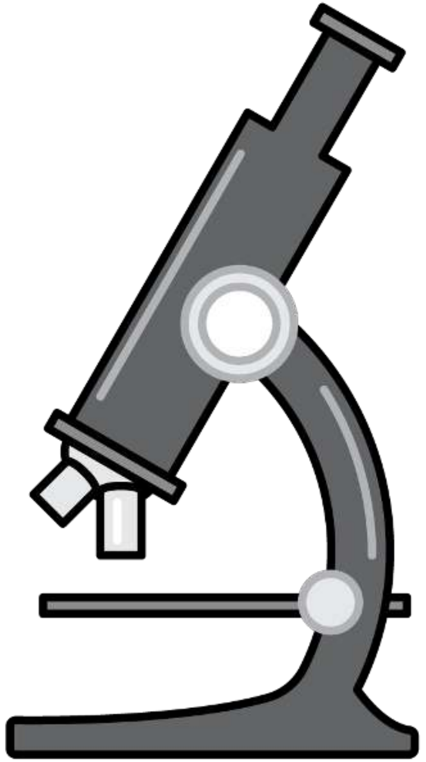
Deep Dive on Amazon RDS

Julien Simon, Principal Technical Evangelist, AWS
julsimon@amazon.fr
@julsimon

What to expect

- Amazon RDS overview (super quick)
- Security
- Metrics and monitoring
- High availability
- Scaling on RDS
- Backups and snapshots
- Migrating to RDS
- Q&A



Amazon Relational Database Service (Amazon RDS)

Launch

No infrastructure management

Application compatibility

Instant provisioning

Cost-effective

AWS Free Tier

Scale up/down

https://aws.amazon.com/rds/whats-new/

Amazon RDS engines

Commercial

ORACLE

Microsoft SQL Server

Open source

MySQL

PostgreSQL

MariaDB

Amazon Aurora

Amazon Aurora

Selected Amazon RDS customers

SAMSUNG

vodafone

intuit

LAMBORGHINI

Unilever

airbnb

ZIFF DAVIS

SEGA®

Kempinski
HOTELIERS SINCE 1897

ROVIO

FCBARCELONA
més que un club

OUTBACK
STEAKHOUSE®

OVERSEAS VOTE
FOUNDATION

Newsweek
& THE DAILY BEAST

The
Washington Post
Company

Trinity Mirror plc

coursera

C O
D E

ZUMBA®
FITNESS

Selected Amazon Aurora customers

Expedia

GE Oil & Gas

ZUMBA FITNESS

ISCS

Earth Networks
Taking the Pulse of the Planet

TalentBin
BY MONSTER

Alfresco

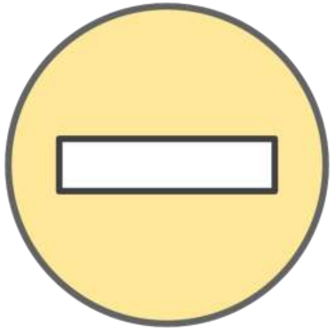
Trade-offs with a managed service

Fully managed host and OS

- No access to the database host operating system
- Limited ability to modify configuration that is managed on the host operating system
- No functions that rely on configuration from the host OS

Fully managed storage

- Max storage limits
  - Microsoft SQL Server—4 TB
  - MySQL, MariaDB, PostgreSQL, Oracle—6 TB
  - Aurora—64 TB
- Growing your database is a process



Amazon RDS: the fine print ☺

PostgreSQL

- Using the rds_superuser Role
- Supported PostgreSQL Database Versions
- Supported PostgreSQL Features and Extensions
- Limits for PostgreSQL DB Instances
- Upgrading a PostgreSQL DB Instance
- Using SSL with a PostgreSQL DB Instance
- Creating Roles
- Managing PostgreSQL Database Access
- Working with PostgreSQL Parameters
- Working with PostgreSQL Autovacuum on Amazon RDS
- Audit Logging for a PostgreSQL DB Instance
- Setting up PostGIS
- Using pgBadger for Log Analysis with PostgreSQL
- Viewing the Contents of pg_config

http://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/CHAP_PostgreSQL.html#PostgreSQL.Concepts.General.FeatureSupport
http://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Appendix.PostgreSQL.CommonDBATasks.html

MySQL

- Killing a Session or Query
- Skipping the Current Replication Error
- Working with InnoDB Tablespaces to Improve Crash Recovery Times
- Managing the Global Status History

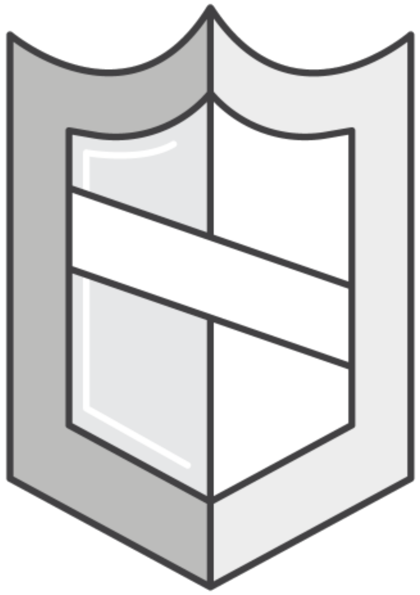
http://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Appendix.MySQL.CommonDBATasks.html

MariaDB

Appendix: Parameters for MariaDB

http://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Appendix.MariaDB.Parameters.html

Security



Amazon Virtual Private Cloud (VPC)

Securely control network configuration

Manage connectivity

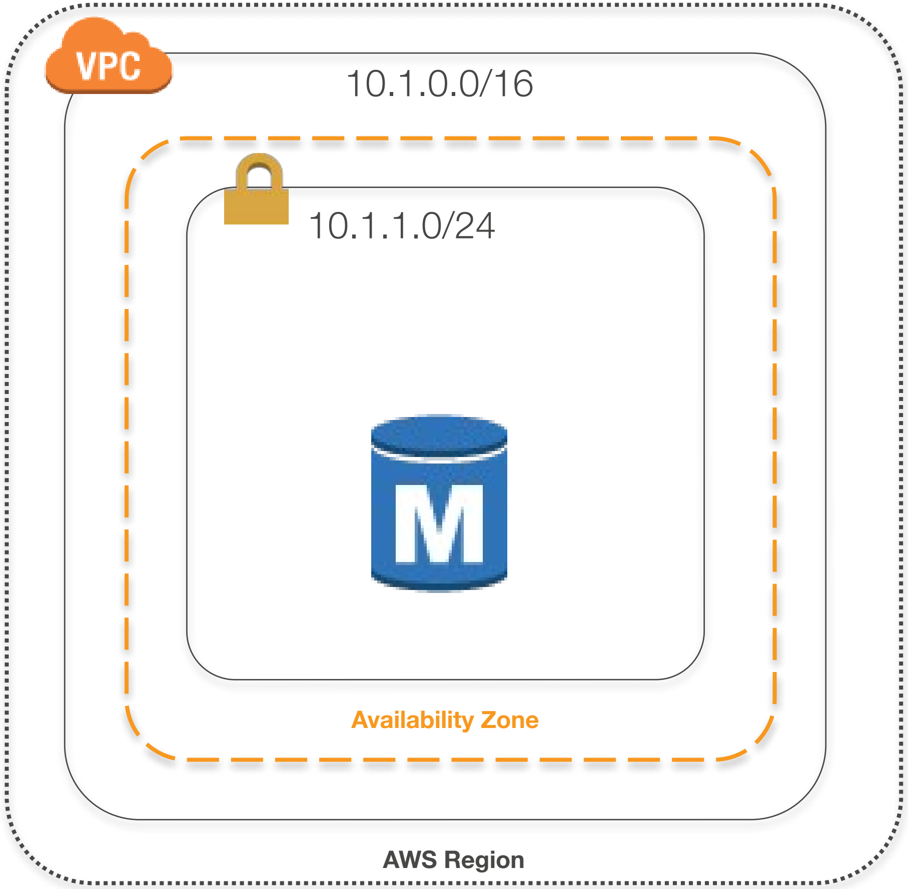
AWS Direct Connect

VPN connection

VPC peering

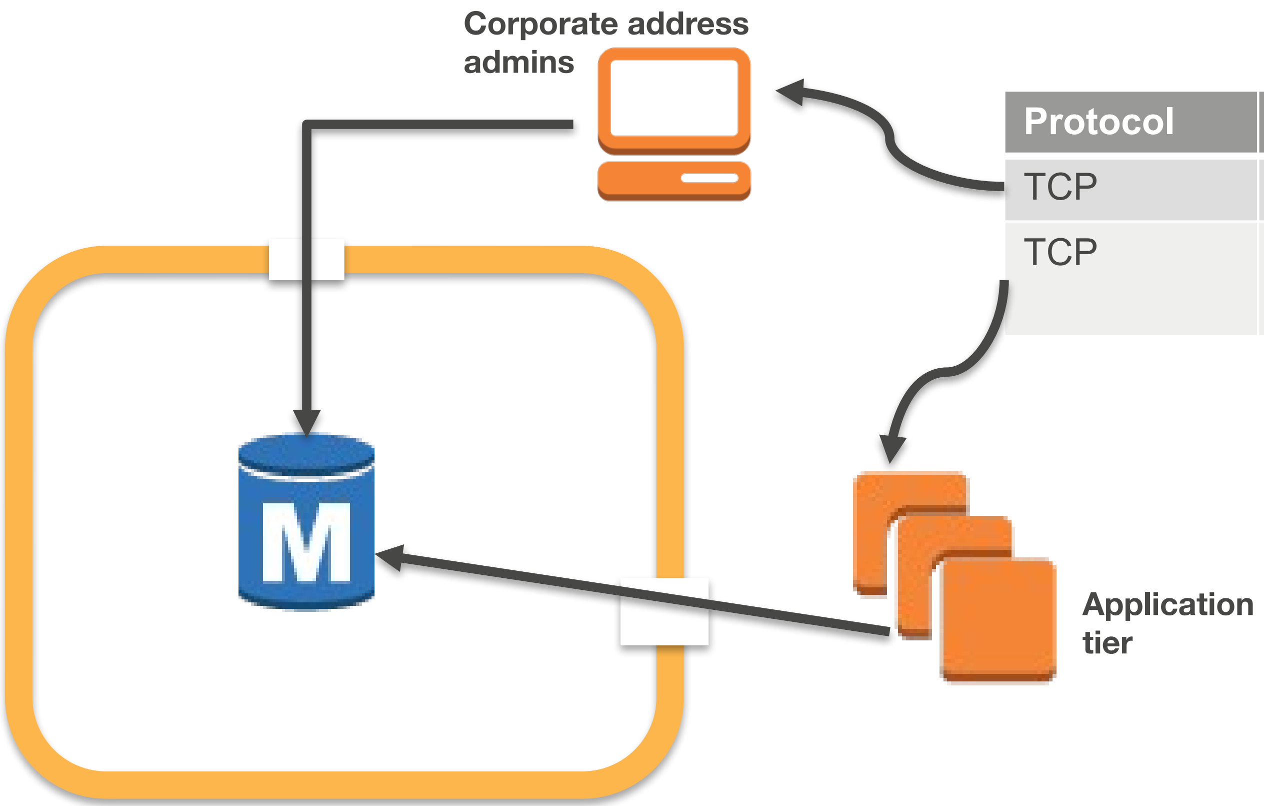
Routing rules

Internet gateway



Security groups

Database IP firewall protection



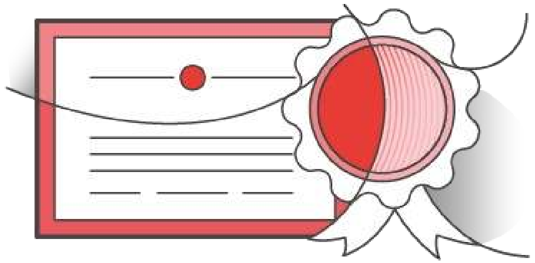
Protocol	Port Range	Source
TCP	3306	172.31.0.0/16
TCP	3306	“Application security group”

Compliance

AICPA SOC

FedRAMP
Federal Risk Authorization Management Program

GOV.UK



Singapore MTCS

PCI DSS
LEVEL 1 SERVICE PROVIDER

ITAR

HIPAA

ISO

27001/9001
27017/27018

https://aws.amazon.com/compliance/

Compliance

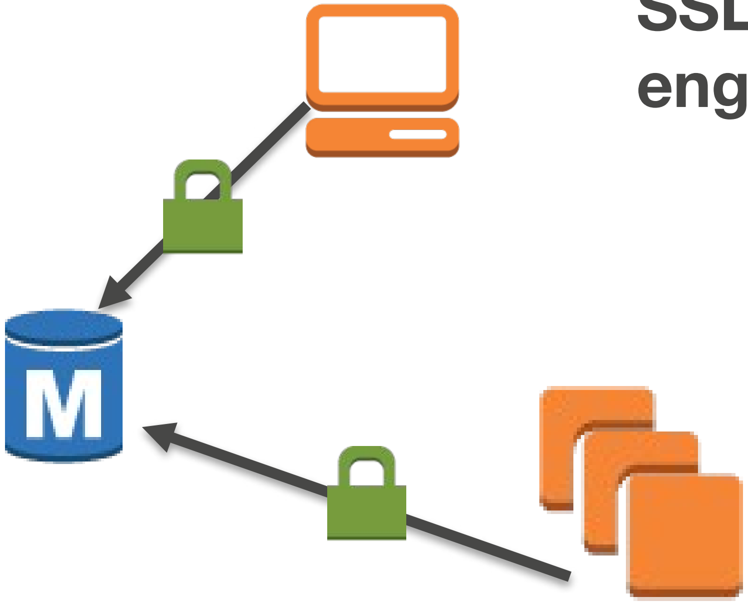
MySQL and Oracle

- SOC 1, 2, and 3
- ISO 27001/9001
- ISO 27017/27018
- PCI DSS
- FedRAMP
- HIPAA BAA
- UK government programs
- Singapore MTCS

SQL Server and PostgreSQL

- SOC 1, 2, and 3
- ISO 27001/9001
- ISO 27017/27018
- PCI DSS
- UK government programs
- Singapore MTCS

In-flight data encryption

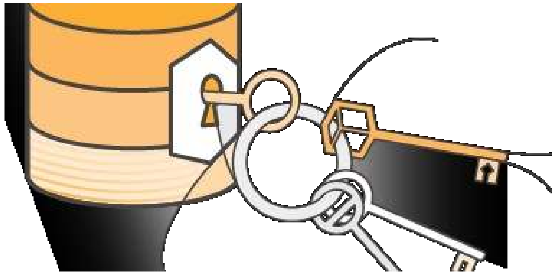


SSL available for all six engines

http://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/UsingWithRDS.SSL.html

At-rest data encryption

- DB instance storage
- Automated backups
- Read Replicas
- Snapshots



- **Available for all six engines**
- **No additional cost**
- **Support compliance requirements**
- **TDE also available for Oracle / SQL Server**

http://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Overview.Encryption.html

Amazon RDS encryption hints

- You can only encrypt on new database creation
- Encryption cannot be removed
- Master and Read Replica must be encrypted
- Unencrypted snapshots cannot be restored to encrypted DB
    - Aurora will allow this
    - You can create encrypted copies of your unencrypted snapshots
- Cannot restore MySQL to Aurora or Aurora to MySQL
- Cannot copy snapshots or replicate DB across regions

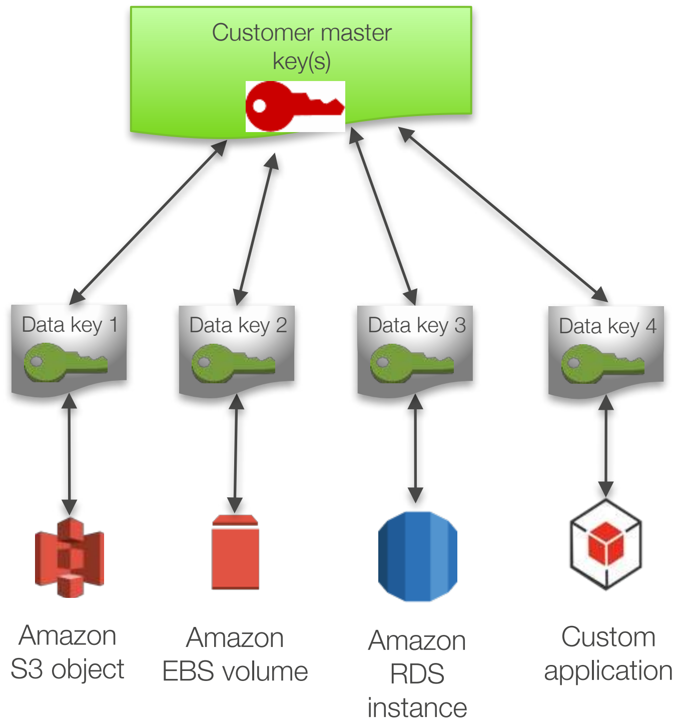
AWS KMS—RDS standard encryption

Two-tiered key hierarchy using envelope encryption:

- Unique data key encrypts customer data
- AWS KMS master keys encrypt data keys

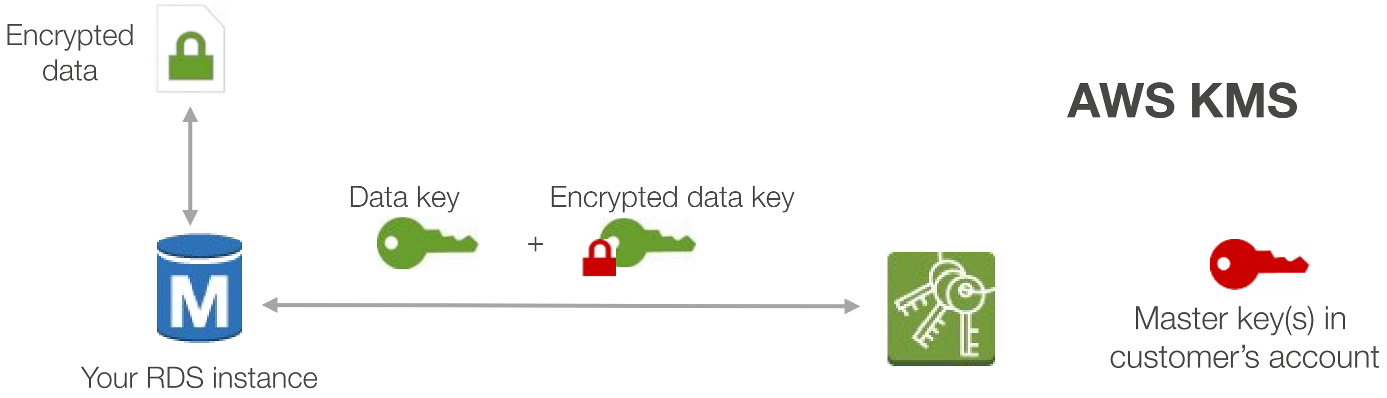
Benefits:

- Limits risk of compromised data key
- Better performance for encrypting large data
- Easier to manage small number of master keys than millions of data keys
- Centralized access and audit of key activity



https://aws.amazon.com/kms/

How keys are used to protect your data



1. RDS instance requests encryption key to use to encrypt data, passes reference to master key in account
2. Client request authenticated based on permissions set on both the user and the key
3. A unique data encryption key is created and encrypted under the KMS master key
4. Plaintext and encrypted data key returned to the client
5. Plaintext data key used to encrypt data and then deleted when practical
6. Encrypted data key is stored; it's sent back to KMS when needed for data decryption

https://aws.amazon.com/kms/

Enabling encryption with the AWS CLI

```
aws rds create-db-instance --region us-west-2 --db-instance-identifier sg-cli-test \
--allocated-storage 20 --storage-encrypted \
--db-instance-class db.m4.large --engine mysql \
--master-username myawsuser --master-user-password myawsuser
```

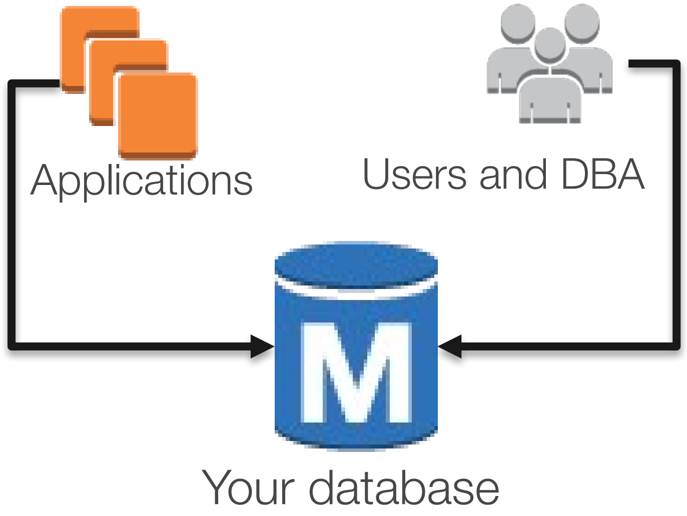
```
aws rds create-db-instance --region us-west-2 --db-instance-identifier sg-cli-test1 \
--allocated-storage 20 --storage-encrypted  --kms-key-id xxxxxxxxxxxxxxxxxx \
--db-instance-class db.m4.large --engine mysql \
--master-username myawsuser   --master-user-password myawsuser
```

http://docs.aws.amazon.com/cli/latest/reference/rds/create-db-instance.html

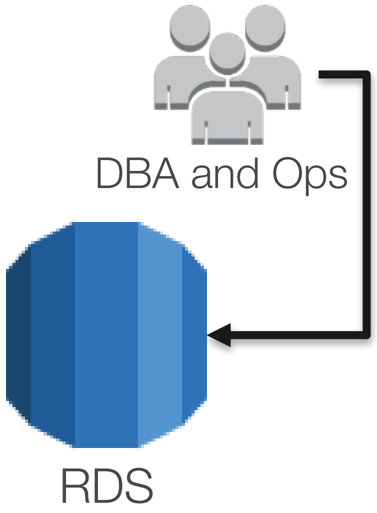
IAM governed access

You can use AWS Identity and Access Management (IAM) to control who can perform actions on RDS

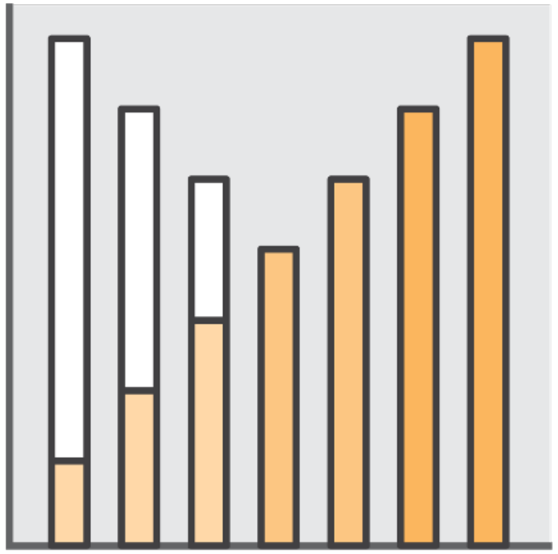
Controlled with database GRANTs



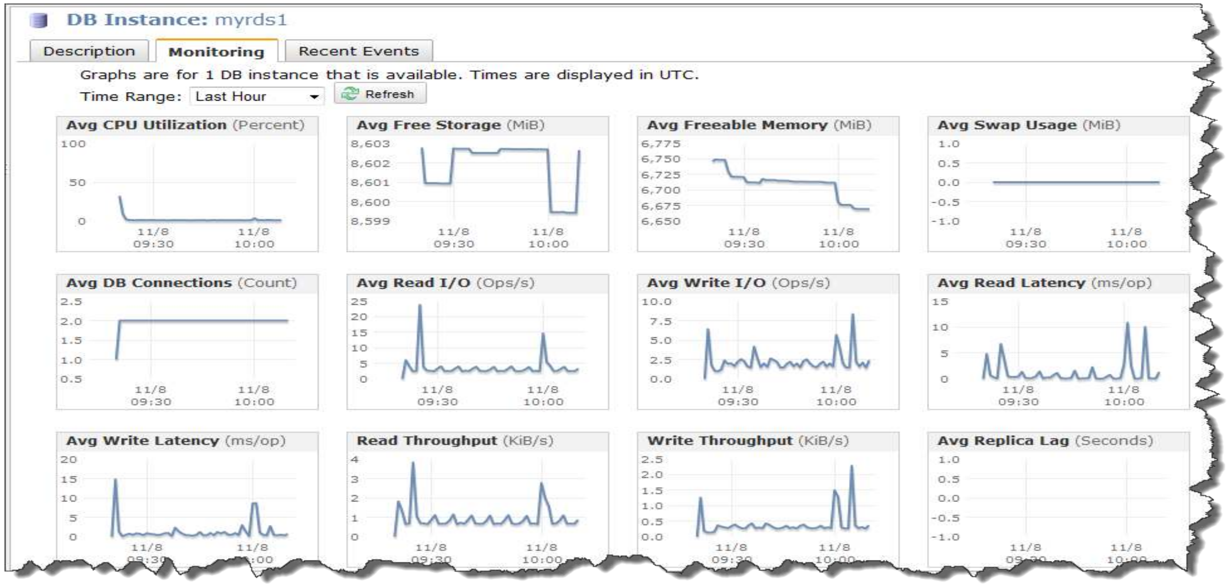
Controlled with IAM



Metrics and monitoring



Standard monitoring


DB Instance: myrds1

Graphs are for 1 DB instance that is available. Times are displayed in UTC.

Time Range: Last Hour

*** NEW (Nov’11) price drop, longer retention & percentile monitoring

https://aws.amazon.com/about-aws/whats-new/2016/11/announcing-cloudwatch-metrics-price-reduction-and-new-volume-based-pricing-tiers/

https://aws.amazon.com/blogs/aws/amazon-cloudwatch-update-percentile-statistics-and-new-dashboard-widgets/

https://aws.amazon.com/about-aws/whats-new/2016/11/cloudwatch-extends-metrics-retention-and-new-user-interface/

Amazon CloudWatch metrics for Amazon RDS

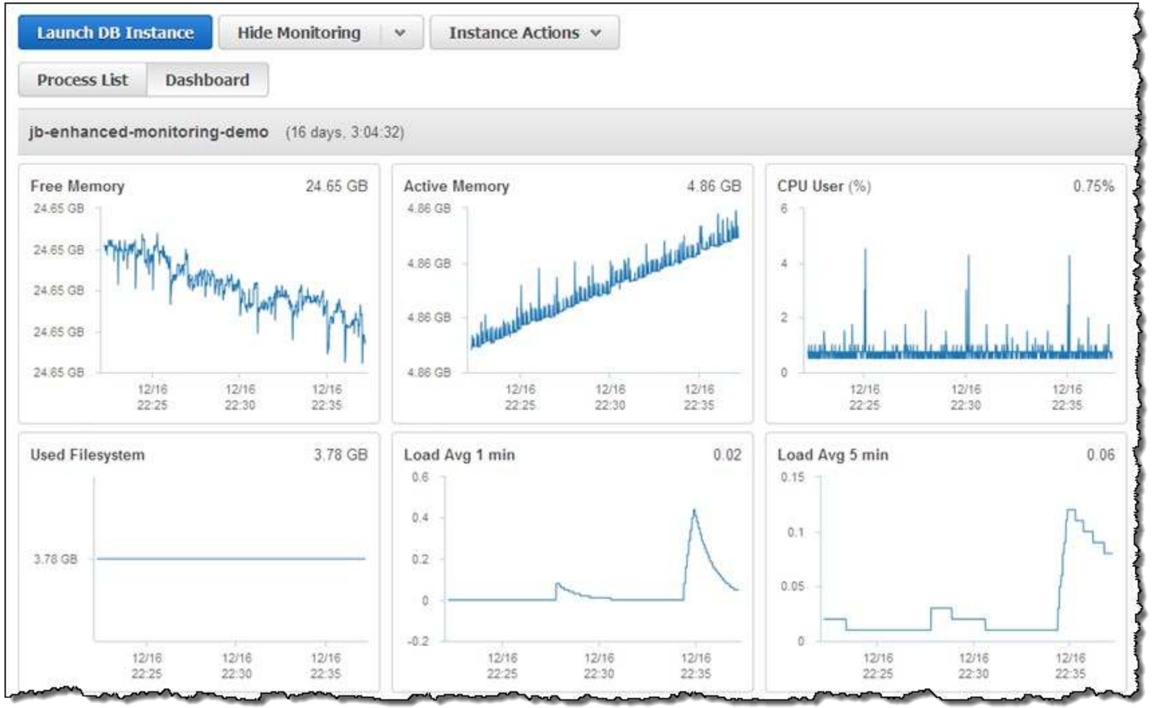
- CPU utilization
- Storage
- Memory
- Swap usage
- DB connections
- I/O (read and write)
- Latency (read and write)
- Throughput (read and write)
- Replica lag
- Many more

Amazon CloudWatch Alarms

- Similar to on-premises custom monitoring tools

Enhanced Monitoring

Access to over 50 new CPU, memory, file system, and disk I/O metrics as low as 1 second intervals (sent to CloudWatch Logs)



jb-enhanced-monitoring-demo (16 days, 3:04:32)

Free Memory 24.65 GB

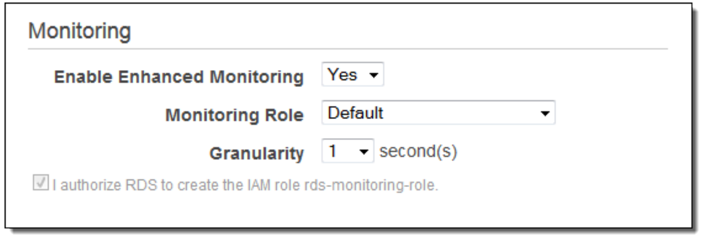
Active Memory 4.86 GB

CPU User (%) 0.75%

Used Filesystem 3.78 GB

Load Avg 1 min 0.02

Load Avg 5 min 0.06



Monitoring

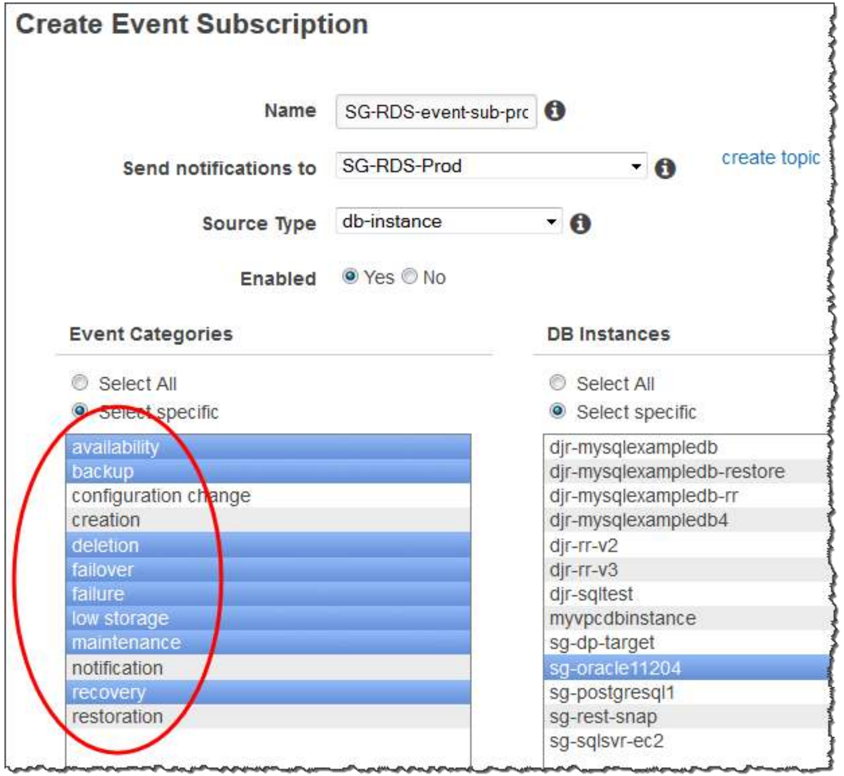
Enable Enhanced Monitoring	Yes
Monitoring Role	Default
Granularity	1 second(s)

☑ I authorize RDS to create the IAM role rds-monitoring-role.

http://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/USER_Monitoring.OS.html

Event notifications

- Uses Amazon Simple Notification Service (Amazon SNS) to notify users when an event occurs
- 17 different event categories (availability, backup, configuration change, and so on)



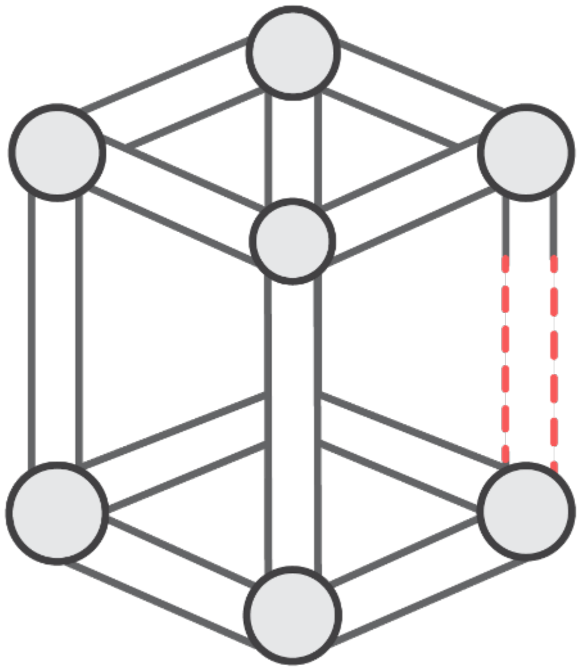
Create Event Subscription

Name	SG-RDS-event-sub-prc
Send notifications to	SG-RDS-Prod
Source Type	db-instance
Enabled	[✓ Yes] No

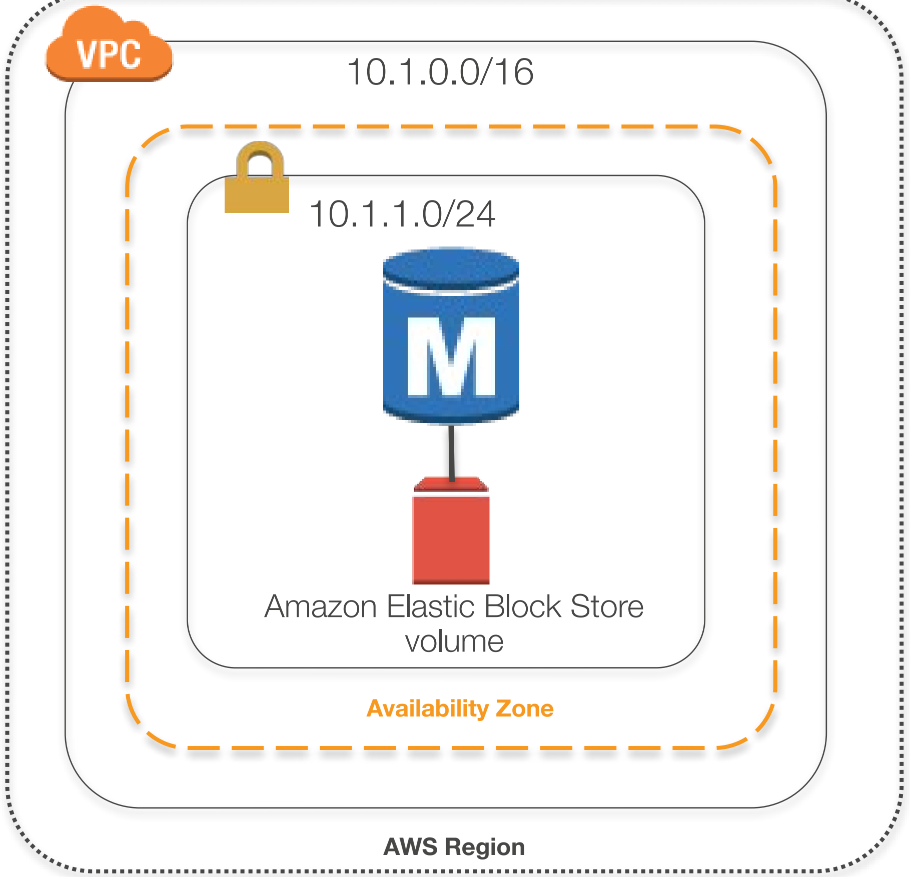
create topic

Event Categories	DB Instances
Select All	Select All
[✓ Select specific]	[✓ Select specific]
[✓ availability]	djr-mysqlexampledb
[✓ backup]	djr-mysqlexampledb-restore
configuration change	djr-mysqlexampledb-rr
creation	djr-mysqlexampledb4
[✓ deletion]	djr-rr-v2
[✓ failover]	djr-rr-v3
[✓ failure]	djr-sqltest
[✓ low storage]	myvpcdbinstance
[✓ maintenance]	sg-dp-target
notification	[✓ sg-oracle11204]
[✓ recovery]	sg-postgresql1
restoration	sg-rest-snap
	sg-sqlsvr-ec2

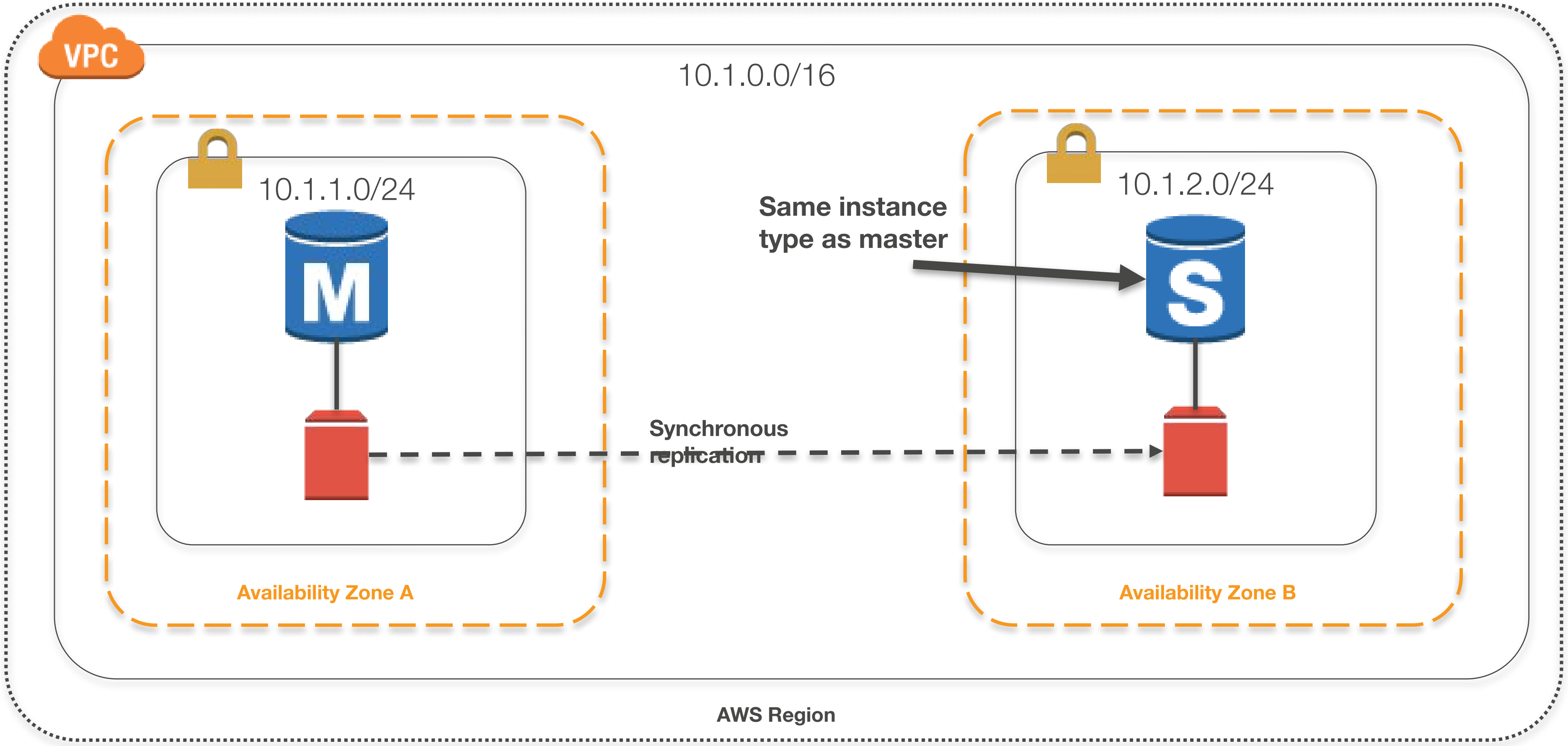
High availability



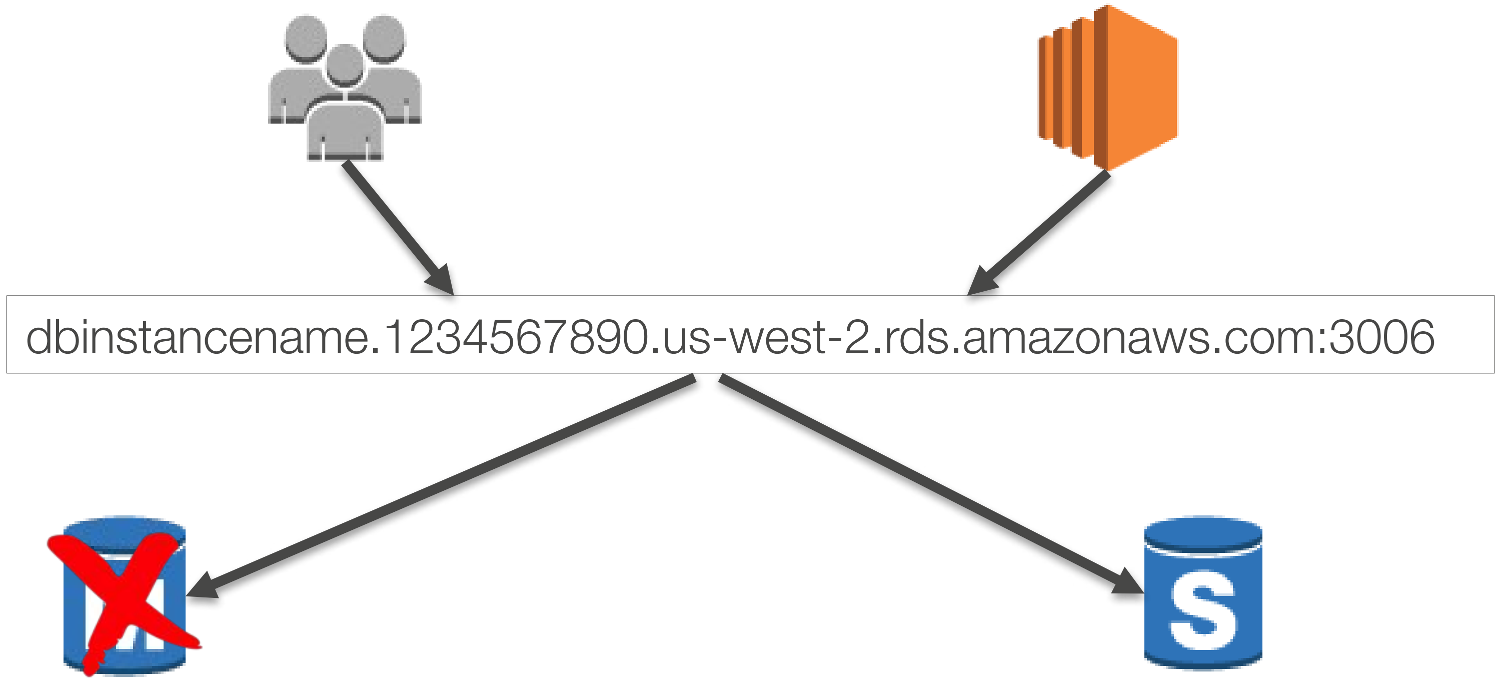
Minimal deployment—single AZ



High availability—Multi-AZ

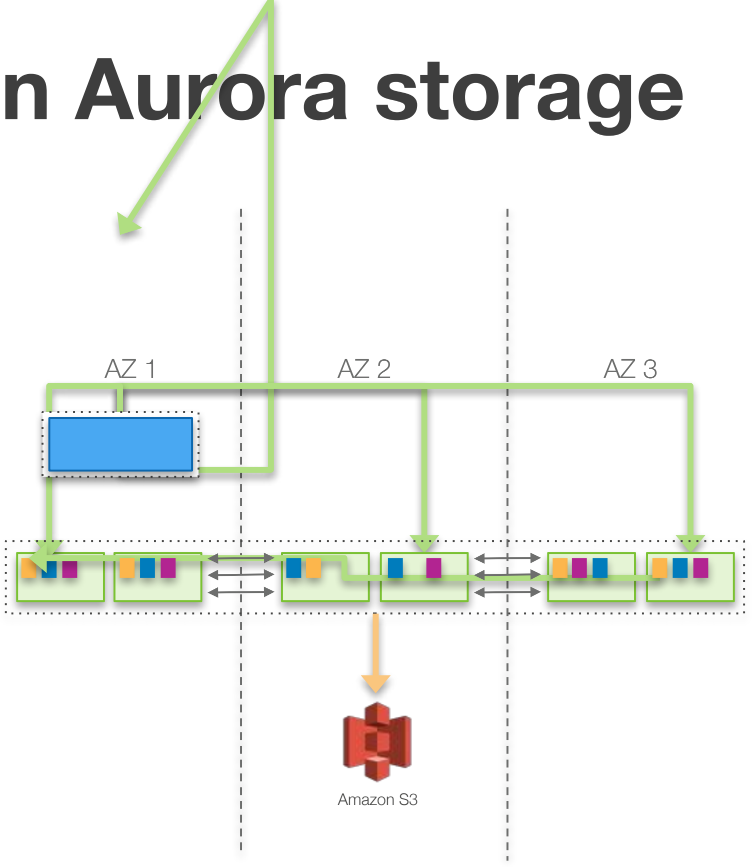


High availability—Multi-AZ to DNS



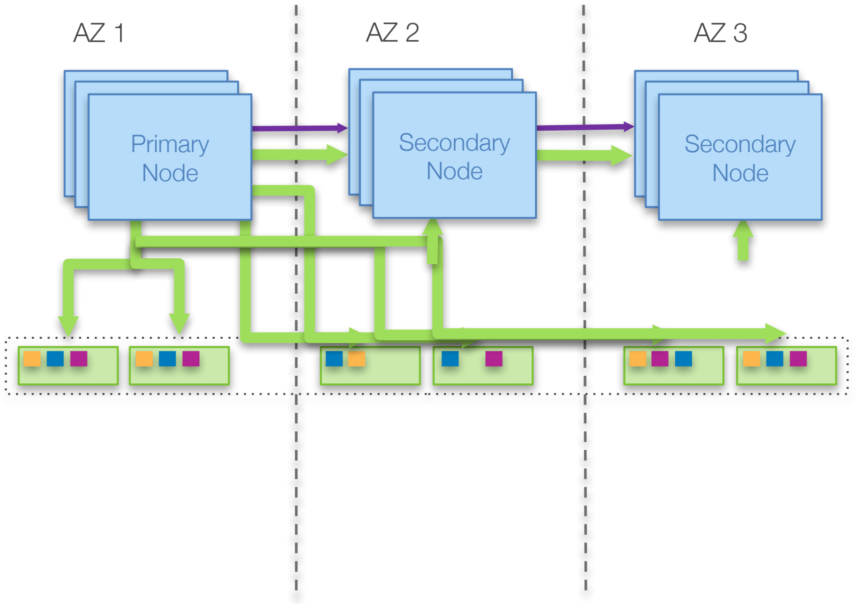
High availability—Amazon Aurora storage

- Storage volume automatically grows up to 64 TB
- 6 copies across 3 AZs
- Quorum system for read/write; latency tolerant
- Peer-to-peer gossip replication to fill in holes
- Continuous backup to Amazon S3 (built for 11 9s durability)
- Continuous monitoring of nodes and disks for repair
- 10 GB segments as unit of repair or hotspot rebalance
- Quorum membership changes do not stall writes

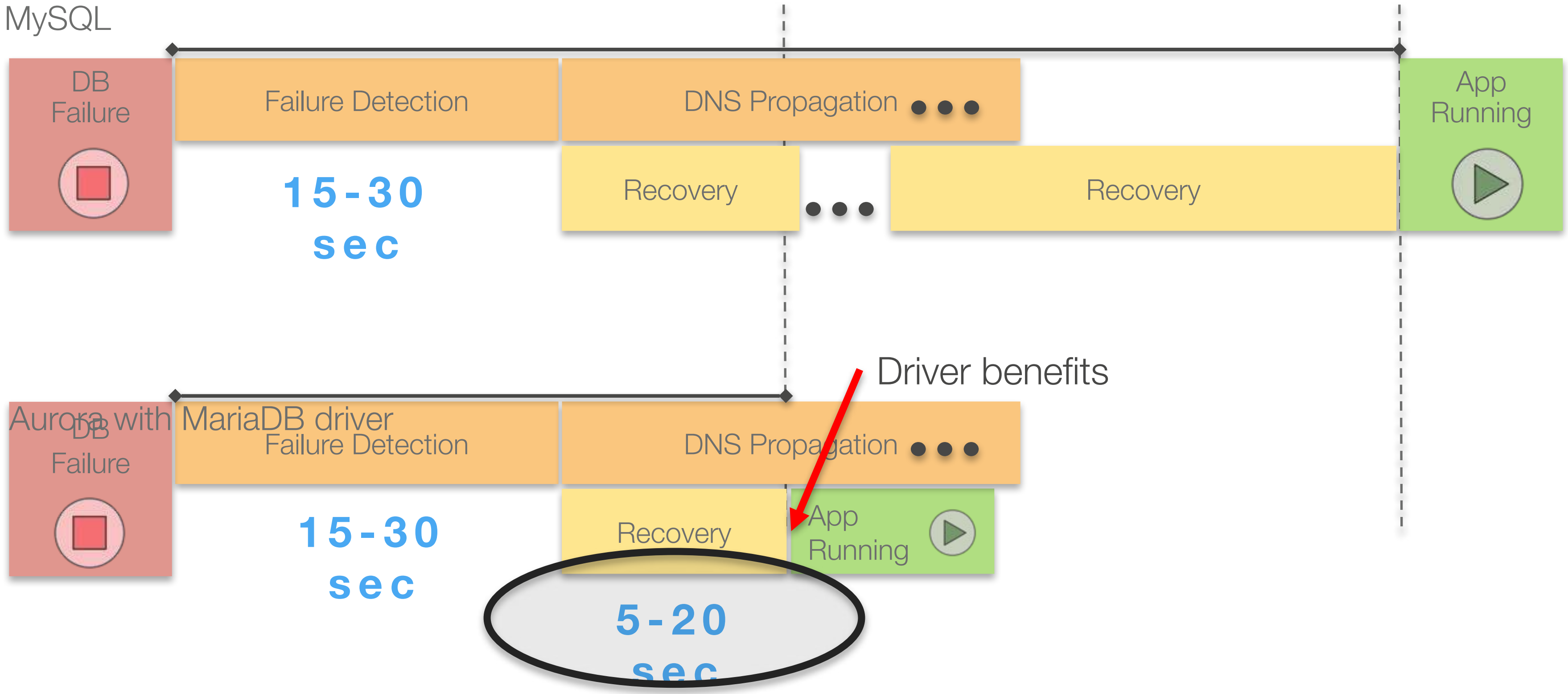


High availability—Aurora nodes

- Aurora cluster contains primary node and up to 15 secondary nodes
- Failing database nodes are automatically detected and replaced
- Failing database processes are automatically detected and recycled
- Secondary nodes automatically promoted on persistent outage, no single point of failure
- Customer application can scale out read traffic across secondary nodes



Failover – MySQL vs Aurora



https://mariadb.com/kb/en/mariadb/failover-and-high-availability-with-mariadb-connector-j/
https://mariadb.com/kb/en/mariadb/about-mariadb-connector-j/

Tips to improve recovery time with MySQL

- DO NOT use the IP address to connect to RDS!
- Set a low TTL on your own CNAME (beware if you use Java)
- Avoid large number of tables :
    - No more than 1000 tables using Standard Storage
    - No more than 10,000 tables using Provisioned IOPS
- Avoid very large tables in your database
- Avoid large transactions
- Make sure you have enough IOPS for recovery
- Use RDS Events to be notified

Simulating Amazon Aurora failures

ALTER SYSTEM CRASH [INSTANCE | DISPATCHER | NODE];

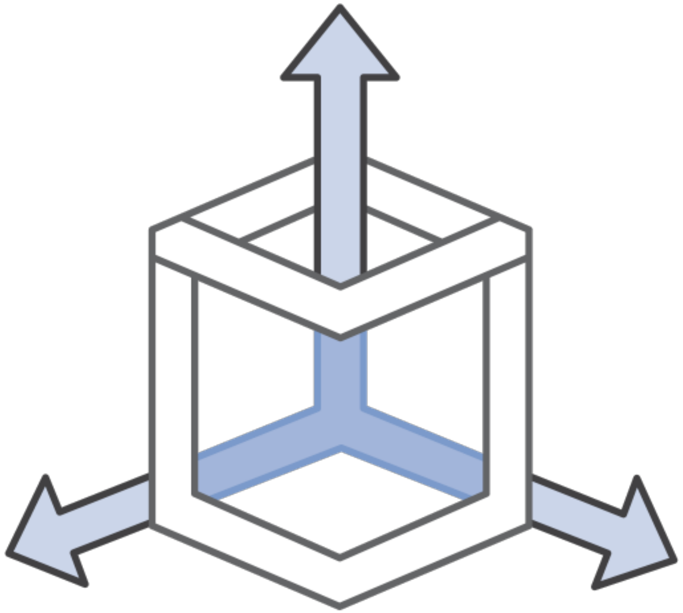
ALTER SYSTEM SIMULATE *percentage_of_failure* PERCENT

- **READ REPLICA FAILURE** [TO ALL | TO "replica name"]
- **DISK FAILURE** [IN DISK *index* | NODE *index*]
- **DISK CONGESTION** BETWEEN *minimum* AND *maximum* MILLISECONDS [IN DISK *index* | NODE *index*]

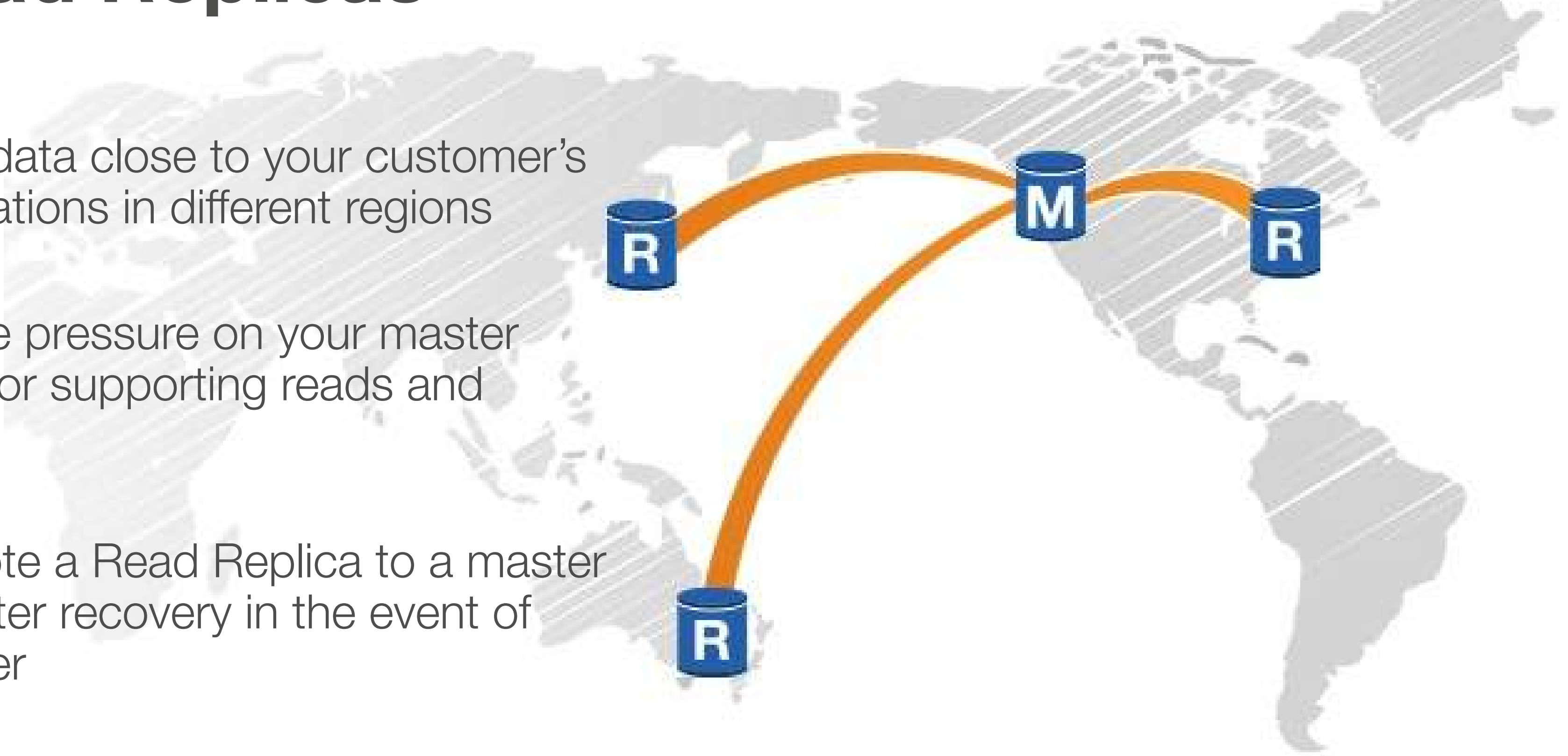
FOR INTERVAL *quantity* [YEAR | QUARTER | MONTH | WEEK| DAY | HOUR | MINUTE | SECOND];

http://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Aurora.Managing.html

Scaling on RDS



Read Replicas



Bring data close to your customer's applications in different regions

Relieve pressure on your master node for supporting reads and writes

Promote a Read Replica to a master for faster recovery in the event of disaster

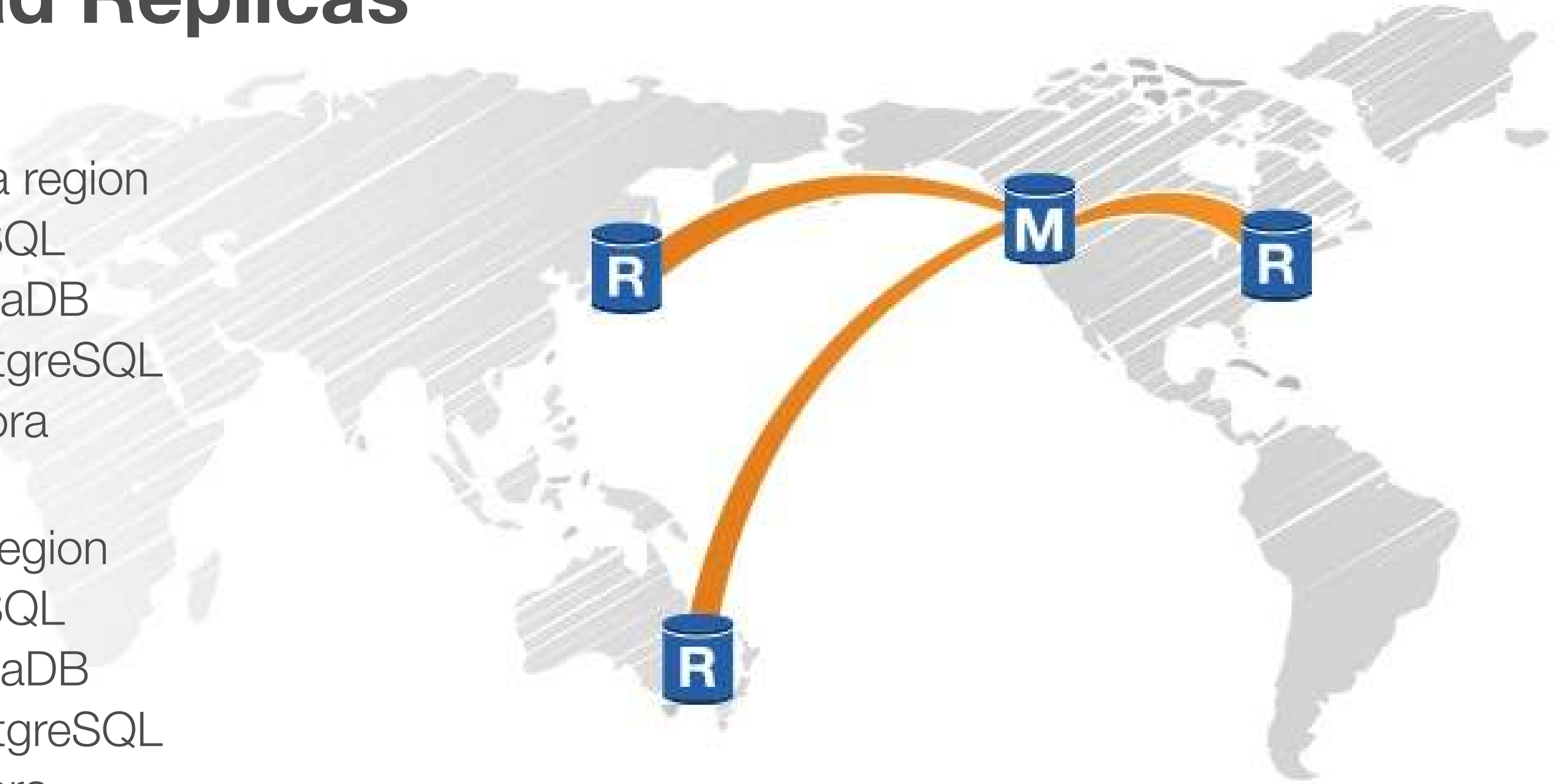
Read Replicas

Within a region

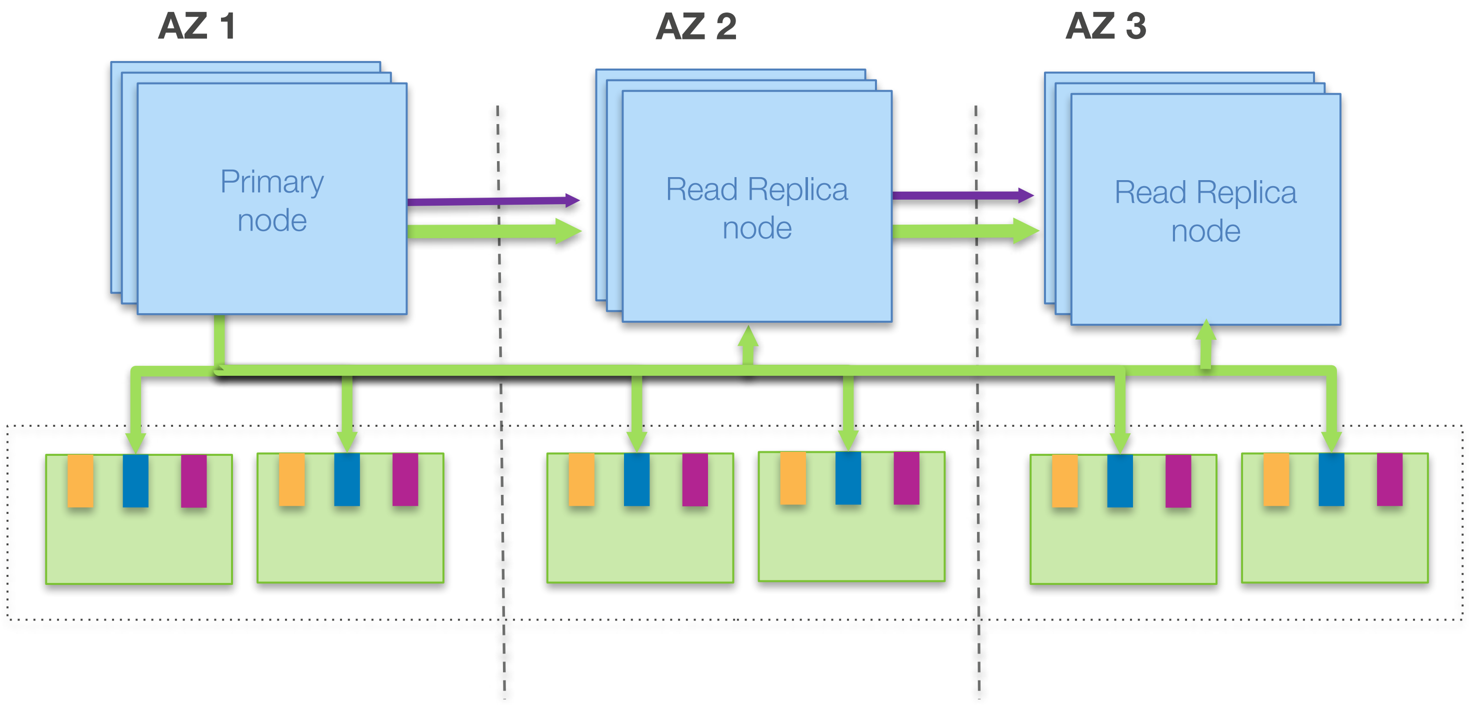
- MySQL
- MariaDB
- PostgreSQL
- Aurora

Cross-region

- MySQL
- MariaDB
- PostgreSQL
- Aurora



Read Replicas for Amazon Aurora



Read Replicas—Oracle and SQL Server

Options

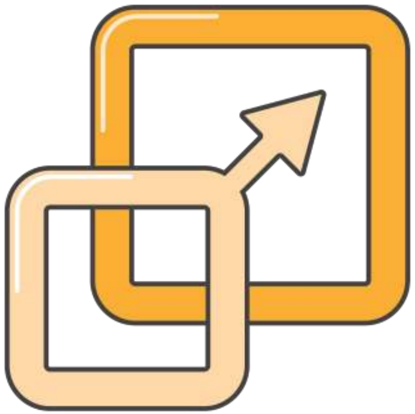
- Oracle GoldenGate
- Third-party replication products
- Snapshots

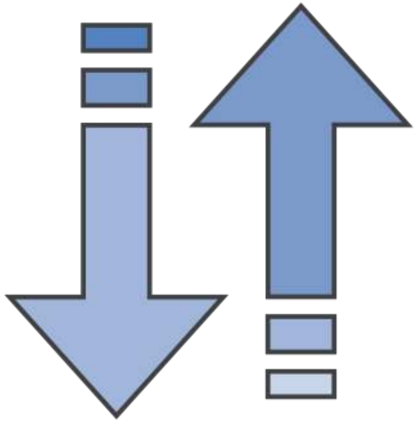
ORACLE®

Microsoft® SQL Server®

Scaling up—or down

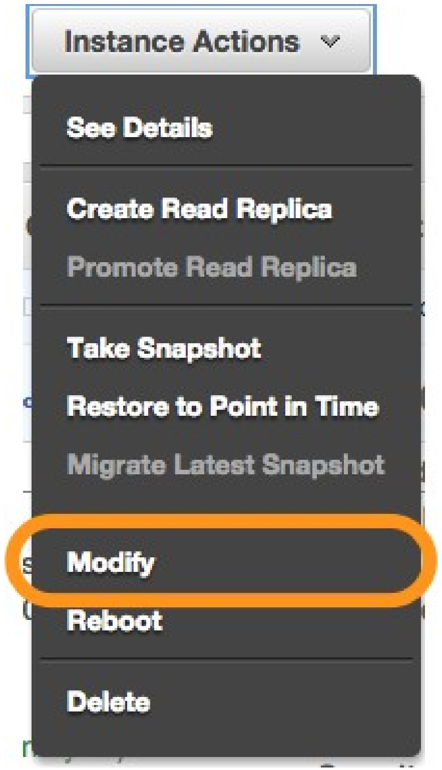
- Handle higher load or lower usage
- Control costs

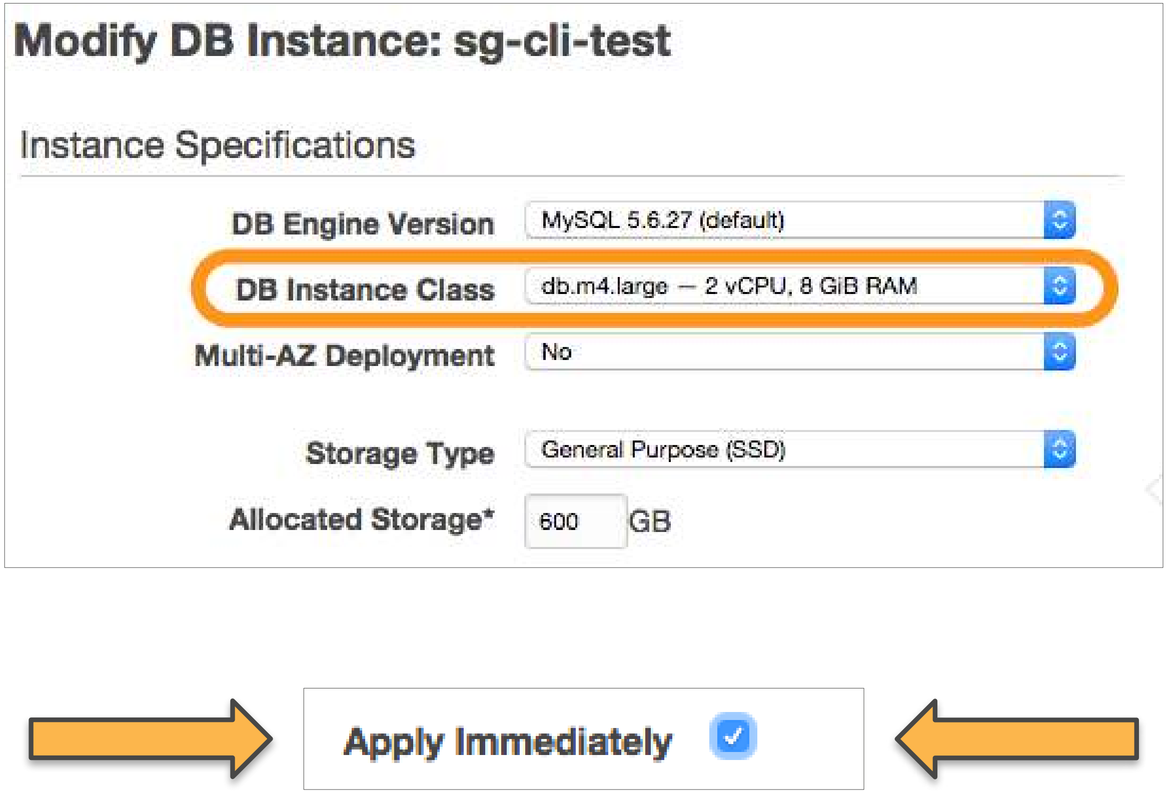




Scaling up—or down

AWS Management Console





Modify DB Instance: sg-cli-test

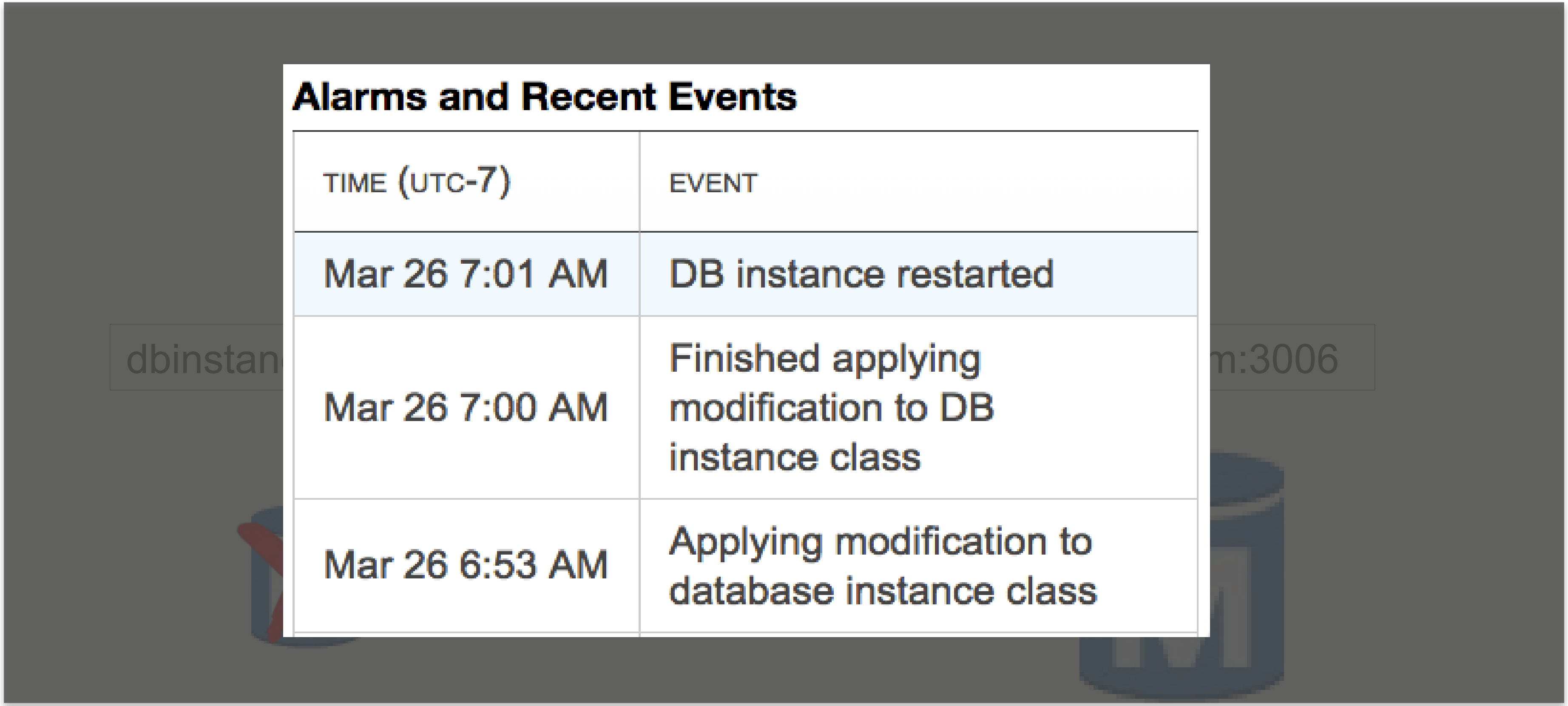
Instance Specifications

DB Engine Version	MySQL 5.6.27 (default)
DB Instance Class	db.m4.large — 2 vCPU, 8 GiB RAM
Multi-AZ Deployment	No
Storage Type	General Purpose (SSD)
Allocated Storage*	600 GB

Apply Immediately ✓

Scaling—single AZ

With single AZ deployment, the master takes an outage

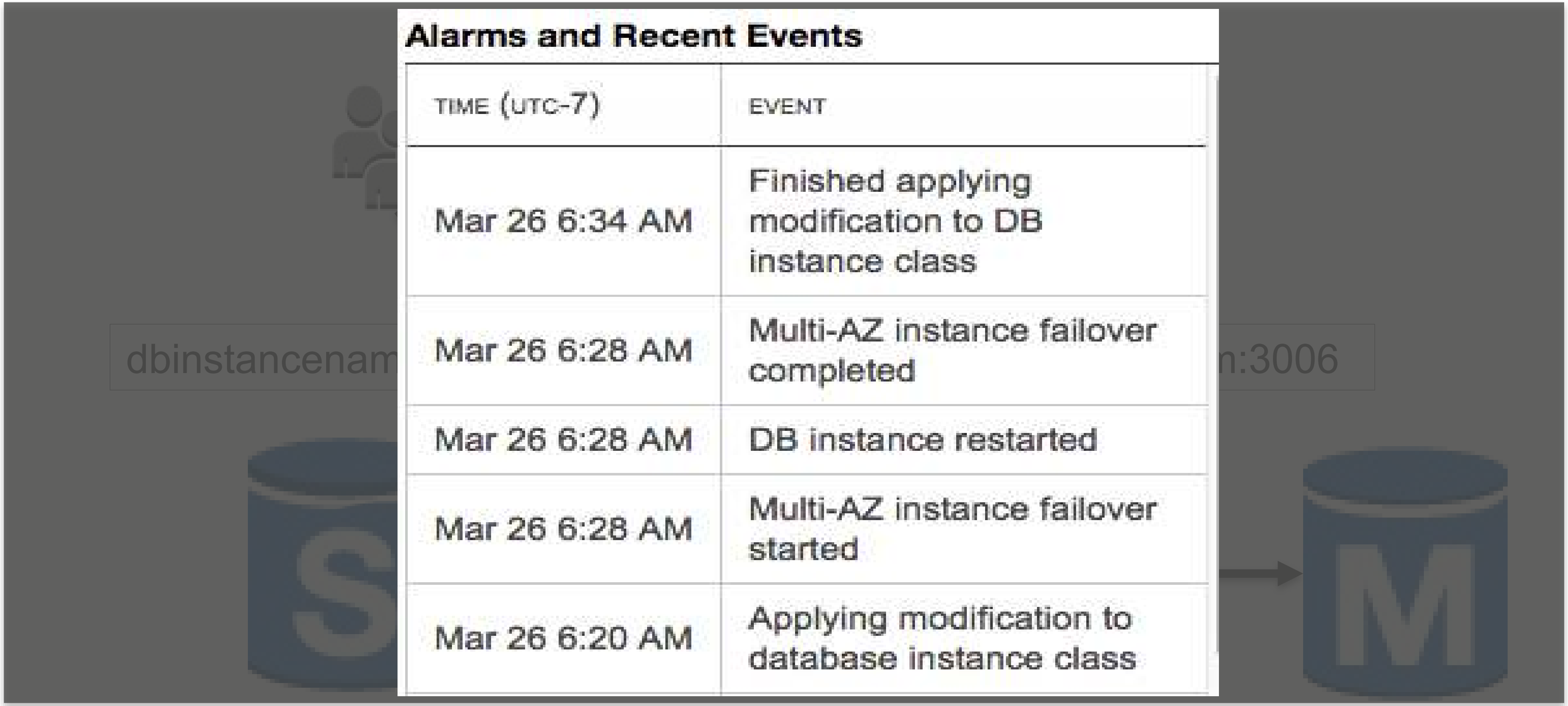


Alarms and Recent Events

TIME (UTC-7)	EVENT
Mar 26 7:01 AM	DB instance restarted
Mar 26 7:00 AM	Finished applying modification to DB instance class
Mar 26 6:53 AM	Applying modification to database instance class

Scaling—Multi-AZ

With Multi-AZ, the standby gets upgraded first



Alarms and Recent Events

TIME (UTC-7)	EVENT
Mar 26 6:34 AM	Finished applying modification to DB instance class
Mar 26 6:28 AM	Multi-AZ instance failover completed
Mar 26 6:28 AM	DB instance restarted
Mar 26 6:28 AM	Multi-AZ instance failover started
Mar 26 6:20 AM	Applying modification to database instance class

Scaling on a schedule – CLI or AWS Lambda

```
aws rds modify-db-instance
--db-instance-identifier sg-cli-test
--db-instance-class db.m4.large
--apply-immediately
```

```
#Scale down at 8:00 PM on Friday
0 20 * * 5
/home/ec2-user/scripts/scale_down_rds.sh

#Scale up at 4:00 AM on Monday
0 4 * * 1
/home/ec2-user/scripts/scale_up_rds.sh
```

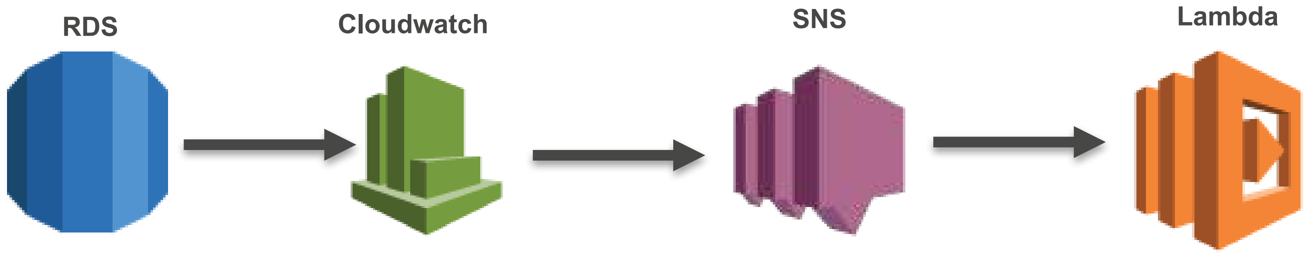
```
import boto3

client=boto3.client('rds')

def lambda_handler(event, context):
    response=client.modify_db_instance(DBInstanceIdentifier='sg-cli-test',
                                       DBInstanceClass='db.m4.xlarge',
                                       ApplyImmediately=True)


    print response
```

Scaling on demand – Cloudwatch & AWS Lambda

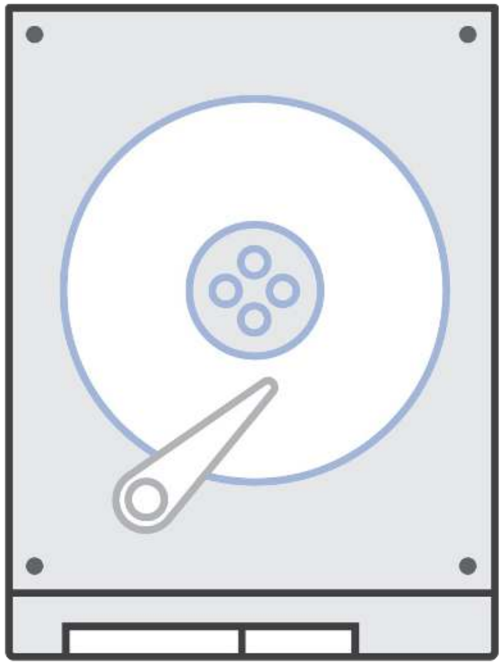


```
import boto3
import json

client=boto3.client('rds')

def lambda_handler(event, context):
      message = event['Records'][0]['Sns']['Message']
      parsed_message=json.loads(message)
      db_instance=parsed_message['Trigger']['Dimensions'][0]['value']
      print 'DB Instance: ' + db_instance
      response=client.modify_db_instance(DBInstanceIdentifier=db_instance,
                                         DBInstanceClass='db.m4.large',
                                         ApplyImmediately=True)
      print response
```

Backups and snapshots



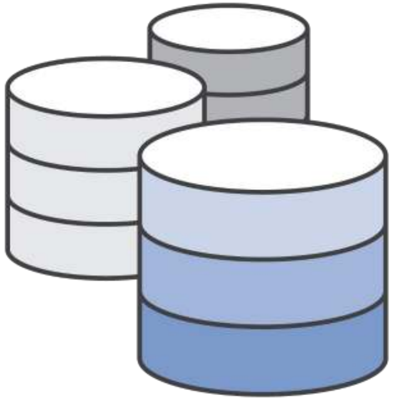
Backups

MySQL, PostgreSQL, MariaDB, Oracle, SQL Server

- Scheduled daily backup of entire instance
- Archive database change logs
- 35 day retention for backups
- Multiple copies in each AZ where you have instances

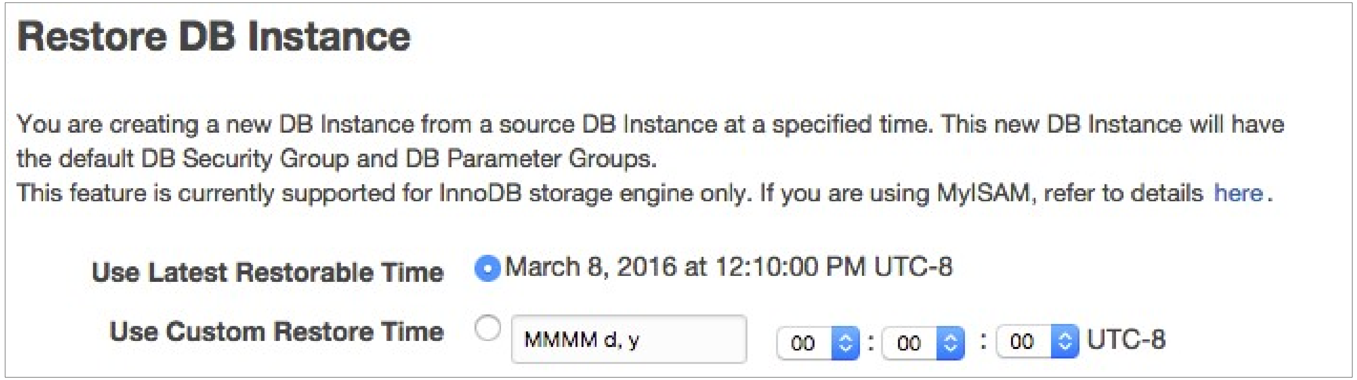
Aurora

- Automatic, continuous, incremental backups
- Point-in-time restore
- No impact on database performance
- 35 day retention



Restoring

- Restoring creates an entirely new database instance
- You define the instance configuration just like a new instance



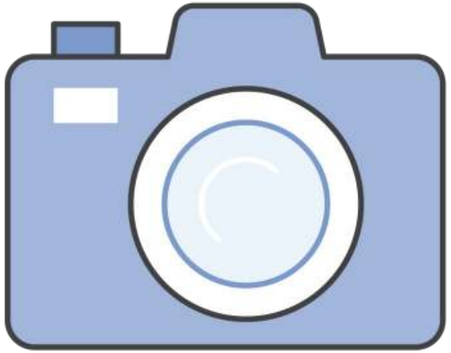
Restore DB Instance

You are creating a new DB Instance from a source DB Instance at a specified time. This new DB Instance will have the default DB Security Group and DB Parameter Groups.
This feature is currently supported for InnoDB storage engine only. If you are using MyISAM, refer to details here.

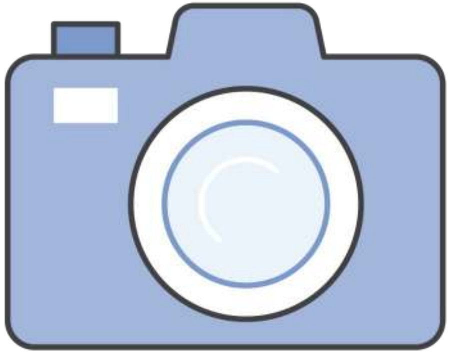
Use Latest Restorable Time	[✓] March 8, 2016 at 12:10:00 PM UTC-8
Use Custom Restore Time	MMMM d, y 00 : 00 : 00 UTC-8

Snapshots

- Full copies of your Amazon RDS database that are different from your scheduled backups
- Backed by Amazon S3
- Used to create a new RDS instance
- Remain encrypted if using encryption



Snapshots



Use cases

- Resolve production issues
- Build non-production environments
- Point-in-time restore
- Final copy before terminating a database
- Disaster recovery
- Cross-region copy
- Copy between accounts

Migrating onto RDS

AWS Database Migration Service

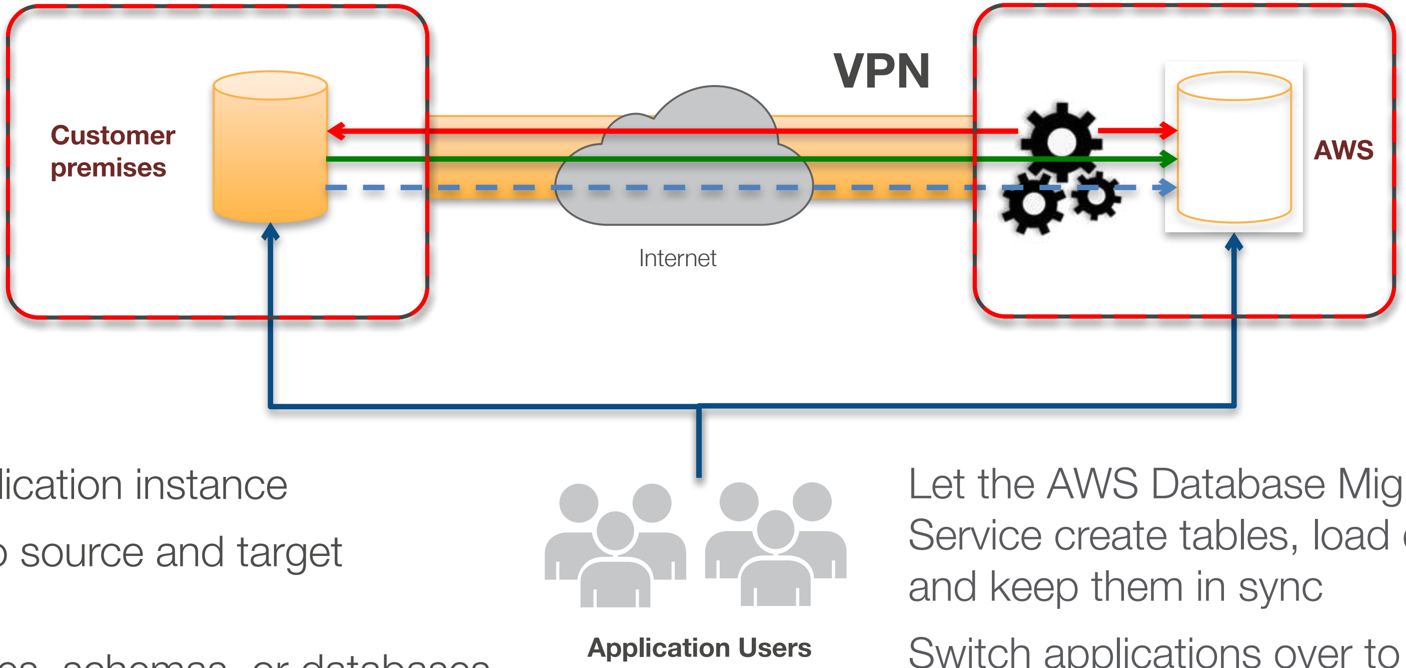
PostgreSQL ORACLE Amazon Aurora

AMAZON REDSHIFT MySQL MariaDB Microsoft SQL Server

- ✓ Move data to the same or different database engine
- ✓ Keep your apps running during the migration
- ✓ Start your first migration in 10 minutes or less
- ✓ Replicate within, to, or from Amazon EC2 or RDS

https://aws.amazon.com/dms/
http://docs.aws.amazon.com/dms/latest/userguide/CHAP_Introduction.Sources.html
http://docs.aws.amazon.com/dms/latest/userguide/CHAP_Introduction.Targets.html
https://aws.amazon.com/blogs/database/database-migration-what-do-you-need-to-know-before-you-start/

Keep your apps running during the migration



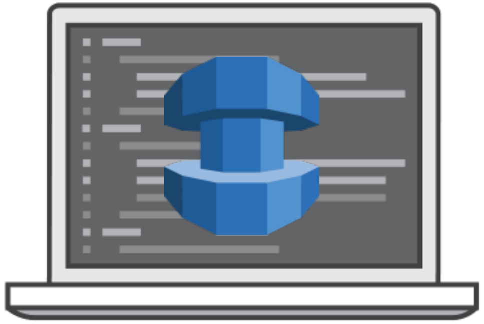
Start a replication instance

Connect to source and target database

Select tables, schemas, or databases

Let the AWS Database Migration Service create tables, load data, and keep them in sync

Switch applications over to the target at your convenience



AWS Schema Conversion Tool

- Move your tables, views, stored procedures, and data manipulation language (DML) to RDS or Amazon Redshift
- Highlight where manual edits are needed

Source Database	Target Database on Amazon RDS
Microsoft SQL Server	Amazon Aurora, MySQL, PostgreSQL, MariaDB
MySQL and MariaDB	PostgreSQL
Oracle	Amazon Aurora, MySQL, PostgreSQL, MariaDB
PostgreSQL	Amazon Aurora, MySQL, MariaDB
Amazon Aurora	PostgreSQL
Oracle Data Warehouse	Amazon Redshift
Teradata	Amazon Redshift
Netezza	Amazon Redshift
Greenplum	Amazon Redshift

https://aws.amazon.com/dms/

AWS SUMMIT

Thank you!

amazon web services