amazon web services

Amazon DynamoDB and Amazon RDS Deep Dive

Sundar Raghavan

raghavas@amazon

amazon web services

Provides real-time weather updates

Makes video gamers happy

Scales to millions of users

Delivers billions of page flips

How?

amazon web services

WeatherBug — Provides real-time weather updates

gumi — Makes video gamers happy

SAMSUNG — Scales to millions of users

Flipboard — Delivers billions of page flips

Amazon DynamoDB

Amazon RDS

amazon web services

YesSQL?

NoSQL?

(Hint)
Focus on your App

amazon web services

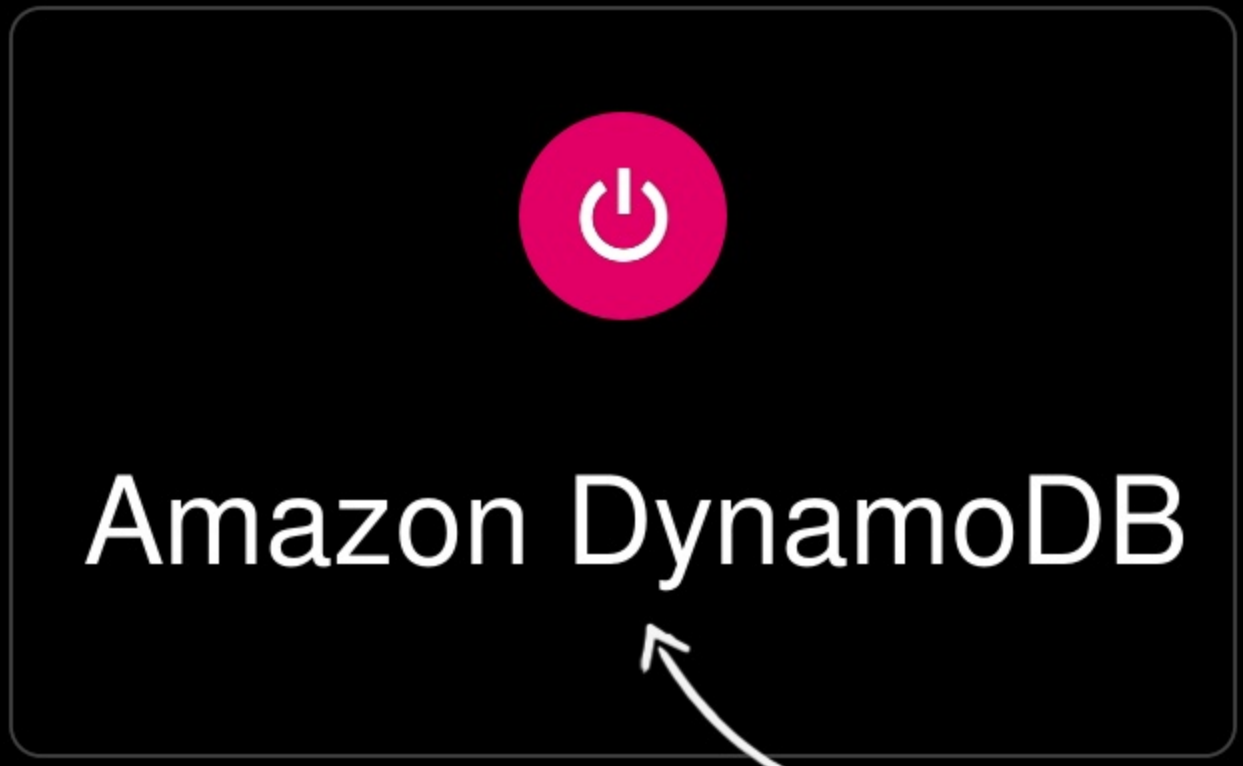
Factors	YesSQL (RDS)	NoSQL (DynamoDB)
Application	• App with complex business logic?	• Web app with lots of users?
Transactions	• Complex txns, joins, updates?	• Simple data model, updates, queries?
Scale	• Developer managed	• Automatic, on-demand scaling
Performance	• Developer architected	• Consistent, high performance at scale
Availability	• Architected for fail-over	• Seamless and transparent
Skills	• Relational + Web expertise	• Web expertise

Best of both worlds: Use SQL and NoSQL models in one app

amazon web services

Focus on your App with Amazon DynamoDB

amazon web services



Amazon DynamoDB

Quick overview

amazon web services

DynamoDB is a managed NoSQL database service.

Store and retrieve any amount of data.

Serve any level of request traffic.

Without any operational burden.

amazon web services

Consistent, predictable performance.

Single digit millisecond latencies.

Backed on solid-state drives.

amazon web services

Flexible data model

Key/attribute pairs.
No schema required.

Easy to create. Easy to adjust.

amazon web services

Seamless scalability

No table size limits. Unlimited storage.

No downtime.

amazon web services

Durable

Consistent, disk-only writes (not memory)

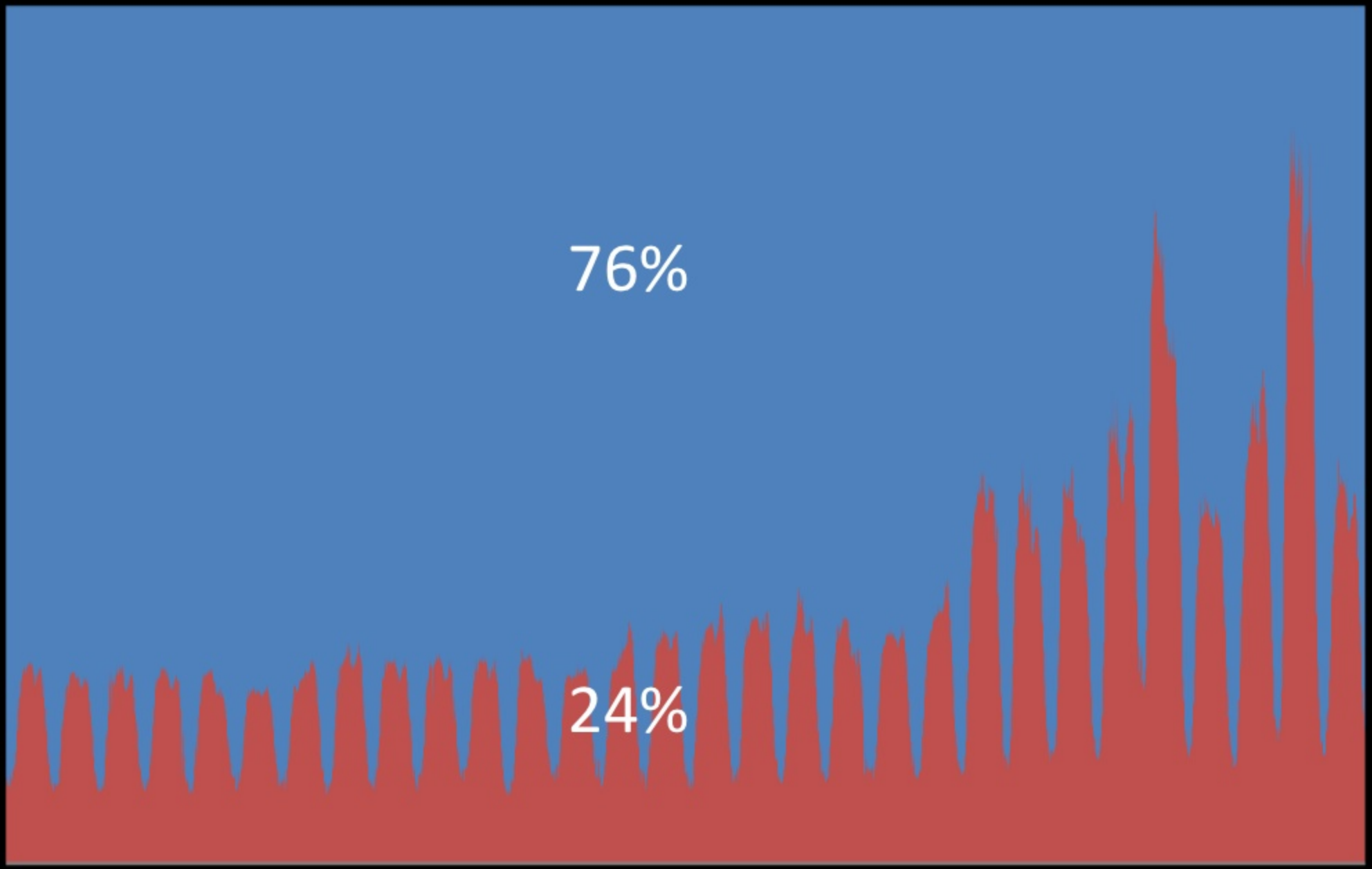
Replication across data centres and availability zones.

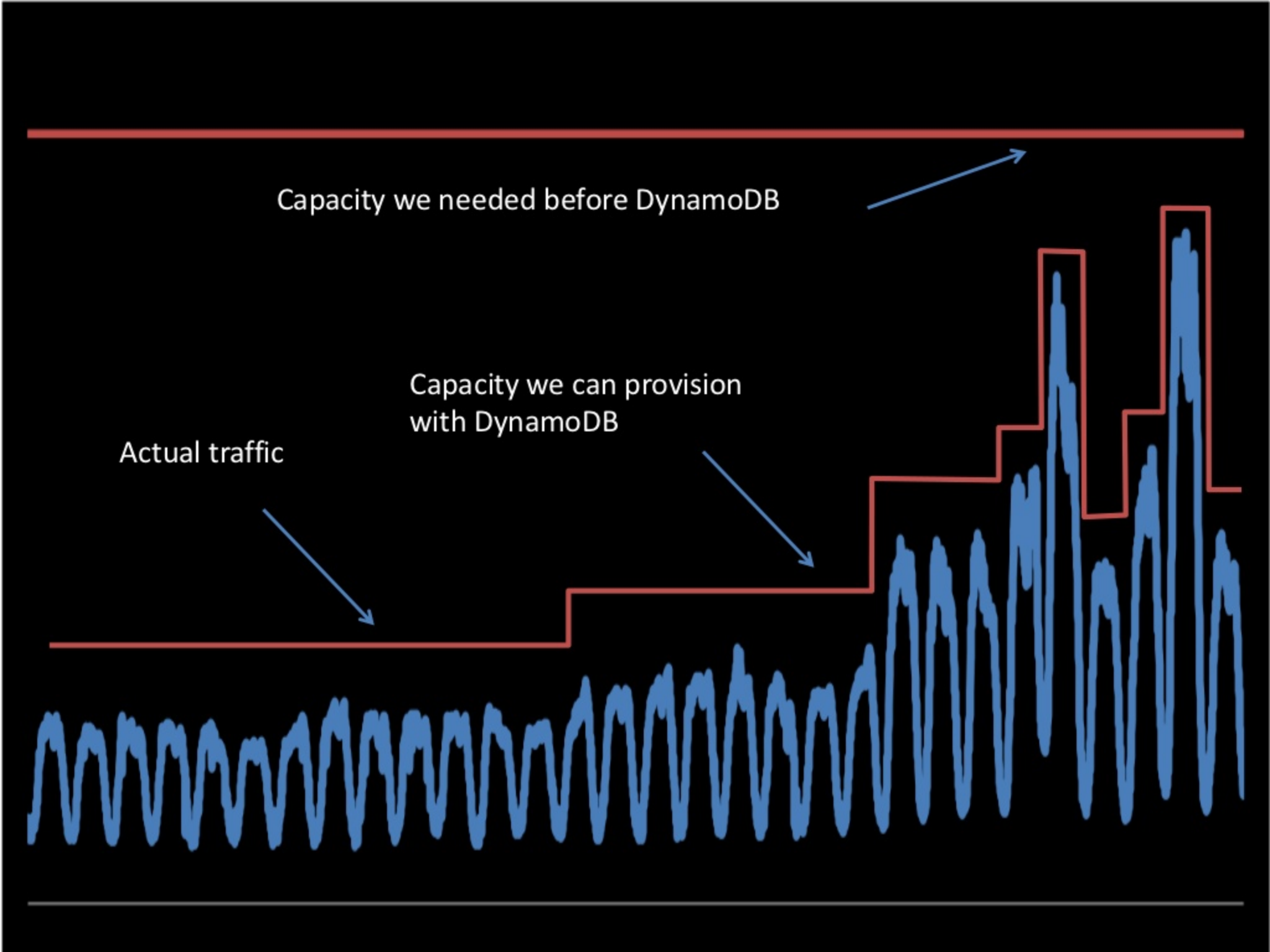
amazon web services

Without the operational burden.

FOCUS ON YOUR APP

November traffic at Amazon.com




Capacity we needed before DynamoDB
Capacity we can provision with DynamoDB
Actual traffic

amazon web services

Two decisions + three clicks = ready for use

amazon web services

Primary keys +
Provisioned level of throughput (easy to change)

Two decisions + three clicks = ready for use


amazon
web services
Order Table
(collection of items)
Attribute (key value pair)
String – unicode, utf8 binary
Number – 38 digit precision
Item (max size 64k, collection of attributes)
id = 100
date = 2012-05-16-09-00-10
total = 25.00
id = 101
date = 2012-05-16-09-00-11
total = 50.00
Hash Key
Range Key

amazon web services

Items are indexed by primary key

Single hash keys and composite keys.

amazon web services

Think of database operations as reads and writes

Item read = 1 read

Item update = 2 reads and 1 write

Item insert = 1 write

amazon web services

Provisioned throughput

Reserve IOPS for reads and writes.

Scale up (or down) at any time.

Priced per hour of provisioned throughput.

amazon web services

Provisioned throughput

Write throughput units = item size * writes/sec

$0.01 per hour for 10 write units

Read throughput units = item size * ½ * reads/sec

$0.01 per hour for 100 read units

amazon web services

Consistent writes

Atomic increment/decrement.

Optimistic concurrency control.

aka: “conditional writes”.

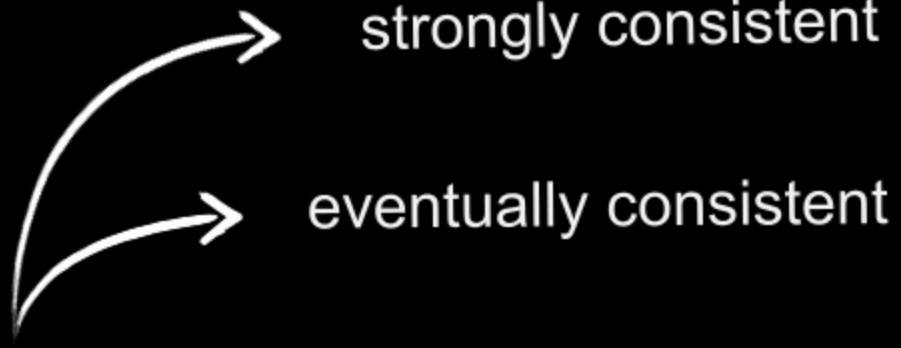
amazon web services

Transactions

Item level transactions only.

Puts, updates and deletes are ACID.


amazon
web services
strongly consistent
eventually consistent
Read throughput
Provisioned units =
size of item x reads/second
$0.01 per hour for 50 read units



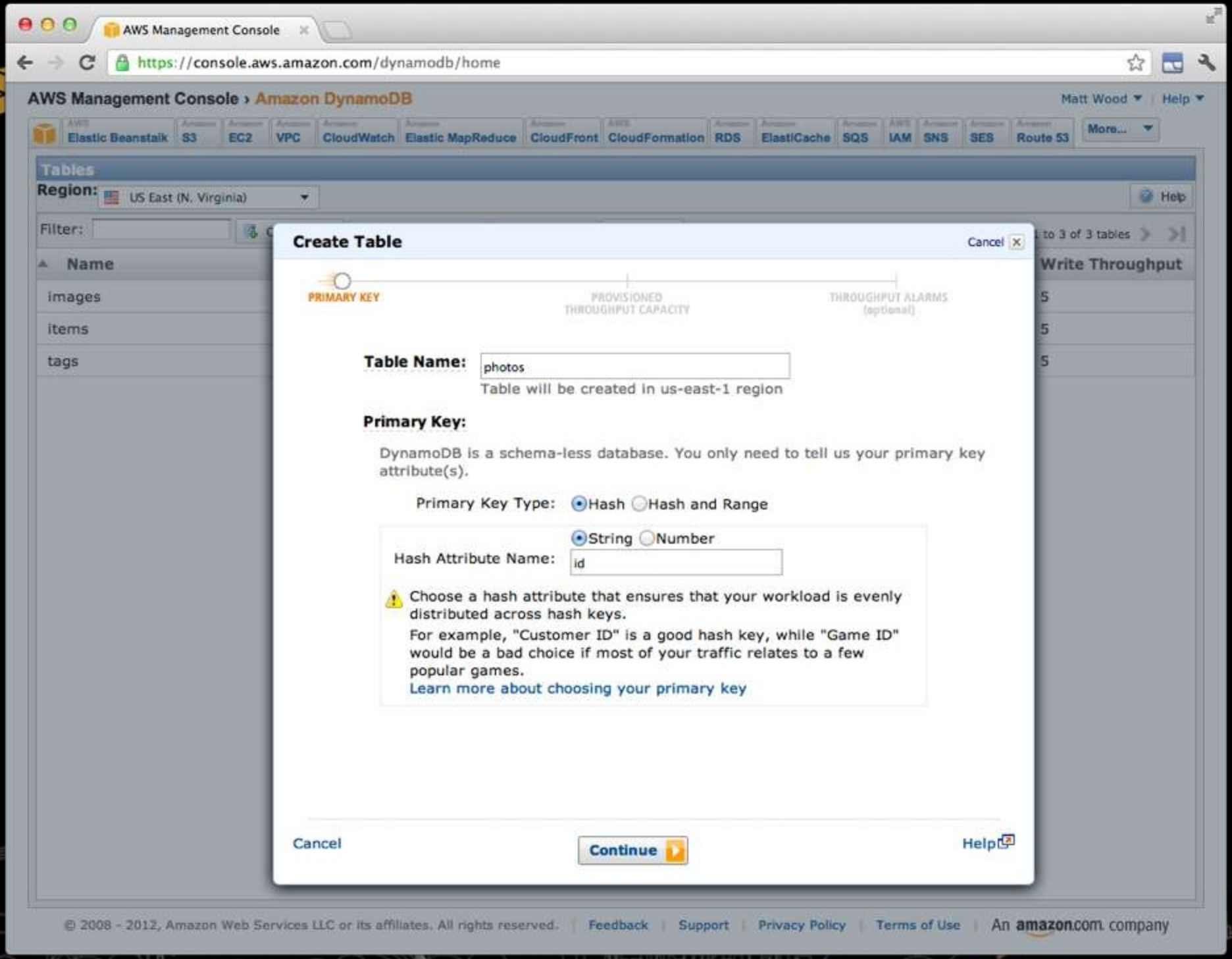
Read throughput

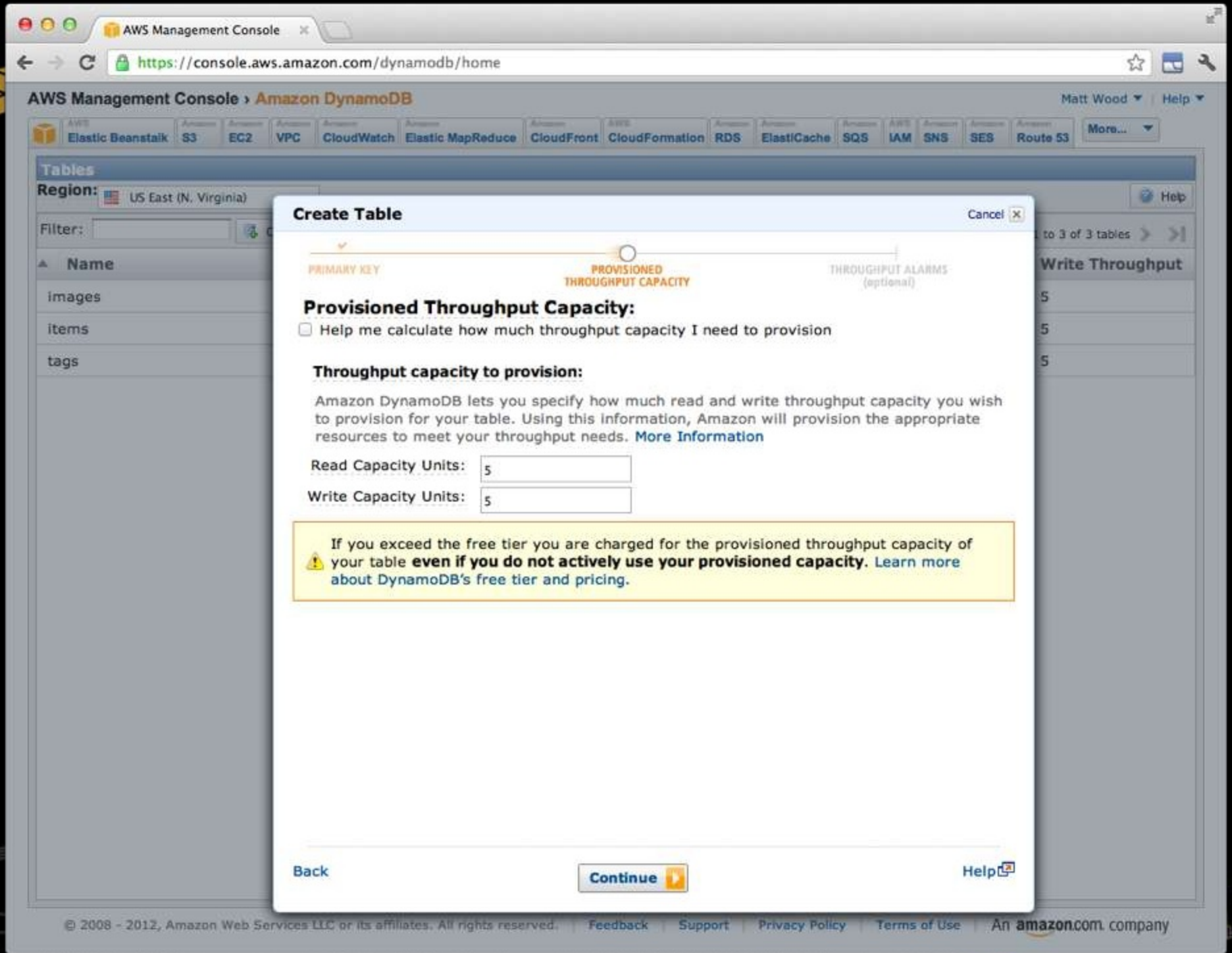
Same latency expectations.

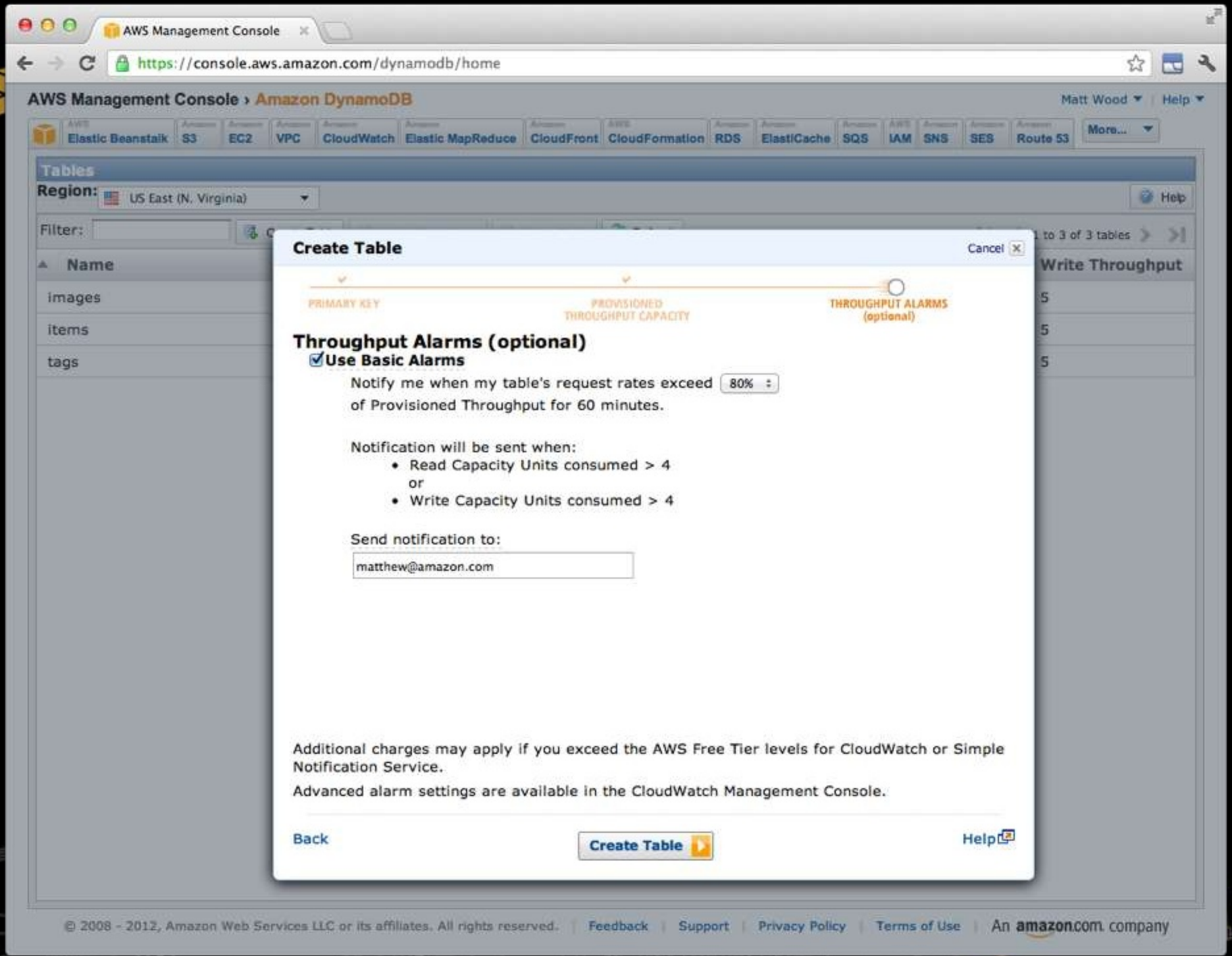
Mix and match at “read time”.

amazon web services

Two decisions + three clicks = ready for use


AWS Management Console
https://console.aws.amazon.com/dynamodb/home
AWS Management Console › Amazon DynamoDB
Matt Wood
Help
Elastic Beanstalk
S3
EC2
VPC
CloudWatch
Elastic MapReduce
CloudFront
CloudFormation
RDS
ElastiCache
SQS
IAM
SNS
SES
Route 53
More...
Tables
Region:
US East (N. Virginia)
Filter:
Name
images
items
tags
Write Throughput
5
5
5
Create Table
Cancel
PRIMARY KEY
PROVISIONED THROUGHPUT CAPACITY
THROUGHPUT ALARMS (optional)
Table Name:
photos
Table will be created in us-east-1 region
Primary Key:
DynamoDB is a schema-less database. You only need to tell us your primary key attribute(s).
Primary Key Type:
Hash
Hash and Range
String
Number
Hash Attribute Name:
id
Choose a hash attribute that ensures that your workload is evenly distributed across hash keys.
For example, "Customer ID" is a good hash key, while "Game ID" would be a bad choice if most of your traffic relates to a few popular games.
Learn more about choosing your primary key
Cancel
Continue
Help
© 2008 - 2012, Amazon Web Services LLC or its affiliates. All rights reserved.
Feedback
Support
Privacy Policy
Terms of Use
An amazon.com company


AWS Management Console
https://console.aws.amazon.com/dynamodb/home
AWS Management Console › Amazon DynamoDB
Matt Wood
Help
Elastic Beanstalk
S3
EC2
VPC
CloudWatch
Elastic MapReduce
CloudFront
CloudFormation
RDS
ElastiCache
SQS
IAM
SNS
SES
Route 53
More...
Tables
Region:
US East (N. Virginia)
Filter:
Name
images
items
tags
Write Throughput
5
5
5
Create Table
Cancel
PRIMARY KEY
PROVISIONED THROUGHPUT CAPACITY
THROUGHPUT ALARMS (optional)
Provisioned Throughput Capacity:
Help me calculate how much throughput capacity I need to provision
Throughput capacity to provision:
Amazon DynamoDB lets you specify how much read and write throughput capacity you wish to provision for your table. Using this information, Amazon will provision the appropriate resources to meet your throughput needs. More Information
Read Capacity Units:
5
Write Capacity Units:
5
If you exceed the free tier you are charged for the provisioned throughput capacity of your table even if you do not actively use your provisioned capacity. Learn more about DynamoDB's free tier and pricing.
Back
Continue
Help
© 2008 - 2012, Amazon Web Services LLC or its affiliates. All rights reserved.
Feedback
Support
Privacy Policy
Terms of Use
An amazon.com company


AWS Management Console
https://console.aws.amazon.com/dynamodb/home
AWS Management Console › Amazon DynamoDB
Matt Wood
Help
Elastic Beanstalk
S3
EC2
VPC
CloudWatch
Elastic MapReduce
CloudFront
CloudFormation
RDS
ElastiCache
SQS
IAM
SNS
SES
Route 53
More...
Tables
Region:
US East (N. Virginia)
Help
Filter:
to 3 of 3 tables
Name
Write Throughput
images
items
tags
5
5
5
Create Table
Cancel
PRIMARY KEY
PROVISIONED THROUGHPUT CAPACITY
THROUGHPUT ALARMS (optional)
Throughput Alarms (optional)
Use Basic Alarms
Notify me when my table's request rates exceed
80%
of Provisioned Throughput for 60 minutes.
Notification will be sent when:
Read Capacity Units consumed > 4
or
Write Capacity Units consumed > 4
Send notification to:
matthew@amazon.com
Additional charges may apply if you exceed the AWS Free Tier levels for CloudWatch or Simple Notification Service.
Advanced alarm settings are available in the CloudWatch Management Console.
Back
Create Table
Help
© 2008 - 2012, Amazon Web Services LLC or its affiliates. All rights reserved.
Feedback
Support
Privacy Policy
Terms of Use
An amazon.com company

amazon web services

Two decisions + three clicks = ready for use

amazon web services

Two decisions + one API call
= ready for use


```
$create_response = $dynamodb->create_table(array(
    'TableName' => 'ProductCatalog',
    'KeySchema' => array(
        'HashKeyElement' => array(
            'AttributeName' => 'Id',
            'AttributeType' => AmazonDynamoDB::TYPE_NUMBER
        )
    ),
    'ProvisionedThroughput' => array(
        'ReadCapacityUnits' => 10,
        'WriteCapacityUnits' => 5
    )
));
```

amazon web services

Two decisions + one API call
= ready for use

amazon web services

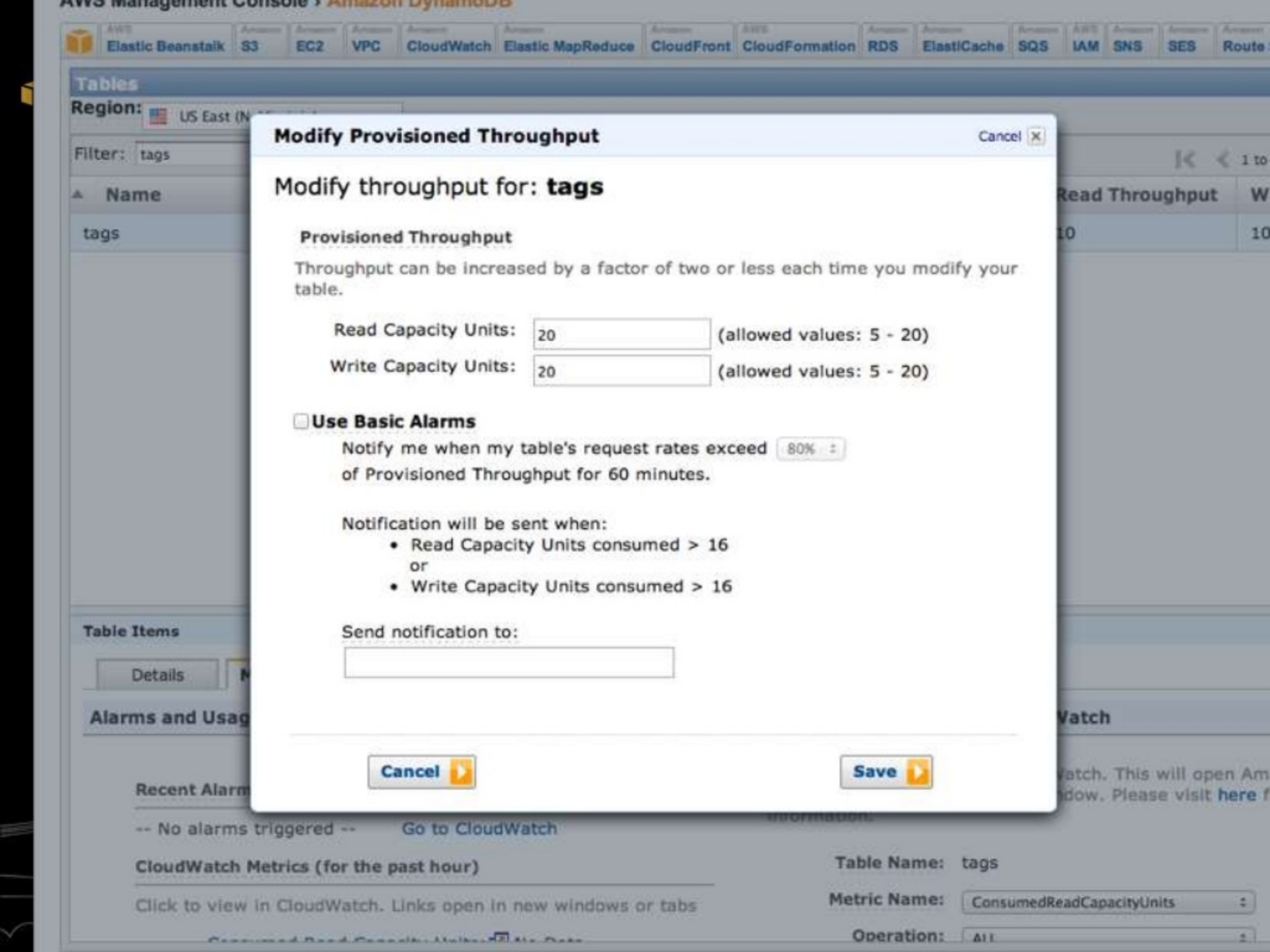
Two decisions + one API call
= ready for development

amazon web services

Two decisions + one API call
= ready for production

amazon web services

Two decisions + one API call
= ready for scale


Modify Provisioned Throughput
Cancel
Modify throughput for: tags
Provisioned Throughput
Throughput can be increased by a factor of two or less each time you modify your table.
Read Capacity Units: 20 (allowed values: 5 - 20)
Write Capacity Units: 20 (allowed values: 5 - 20)
Use Basic Alarms
Notify me when my table's request rates exceed 80%
of Provisioned Throughput for 60 minutes.
Notification will be sent when:
Read Capacity Units consumed > 16
or
Write Capacity Units consumed > 16
Send notification to:
Cancel
Save

amazon web services

Authentication

Session based to minimize latency.

Uses Amazon Security Token Service.

Handled by AWS SDKs.

Integrates with IAM.

amazon web services

Monitoring

CloudWatch metrics:
latency, consumed read and write throughput, errors and throttling.

amazon web services

Libraries, mappers & mocks

ColdFusion, Django, Erlang, Java, .Net, Node.js, Perl, PHP, Python, Ruby

http://j.mp/dynamodb-libs

amazon web services

(And I can't find my joins)

Ma, I lost my schema!

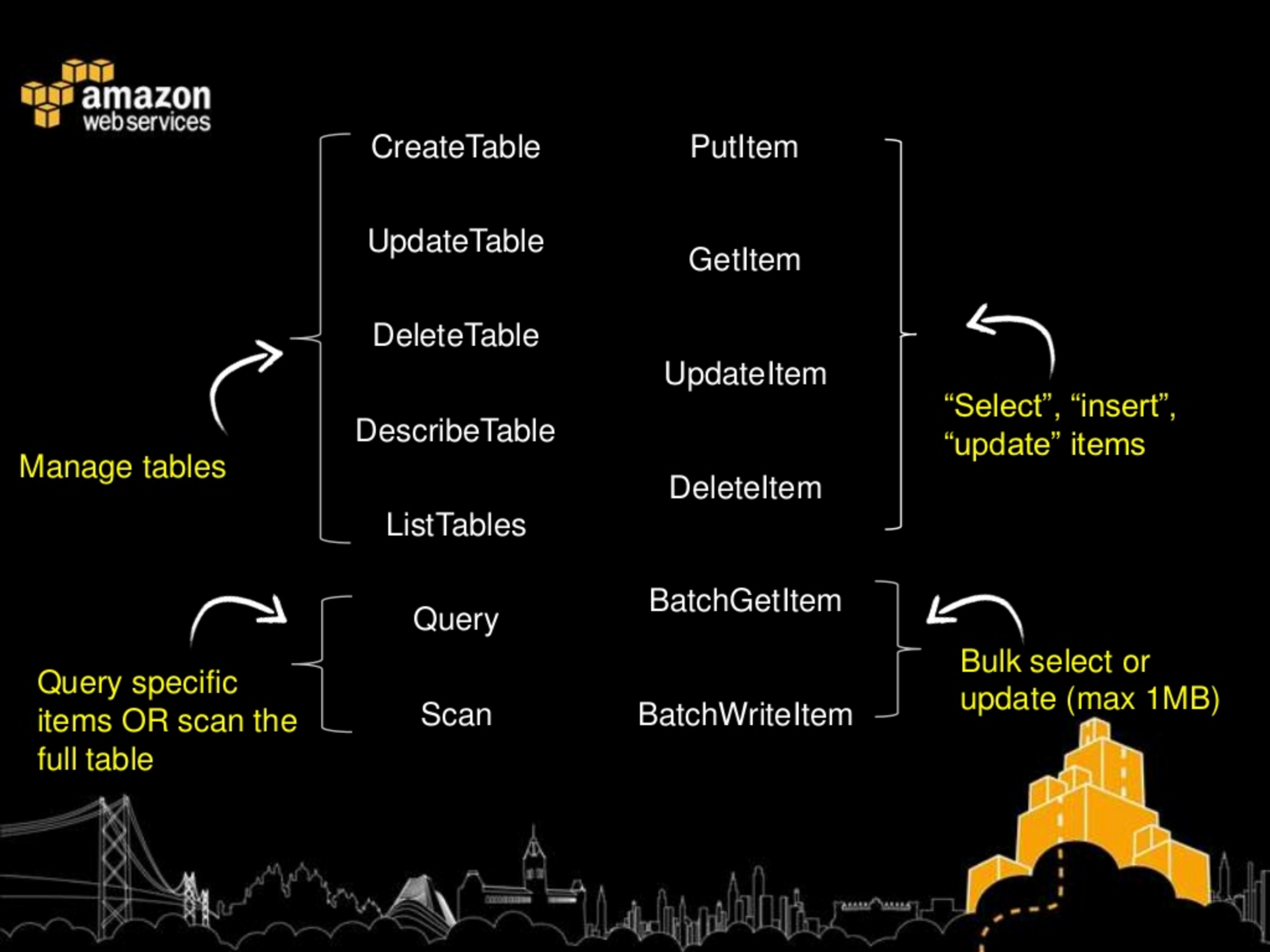
Tables do not require a formal schema.
Items are an arbitrary sized hash.
Just need to specify the primary key.

amazon web services

Programming DynamoDB

Small but perfectly formed.

Whole programming interface fits on one slide.


amazon web services
CreateTable
UpdateTable
DeleteTable
DescribeTable
ListTables
Manage tables
PutItem
GetItem
UpdateItem
DeleteItem
"Select", "insert", "update" items
Query
Scan
Query specific items OR scan the full table
BatchGetItem
BatchWriteItem
Bulk select or update (max 1MB)

amazon web services

Query patterns

Retrieve all items by hash key.

Range key conditions:
==, <, >, >=, <=, begins with, between.

Counts. Top and bottom n values.
Paged responses.

amazon web services

Modeling patterns

amazon web services

1. Mapping relationships with range keys.

No cross-table joins in DynamoDB.

Use composite keys to model relationships.

amazon web services

2. Handling large items.

Unlimited attributes per item.
Unlimited items per table.

Max 64k per item.

amazon web services

Store a pointer to objects in Amazon S3

Large data stored in S3.
Location stored in DynamoDB.
99.999999999% data durability in S3.

amazon web services

3. Managing secondary indices.

Not currently supported by DynamoDB.

You can create on your own.

amazon web services

4. Time series data.

Logging, click through, ad views, game play data, application usage.

Non-uniform access patterns.
Newer data is ‘live’.
Older data is read only.

amazon web services

Hot and cold tables.

Dec Jan Feb Mar April May

lower throughput

higher throughput

Amazon CloudDrive

- Stores user objects in Cloud
  - Metadata in DynamoDB
  - Objects in S3
- Queries and object searches are served by DynamoDB
  - What are all the objects in my drive?
  - Find me all the music albums stored on my drive

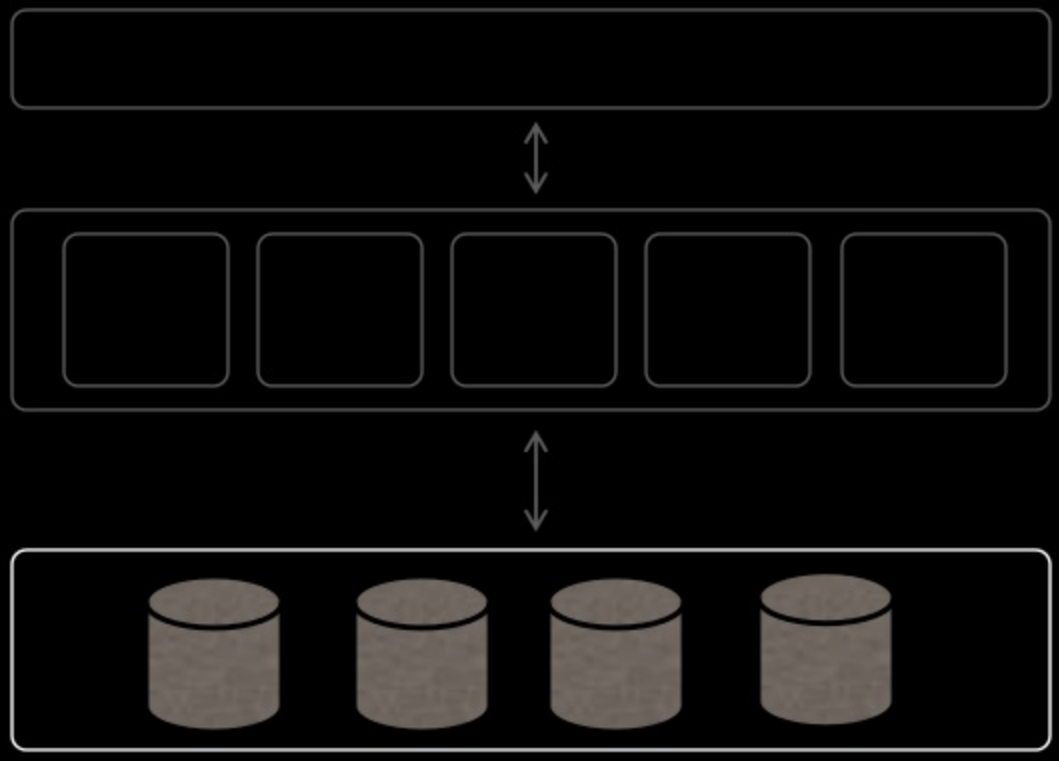
*"We were excited by how fast we were able to put DynamoDB into production and **how much developer time we saved**. In addition, DynamoDB **lets us scale up and down** easily by simply reserving increased throughput capacity when we need it and dialing it back when we don't"*

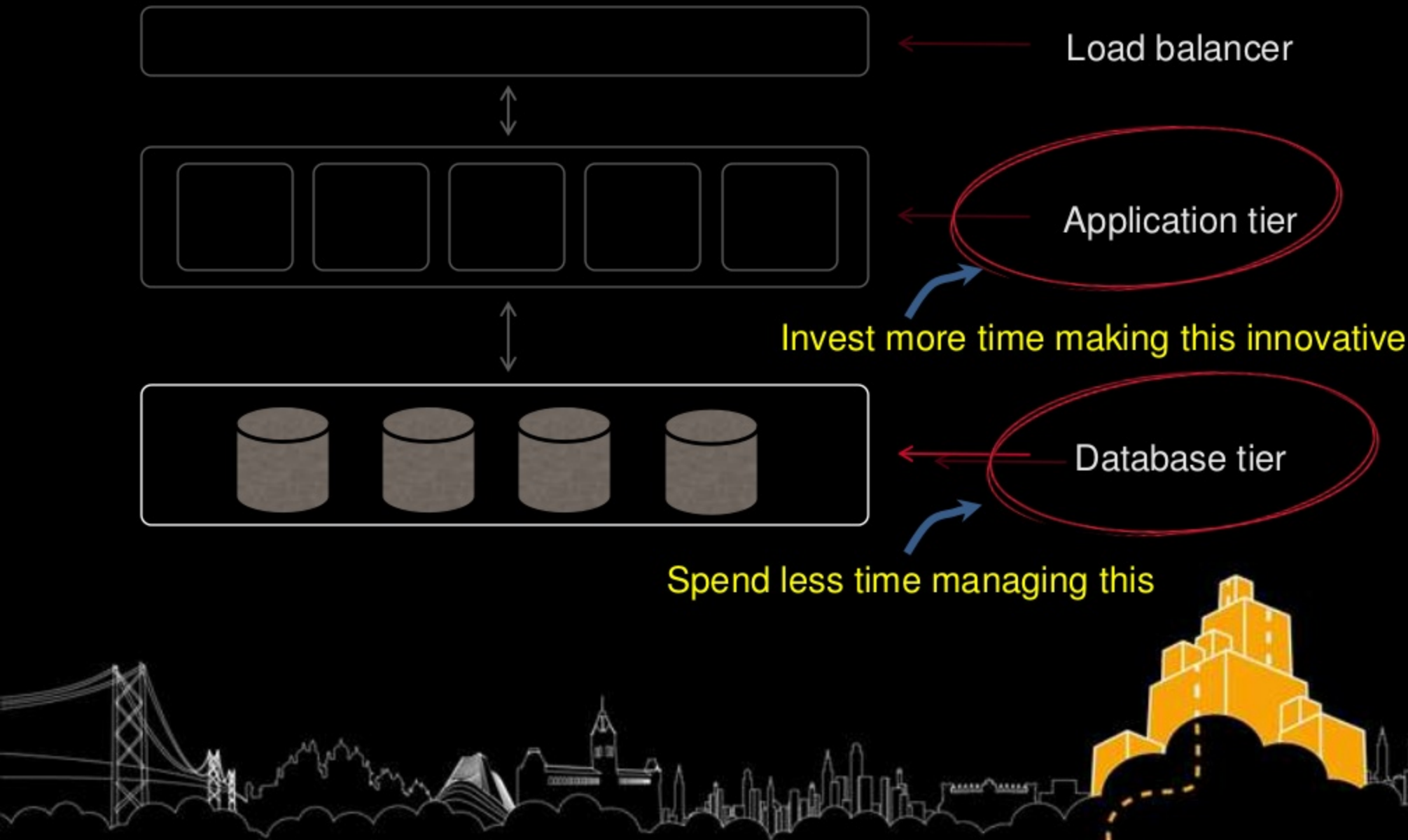
- Russel Dicker, Amazon CloudDrive

DynamoDB Customer Highlights

amazon web services

Focus on your App with Amazon RDS


Load balancer
Application tier
Database tier


Load balancer
Application tier
Invest more time making this innovative
Database tier
Spend less time managing this

amazon web services

Migration

Backup and recovery

Schema design

Patching

Query construction

Configuration

Software upgrades

Frequent server upgrades

Storage upgrades

Hardware crash

Query optimization

amazon web services

Schema design

Query construction

Query optimization

Focus on the “buck”

Migration

Backup and recovery

Patching

Configuration

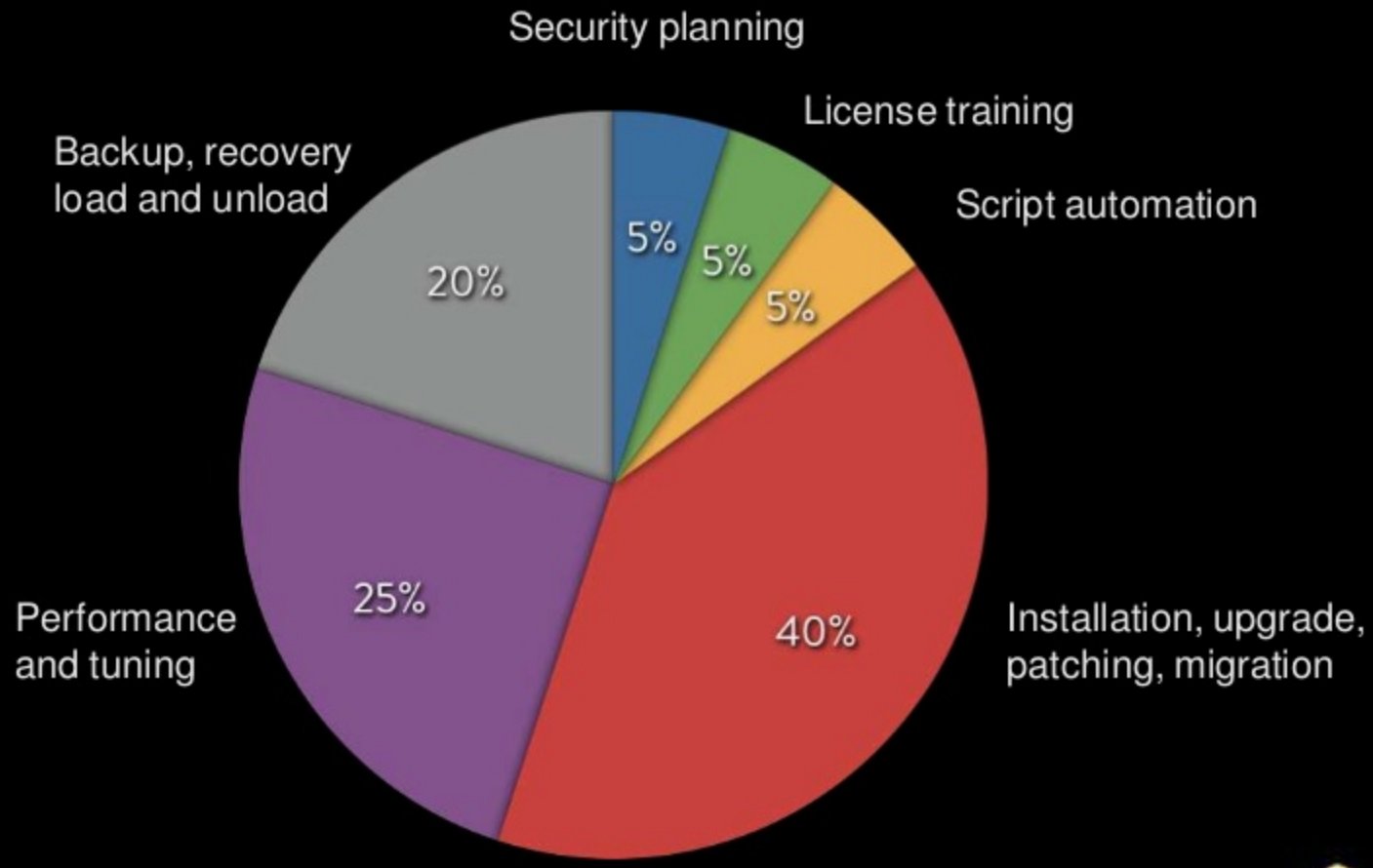
Software upgrades

Storage upgrades

Frequent server upgrades

Hardware crash

Off load the “muck”



Source: Forrester

amazon web services

Amazon RDS

Relational Database Service

Quick introduction

Amazon RDS Overview

Amazon RDS is a fully managed relational database service.

- Choice of Database engines
- Simple to deploy
- Easy to operate and scale
- Reliable
- Cost effective

amazon web services

Choice of database engines

- MySQL
- Oracle
- Microsoft SQL Server

amazon web services

Three decisions + a few clicks = ready for use

amazon web services

Database engine +
Instance size & storage +
Availability

Easy to change

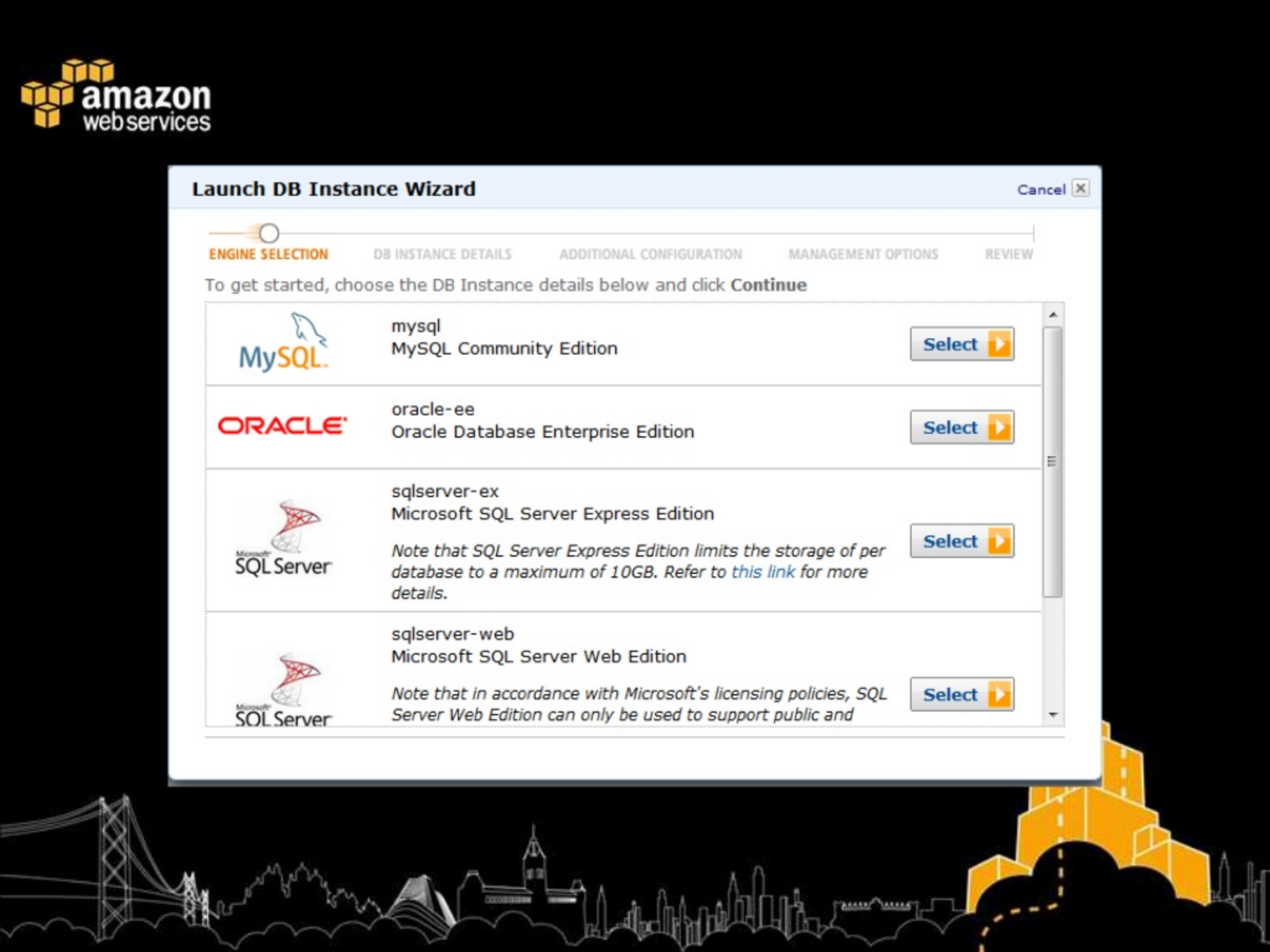
Three decisions + a few clicks = ready for use

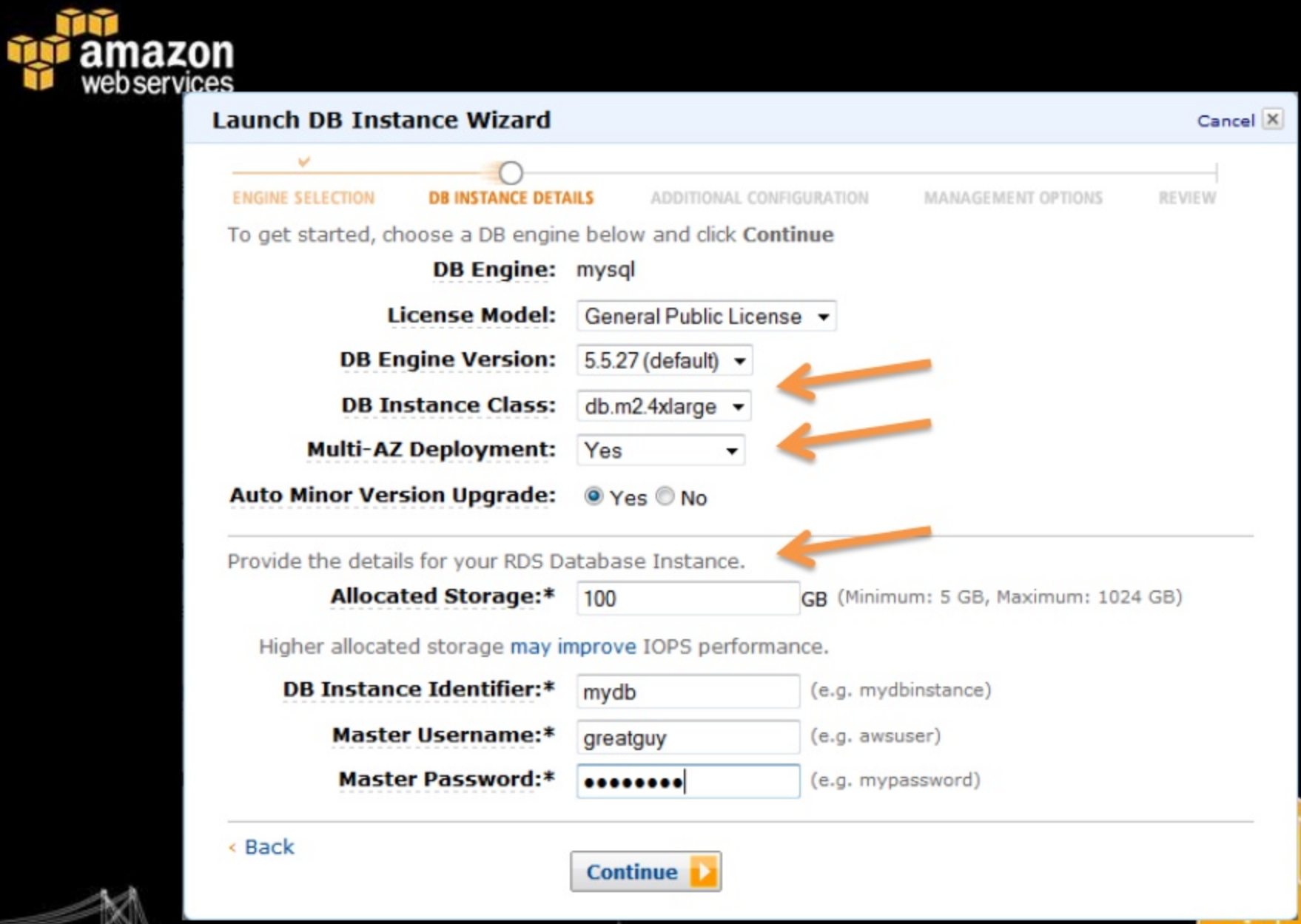
amazon web services

Database engine +
Instance size & storage +
Availability

Easy to change

Three decisions + a few clicks = ready for use


amazon
web services
Launch DB Instance Wizard
Cancel
ENGINE SELECTION
DB INSTANCE DETAILS
ADDITIONAL CONFIGURATION
MANAGEMENT OPTIONS
REVIEW
To get started, choose the DB Instance details below and click Continue
MySQL
mysql
MySQL Community Edition
Select
ORACLE
oracle-ee
Oracle Database Enterprise Edition
Select
Microsoft SQL Server
sqlserver-ex
Microsoft SQL Server Express Edition
Note that SQL Server Express Edition limits the storage of per database to a maximum of 10GB. Refer to this link for more details.
Select
Microsoft SQL Server
sqlserver-web
Microsoft SQL Server Web Edition
Note that in accordance with Microsoft's licensing policies, SQL Server Web Edition can only be used to support public and
Select


Launch DB Instance Wizard
Cancel
ENGINE SELECTION
DB INSTANCE DETAILS
ADDITIONAL CONFIGURATION
MANAGEMENT OPTIONS
REVIEW
To get started, choose a DB engine below and click Continue
DB Engine: mysql
License Model: General Public License
DB Engine Version: 5.5.27 (default)
DB Instance Class: db.m2.4xlarge
Multi-AZ Deployment: Yes
Auto Minor Version Upgrade: Yes No
Provide the details for your RDS Database Instance.
Allocated Storage:* 100 GB (Minimum: 5 GB, Maximum: 1024 GB)
Higher allocated storage may improve IOPS performance.
DB Instance Identifier:* mydb (e.g. mydbinstance)
Master Username:* greatguy (e.g. awsuser)
Master Password:* ••••••••• (e.g. mypassword)
Back
Continue

Launch DB Instance Wizard

Cancel

ENGINE SELECTION ✓ | DB INSTANCE DETAILS ✓ | ADDITIONAL CONFIGURATION ✓ | MANAGEMENT OPTIONS ✓ | REVIEW

Please review the information below, then click **Launch DB Instance**.

Engine:	mysql
Engine Version:	5.5.27
License Model:	general-public-license
Auto Minor Ver. Upgrade:	Yes
DB Instance Class:	db.m2.4xlarge
Multi-AZ Deployment:	Yes
Allocated Storage:	100
DB Instance Identifier:	mydb
Master User Name:	greatguy
Master User Password:	********
Database Name:	aaa
Database Port:	3306
Availability Zone:	Using a Multi-AZ Deployment disables this preference.
Option Group:	default:mysql-5-5
DB Parameter Group:	default.mysql5.5
DB Security Group(s):	default
DB Subnet Group:	
Backup Retention Period:	1
Backup Window:	No Preference
Maintenance Window:	No Preference

‹ Back

Launch DB Instance

amazon web services

Three decisions + a few clicks
= ready for use

amazon web services

Three decisions + a few clicks
= ready for development

amazon web services

Three decisions + a few clicks
= ready for scale

amazon web services

Three decisions + a few clicks
= ready for production

amazon web services

Three ways to remove the “muck”

amazon web services

Productivity

Performance

Data Protection

Productivity

Most admin tasks are managed by RDS

- Provisioning
- Database backups
- Patching
- Performance management

Productivity

Database snapshots

- Automatic
- Point in time recovery
- Faster time to recover

amazon web services

Productivity

- ✓ Multiple databases per instance
- ✓ Standard user accounts
- ✓ Connect and query using common MySQL tools & drivers
- ✓ Tune engine parameters
- ✓ Import and export data using standard MySQL tools (mysqldump)
- ✓ Diagnostics
- ✓ Native MySQL replication
- ✓ SSL for encryption over the wire
- ✓ Monitor metrics

- ✕ Shell, super user or direct file system access (**Think security**!)

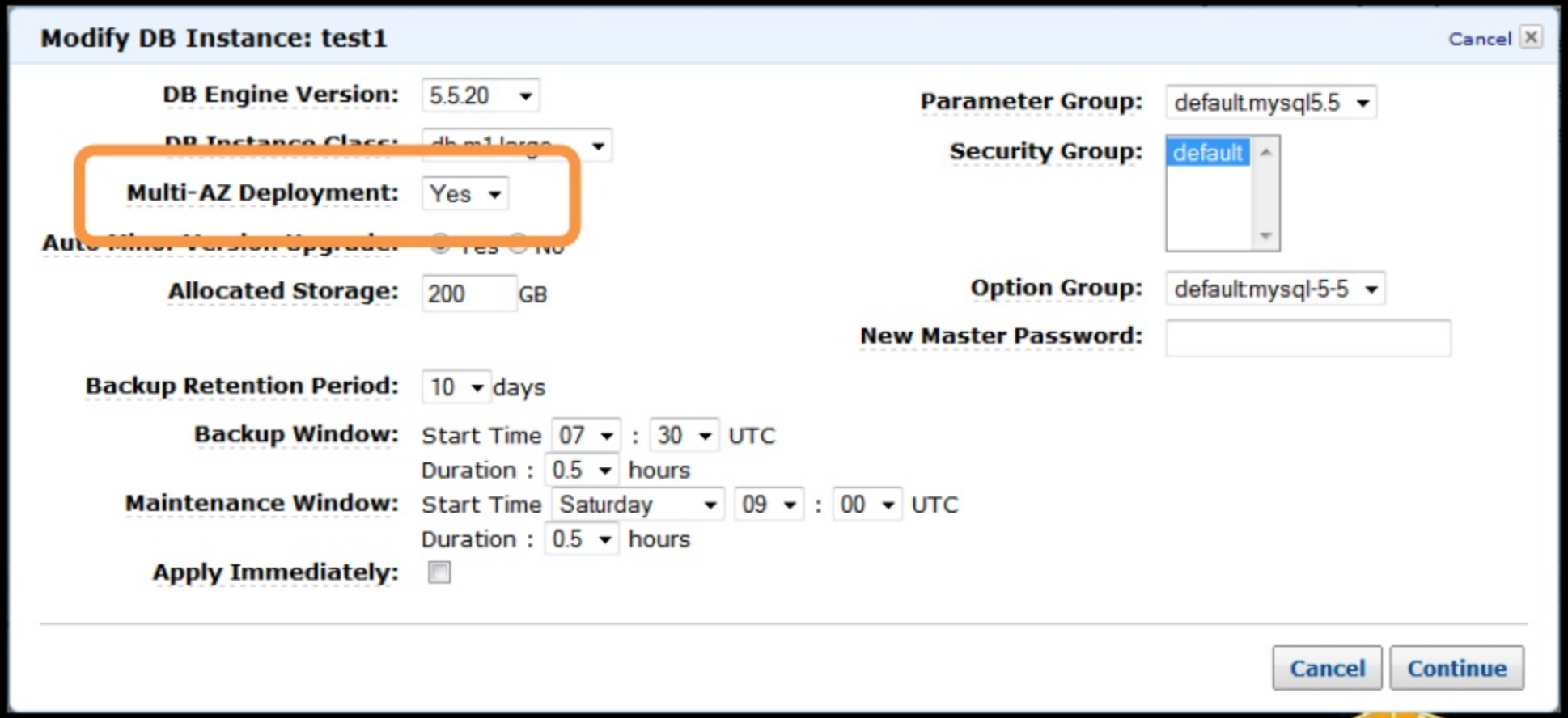
Productivity

Near zero administration

Painless patching, automatic upgrades

Cloudwatch monitoring, Metric alarms

One click. High Availability.


Modify DB Instance: test1
Cancel
DB Engine Version:
5.5.20
Parameter Group:
default.mysql5.5
Security Group:
default
Multi-AZ Deployment:
Yes
Allocated Storage:
200
GB
Option Group:
default:mysql-5-5
New Master Password:
Backup Retention Period:
10
days
Backup Window:
Start Time
07
30
UTC
Duration :
0.5
hours
Maintenance Window:
Start Time
Saturday
09
00
UTC
Duration :
0.5
hours
Apply Immediately:
Cancel
Continue

Productivity

One click. High availability with Multi-AZ

Automated deployment across multiple AZs

Synchronous replication from master to replica

Automatic fail-over; replica promoted to master

Test fail-over

Performance

Push-button scale, high performance

- Scale storage from 5Gb to 1Tb of storage
- Scale instance from small to 4XL (better I/O)
- Add read replicas with asynchronous replication
- Add **ElastiCache** for performance

Performance

Amazon ElastiCache

- In-memory cache service
- Ideal front end to RDS for read-heavy applications
- Low latency

Data Protection

Painless data protection

Daily database back-ups (save up to 35 days)

Snapshot any time. Point in time restore-use innodb

Recover everything up to final 5 minutes-innodb!

VPC, SSL, DB Security groups

Samsung saved $34M

Problem:

- Needed to reduce IT costs and were looking to create a more flexible IT environment

Solution:

- AWS and RDS services. With every request, the application authenticates devices, delivers apps and content, and pushes notifications.

Business Benefits:

- Saved $34M in hardware and maintenance expenses, 85% less than running on-premises

amazon web services

YesSQL

NoSQL

Focus on your App

Amazon RDS Customer Highlights

SEGA

airbnb
Travel like a human.

Tapjoy

The Washington Post Company

gumi

TweetDeck by Twitter

Flipboard

Newsweek & THE DAILY BEAST

amazon web services

Thank you

Free tier or
Free trial

aws.amazon.com/dynamodb

aws.amazon.com/rds

raghavas@amazon