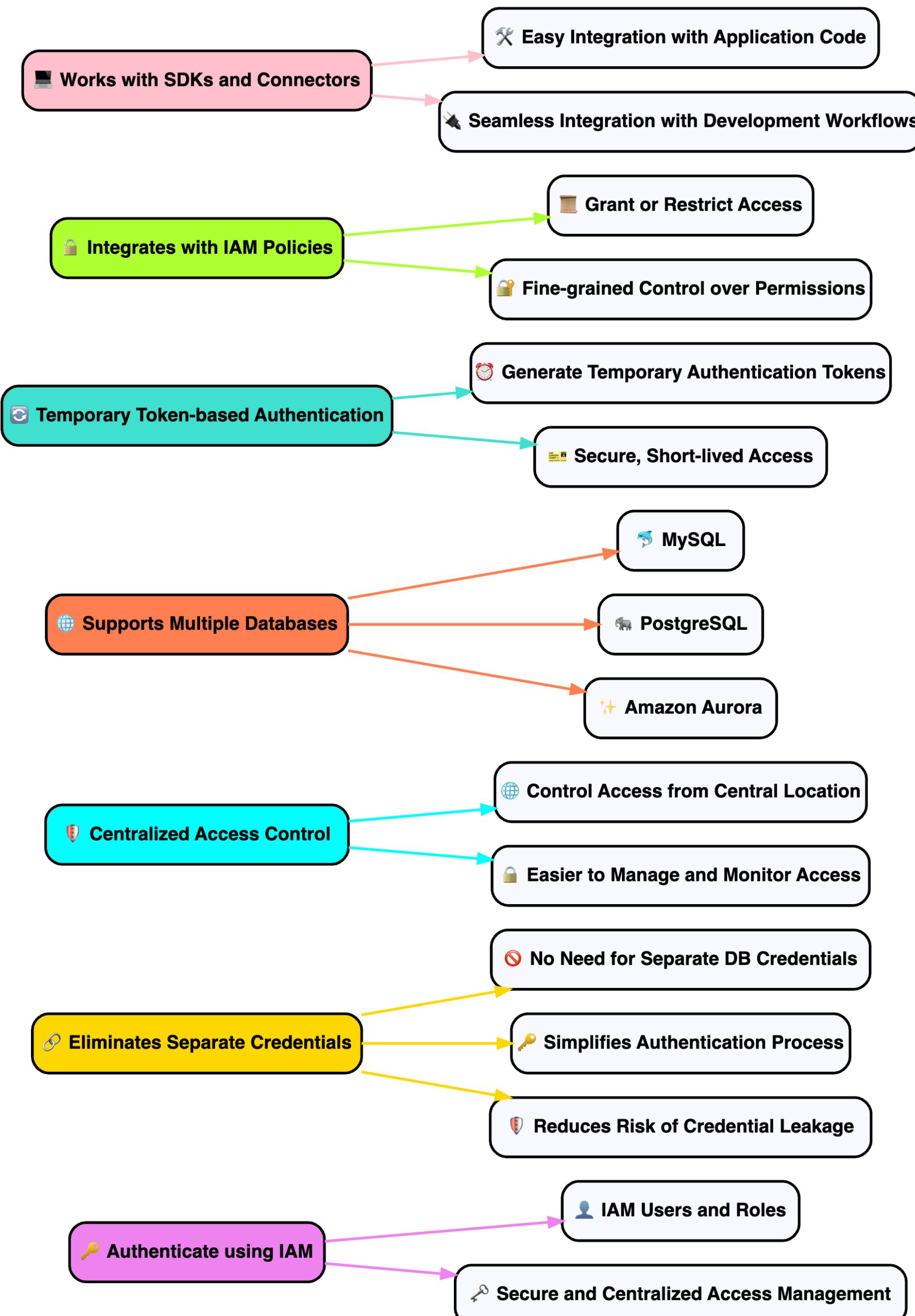
Flexibility for data storage

Disaster recovery scenarios

Powerful analytics

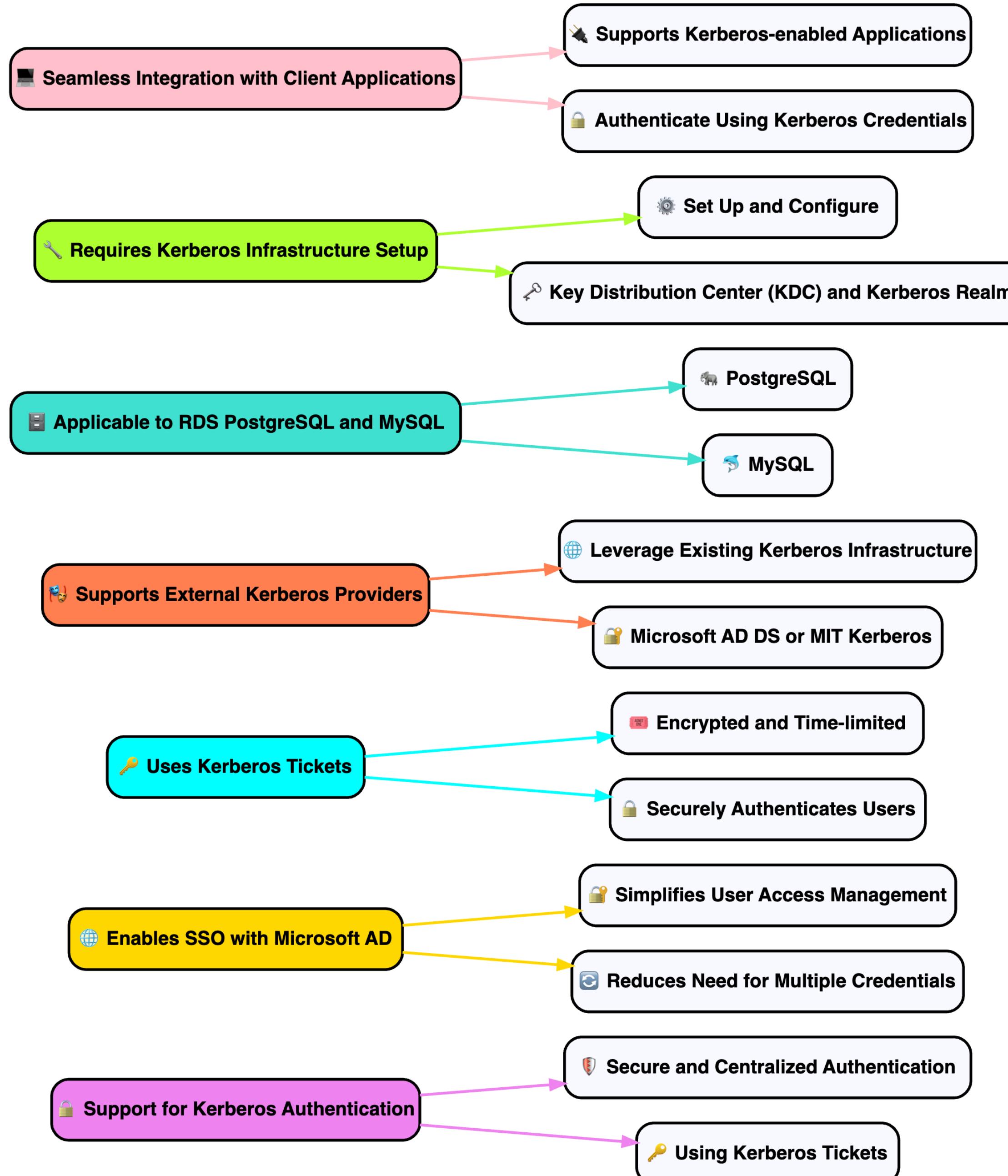
Data processing capabilities

IAM Database Authentication in Amazon RDS



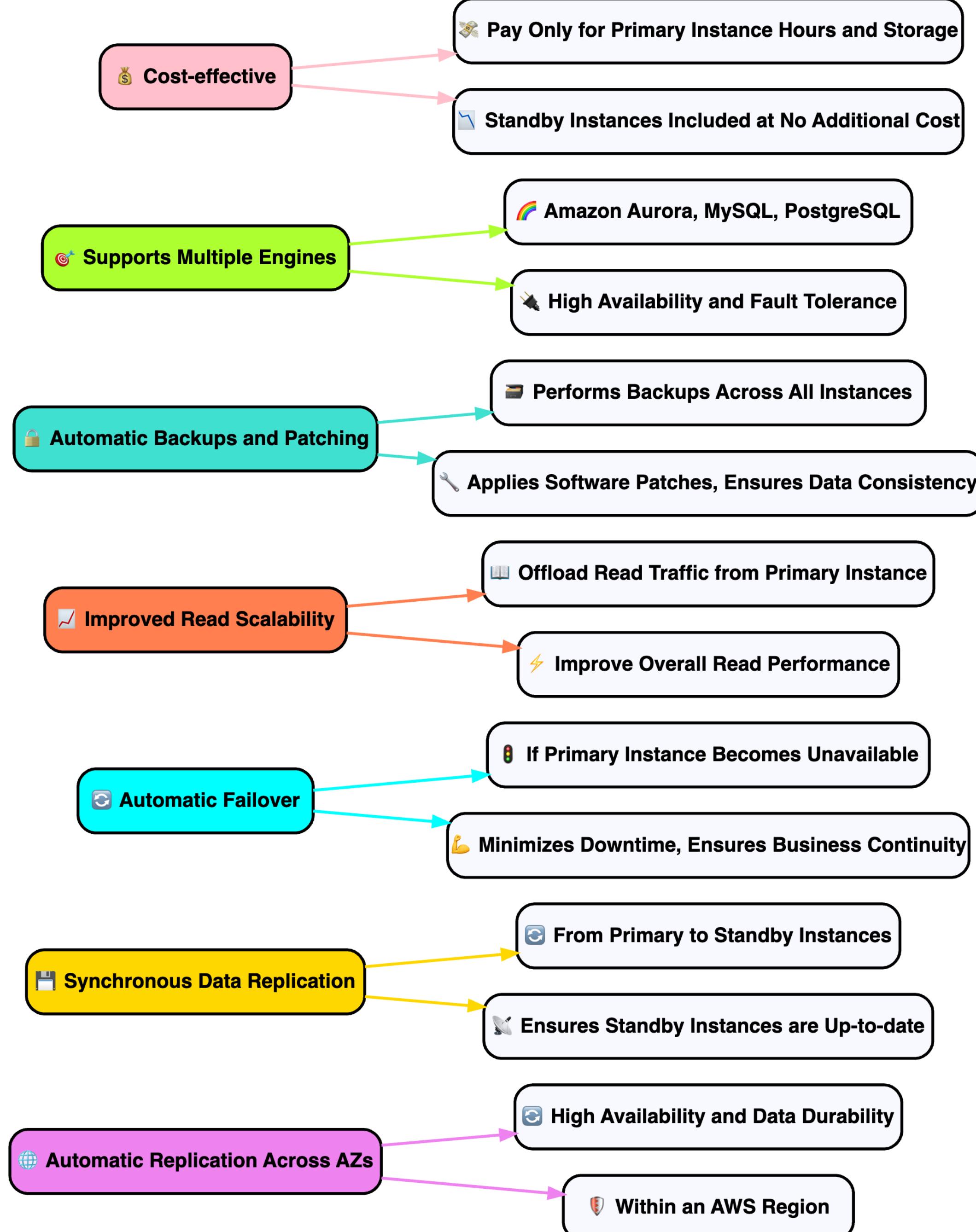
1. **Authenticate using IAM:** IAM users and roles, Secure and centralized access management
2. **Eliminates need for separate database credentials:** No separate database usernames and passwords, Simplifies authentication process, Reduces risk of credential leakage
3. **Centralized control over database access:** Control access from central location, Easier to manage and monitor access
4. **Supports multiple databases:** MySQL, PostgreSQL, Amazon Aurora
5. **Enables temporary, token-based authentication:** Generate temporary authentication tokens, Secure, short-lived access
6. **Integrates with IAM policies and permissions:** Grant or restrict access, Fine-grained control over permissions
7. **Works with AWS SDKs and database connectors:** Easy integration with application code, Seamless integration with development workflows

Kerberos Authentication in Amazon RDS

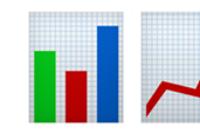


- Support for Kerberos authentication:** Secure and centralized authentication, using Kerberos tickets.
- Enables SSO with Microsoft AD:** Simplifies user access management, reduces need for multiple credentials.
- Uses Kerberos Tickets:** Encrypted and time-limited, securely authenticates users.
- Supports external Kerberos providers:** Leverages existing Kerberos infrastructure, Microsoft AD DS or MIT Kerberos.
- Applicable to RDS PostgreSQL and MySQL:** PostgreSQL, MySQL.
- Requires Kerberos infrastructure setup:** Set up and configure, Key Distribution Center (KDC) and Kerberos realm.
- Seamless integration with client applications:** Supports Kerberos-enabled applications, authenticate using Kerberos credentials.

Multi-AZ DB Clusters in Amazon RDS



1. **Automatic replication across multiple Availability Zones:** Replicates data across AZs, Provides high availability and data durability
2. **Synchronous data replication to standby instances:** From primary to standby instances, Ensures standby instances are up-to-date
3. **Automatic failover to healthy instances:** If primary instance becomes unavailable, Minimizes downtime, ensures business continuity
4. **Improved read scalability with read replicas:** Offload read traffic from primary instance, Improves overall read performance
5. **Automatic backups and patching across instances:** Performs backups across all instances, Applies software patches, ensures data consistency
6. **Supports Aurora, MySQL, and PostgreSQL engines:** Available for popular database engines, Provides high availability and fault tolerance
7. **Pay only for primary instance hours and storage:** No additional cost for standby instances, Cost-effective solution



Performance Insights



1. 🔎 **Real-time performance metrics and visualizations:** 📈 Identifies bottlenecks, 🌐 Optimize performance
2. 🧠 **Automatic detection of performance issues:** 🚨 High CPU, I/O bottlenecks, slow queries, 🛠 Easy to pinpoint and resolve
3. 📊 **Detailed analysis of database load and queries:** 🔎 Identify resource-intensive queries, 📋 Optimize for better performance
4. ⏳ **Historical data retention for up to 2 years:** 📅 Analyze long-term trends, 📈 Compare performance over time
5. 🛠 **Integration with AWS management tools:** 🌃 Monitor, log, analyze, 🌄 Alongside other AWS resources
6. 🔒 **Secure access control with IAM policies:** 💾 Authorized users only, 🛡 View and analyze data
7. 💰 **No additional cost for using Performance Insights:** 💸 Available at no extra cost, 💵 Cost-effective solution



RDS Custom: Tailored Control for Amazon RDS



1. Customizable database environment

Install custom patches, packages, configurations

3. Full control over database and OS

Fine-tune settings

5. Ideal for legacy and packaged applications

Specific database engine versions

7. Pay only for the resources consumed

Compute, storage, I/O resources

Cost-effective for customized environments

Full access to underlying OS

Install custom software, apply optimizations

2. Access to underlying operating system

4. Supports custom patching and configurations

6. Automated monitoring, backups, and recovery

8. Network setting adjustments

Secure shell (SSH) access

Flexibility to customize database environment

Address compatibility requirements

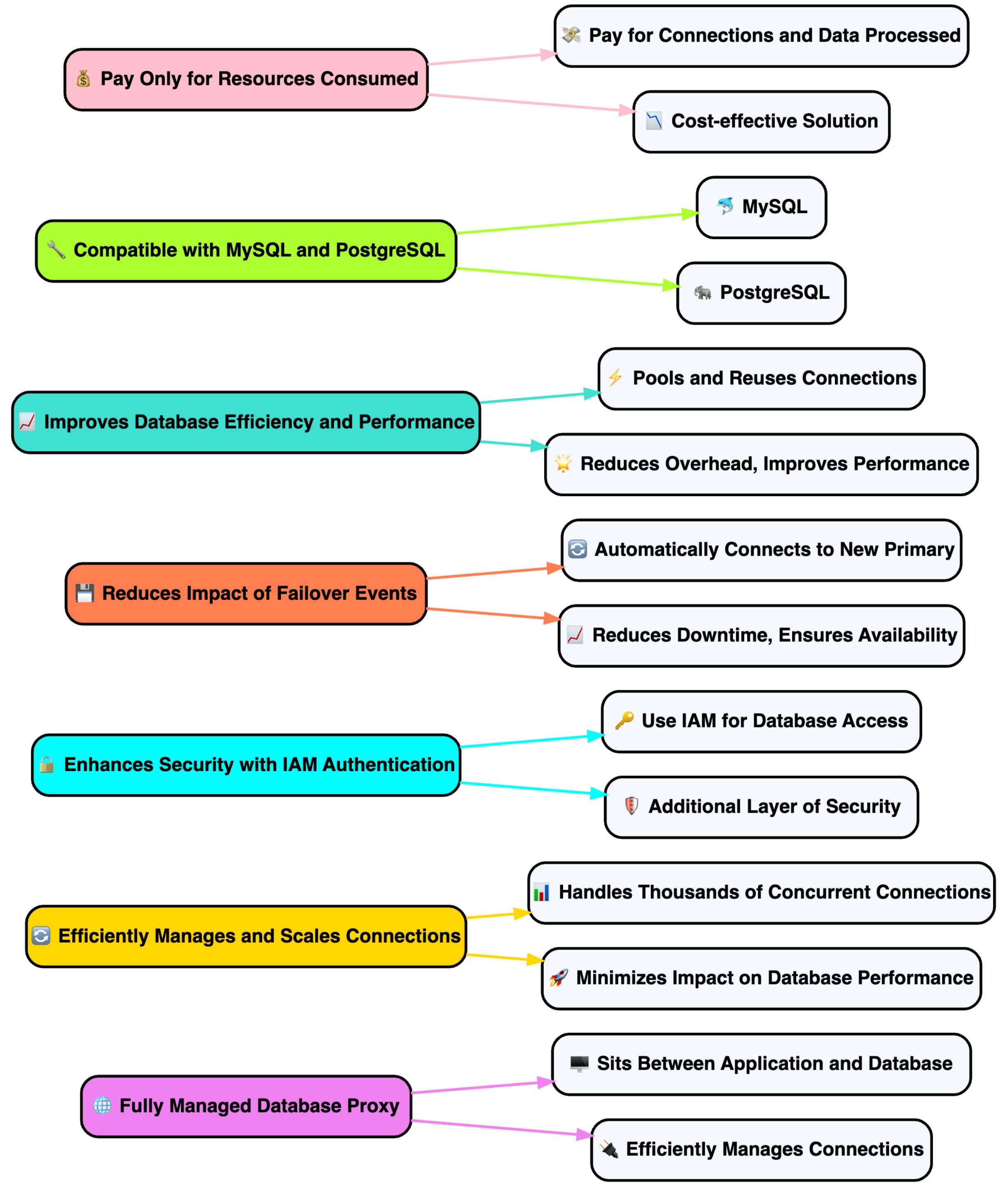
Performance optimizations

Essential RDS features

Simplifies database management tasks

Better integration with on-premises environments

Compliance with specific networking policies



- Amazon RDS Proxy: Enhancing Scalability and Security **
1.  **Fully managed database proxy for Amazon RDS:** 
Sits between application and database,  Efficiently manages connections
 2.  **Efficiently manages and scales database connections:**  Handles thousands of concurrent connections,  Minimizes impact on database performance
 3.  **Enhances application security with IAM authentication:**  Use IAM for database access,  Additional layer of security
 4.  **Reduces the impact of failover events:** 
Automatically connects to new primary,  Reduces downtime, ensures availability
 5.  **Improves database efficiency and application performance:**  Pools and reuses connections, 
Reduces overhead, improves performance
 6.  **Compatible with MySQL and PostgreSQL engines:** 
MySQL,  PostgreSQL
 7.  **Pay only for the RDS Proxy resources consumed:** 
Pay for connections and data processed,  Cost-effective solution



Secrets Manager Integration with Amazon RDS



1. Securely store and manage database credentials

Ensures sensitive information is protected

Usernames, passwords for RDS instances

2. Automatic rotation of database secrets

Reduces risk of unauthorized access

Ensures credentials are regularly updated

3. Access secrets via API or AWS SDKs

Retrieve secrets using Secrets Manager API or SDKs

Easy integration into application code

4. Seamless integration with Amazon RDS

Associate secrets with RDS instances

Centralized secrets management

5. Eliminates hard-coded credentials in application code

Reduces risk of accidental exposure

Improves security

6. Encrypts secrets using AWS KMS keys

Additional layer of security

Secrets protected at rest

7. Centralized secrets management for multiple RDS instances

Simplifies secrets management

Ensures consistency across database environment



DB Instance Billing for Amazon RDS



1. 💰 Pay only for resources consumed

⌚ DB instance hours, storage, data transfer

🔍 Cost optimization and flexibility

2. ⏳ Charged on a per-second basis

⌚ Minimum of 10 minutes

⌚ Pay for precise running time

3. 🔧 Multiple pricing options available

🕒 On-Demand, Reserved, Spot Instances

💰 Choose cost-effective option based on workload

4. 📁 Storage billed separately based on allocated size

📦 Billed based on provisioned space

💾 Regardless of actual space used

5. ✅ Data transfer out incurs additional charges

🌐 To internet or other AWS regions

💰 Billed based on amount of data transferred

6. 💯 No charge for data transfer in

🌐 From internet or other AWS services within region

FREE No additional charge

7. 💵 Reserved Instances offer significant savings

📅 Commit to specific DB instance configuration and term

💰 Significant cost savings compared to On-Demand pricing



**Thanks
for
Watching**