AWS re:Invent

ANT240-S

Build an in-place Amazon S3 data lake with MongoDB Atlas

Sahir Azam
Senior VP, Cloud Products & GTM
MongoDB

Wisdom Omuya
Engineering Lead
MongoDB

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Agenda



Amazon S3 for data lakes



AWS services



Atlas Data Lake architecture

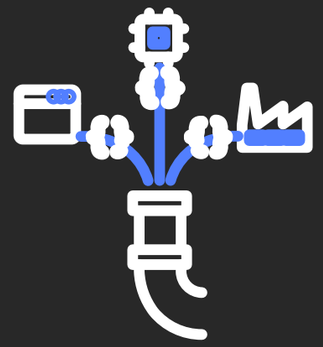


Building an Atlas Data Lake

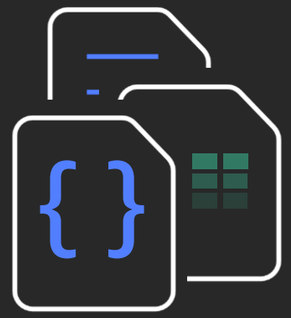
Modern business data is complex



Humongous amounts



Tons of sources



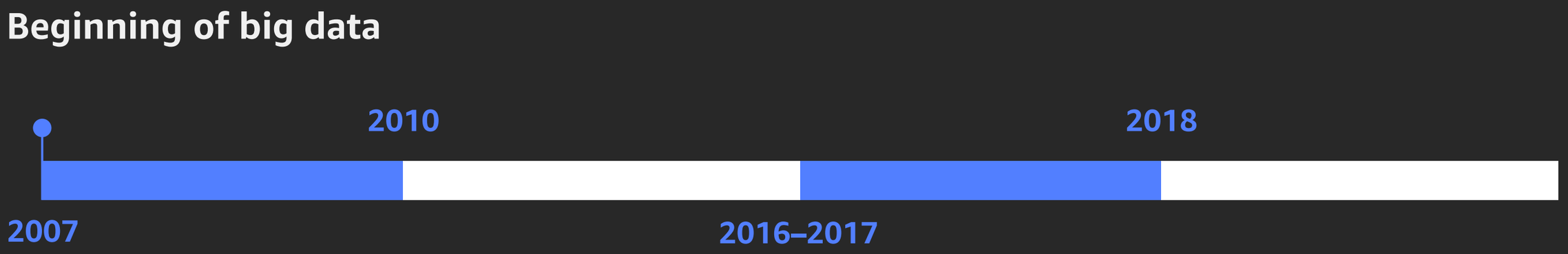
Lots of formats

The cost of storing data keeps dropping

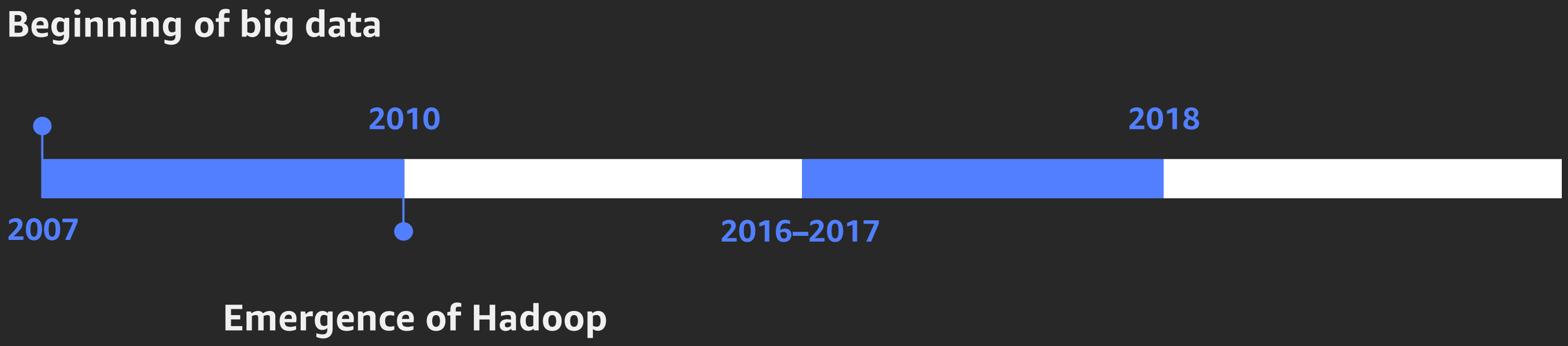
If you spent $1,000,000 to store data about 300 million people (3 copies each), how much could you store per person across time?

1980	$193,000/GB	6 bytes / person

Data lakes emerged to wrangle big data



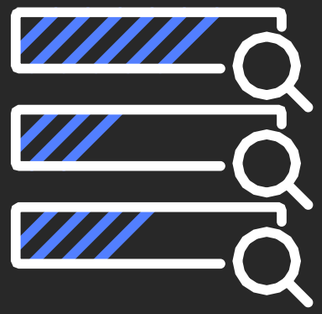
Data lakes emerged to wrangle big data



And the promise was great …



Cheap storage



Parallelizable queries

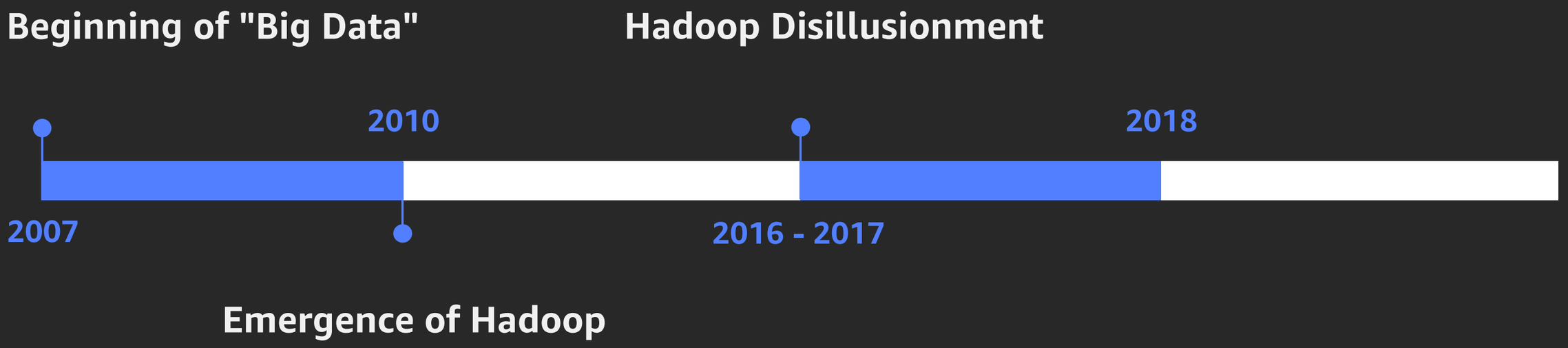


Easy to ingest



Easy to query

But the vision wasn't delivered….



Hadoop challenges



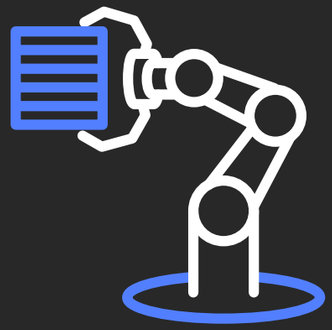
Only the largest enterprises can afford it



Wrong architecture for the public cloud

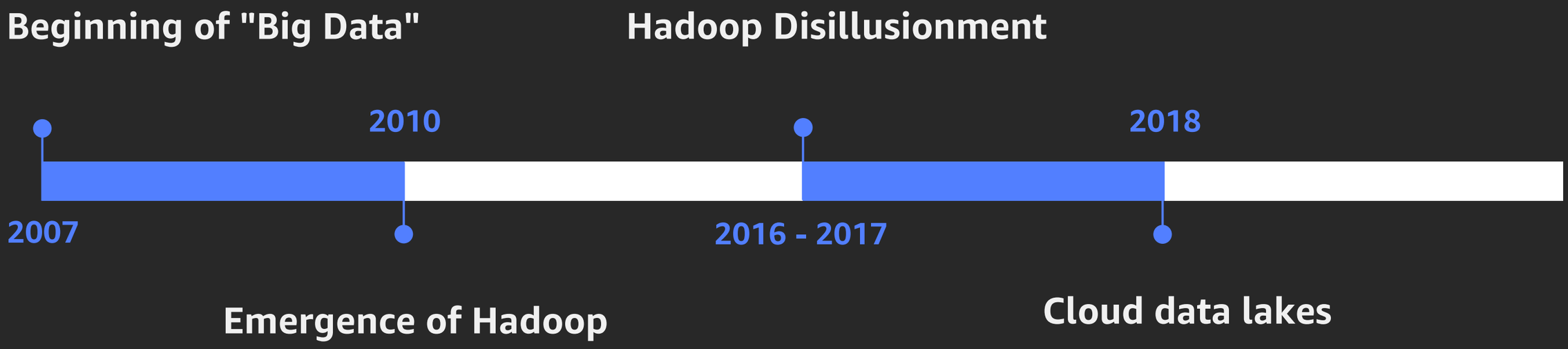


Complicated ecosystem of tools



Batch-oriented

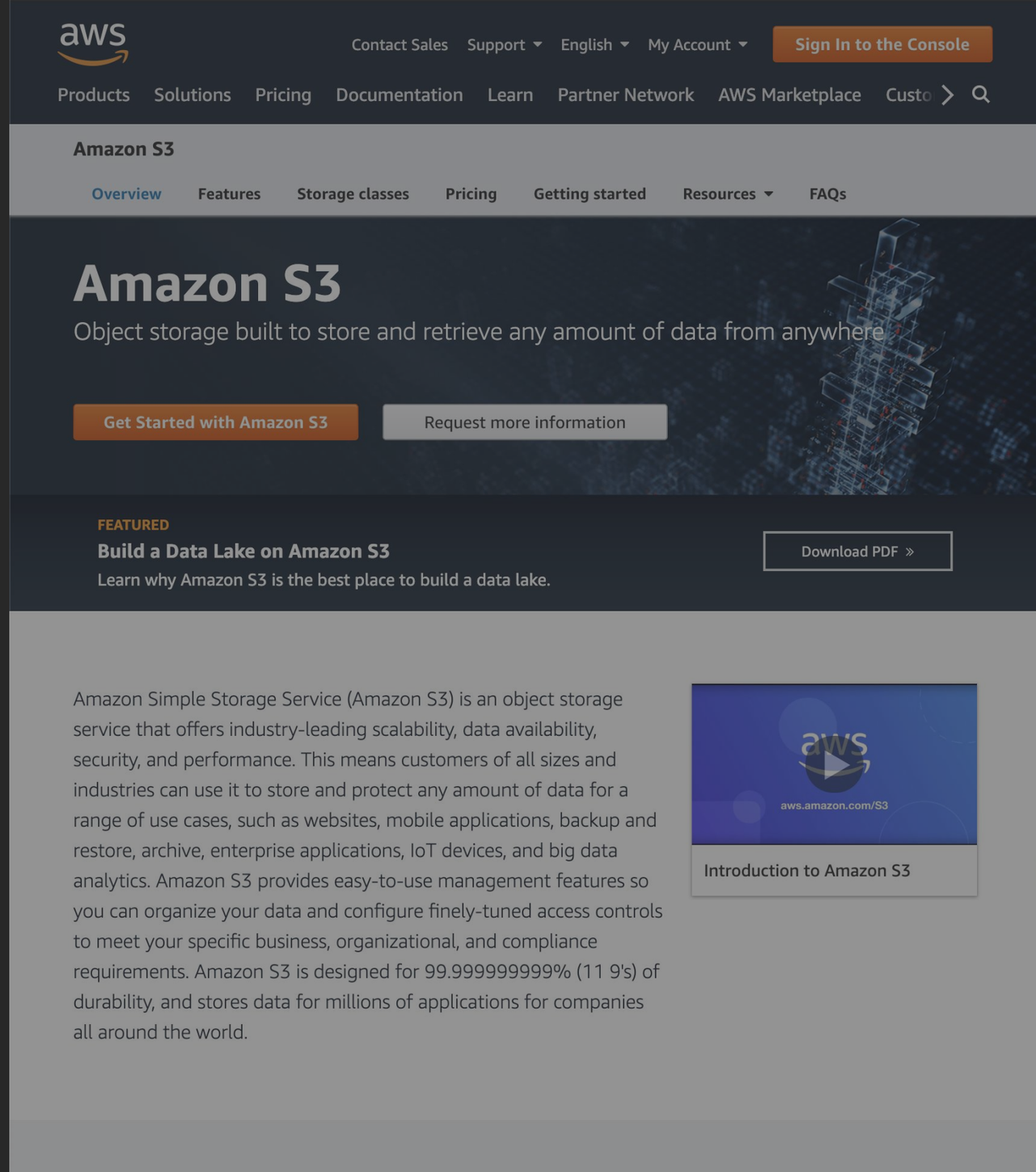
Cloud data lakes emerge to deliver the new data lake



Amazon Simple Storage Service (Amazon S3)

Amazon S3 is an object storage service that offers industry-leading scalability, data availability, security, and performance

docs.aws.amazon.com/AmazonS3/latest/gsg/GetStartedWithS3



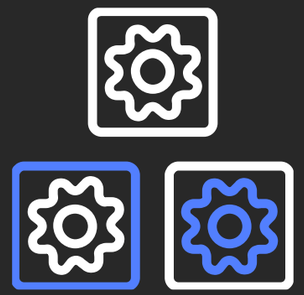
Amazon S3 is the best place to build data lakes



Unmatched durability, availability, and scalability



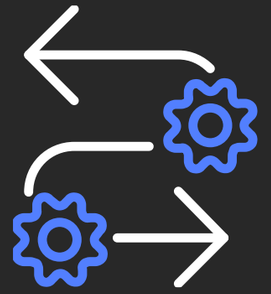
Best security, compliance, and audit capabilities



Object-level controls

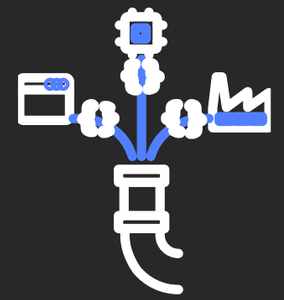


Business insights into your data



Most ways to bring data in

The need for many formats …

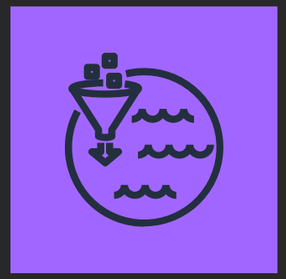


- There's no such thing as the "best" data format
- CSV, TSV, JSON are easy but not as efficient (compress/archive them!)
- Columnar compressed have a smaller storage footprint
- Row-oriented (AVRO) good for full data scans
- Organize into partitions
- Coalescing to larger partitions over time

Key considerations are cost, performance, and support

AWS has built strong services to tackle the data lake

Getting data in with



AWS Lake Formation



AWS Glue



Amazon Kinesis

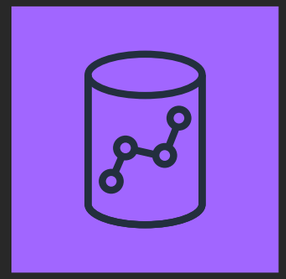
Getting data out with



Amazon Athena



Amazon S3 Select

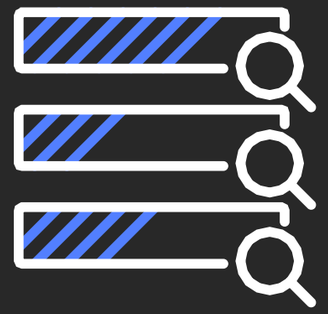


Amazon RedShift Spectrum

Characteristics of good data lakes



Cheap storage



Parallelizable queries



Easy to ingest



Easy to query

Making data easy is what MongoDB is all about (since 2008)

Making data easy is what MongoDB is all about



MongoDB Query Language

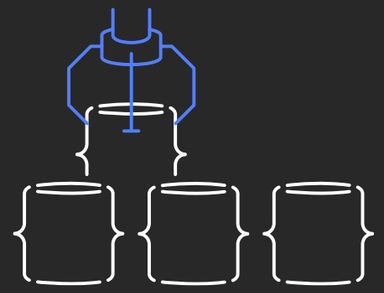
Expressive and natural for developers

Simple for complex, varied data

Emulated by many technologies

Worldwide adoption

MongoDB Atlas – Global cloud database





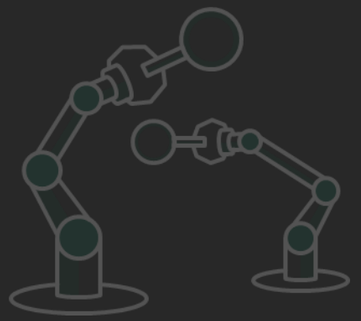
Self-service & elastic



Global



Enterprise-grade security & SLAs



Comprehensive monitoring

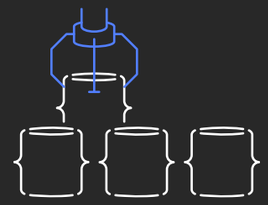


Managed backup



Stitch: Serverless platform services

MongoDB Atlas – Global cloud database





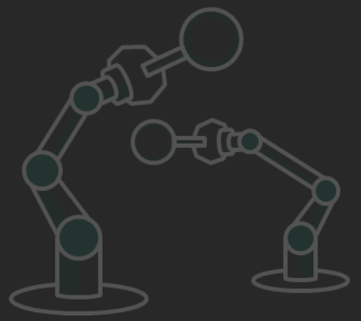
Self-service & elastic



Global



Enterprise-grade security & SLAs



Comprehensive monitoring

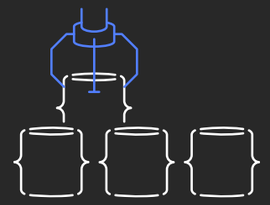


Managed backup



Stitch: Serverless platform services

MongoDB Atlas – Global cloud database





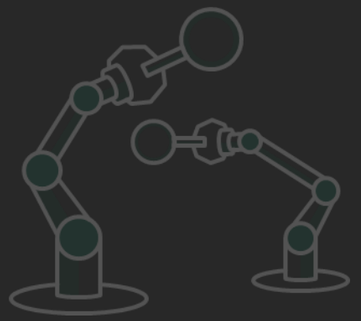
Self-service & elastic



Global



Enterprise-grade security & SLAs



Comprehensive monitoring

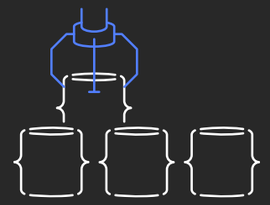


Managed backup



Stitch: Serverless platform services

MongoDB Atlas – Global cloud database





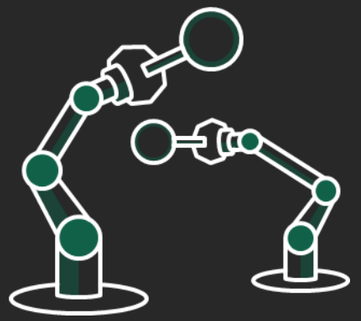
Self-service & elastic



Global



Enterprise-grade security & SLAs



Comprehensive monitoring

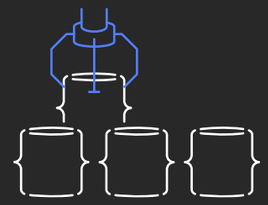


Managed backup



Stitch: Serverless platform services

MongoDB Atlas – Global cloud database





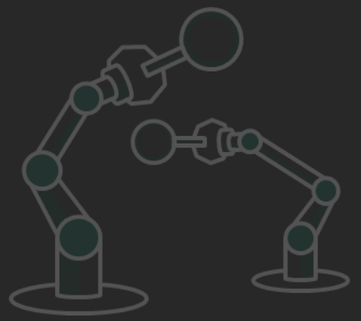
Self-service & elastic



Global



Enterprise-grade security & SLAs



Comprehensive monitoring

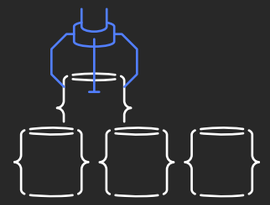


Managed backup



Stitch: Serverless platform services

MongoDB Atlas – Global cloud database





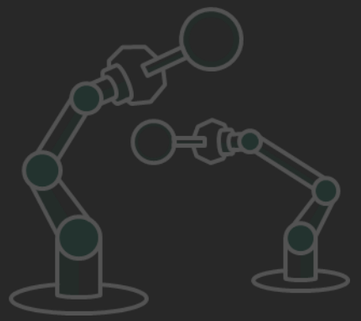
Self-service & elastic



Global



Enterprise-grade security & SLAs



Comprehensive monitoring

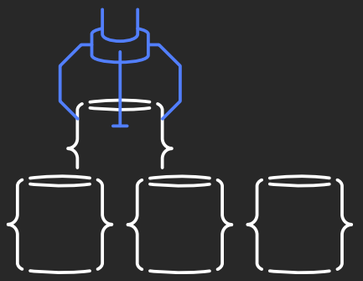


Managed backup



Stitch: Serverless platform services

MongoDB Atlas – Global cloud database





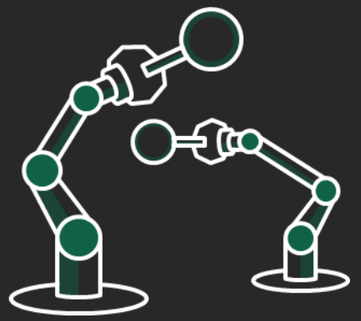
Self-service & elastic



Global



Enterprise-grade security & SLAs



Comprehensive monitoring



Managed backup



Stitch: Serverless platform services

Atlas usage statistics

	Total Atlas	AWS only
Number of read and write operations in a month	4,432,729,003,122	3,581,526,872,106

Numbers from October 2019



Atlas Data Lake



Serverless/pay for usage

Query Amazon S3 data in place

JSON, BSON, CSV, TSV, Avro, and Parquet formats

Supports compression

Standard MQL and MongoDB API

Compatible with all tools

Build an in-place S3 Data Lake with MongoDB Atlas

Wisdom Omuya

Engineering Lead

MongoDB

AWS re:Invent

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

aws

Agenda



Amazon S3 for data lakes

AWS services

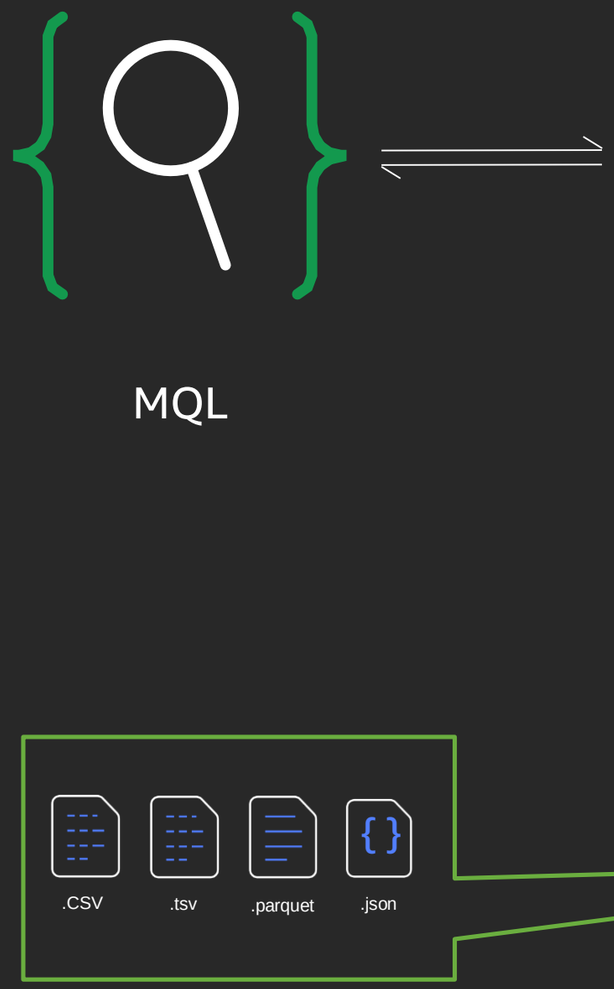


Atlas Data Lake architecture



Building an Atlas Data Lake

Atlas Data Lake architecture



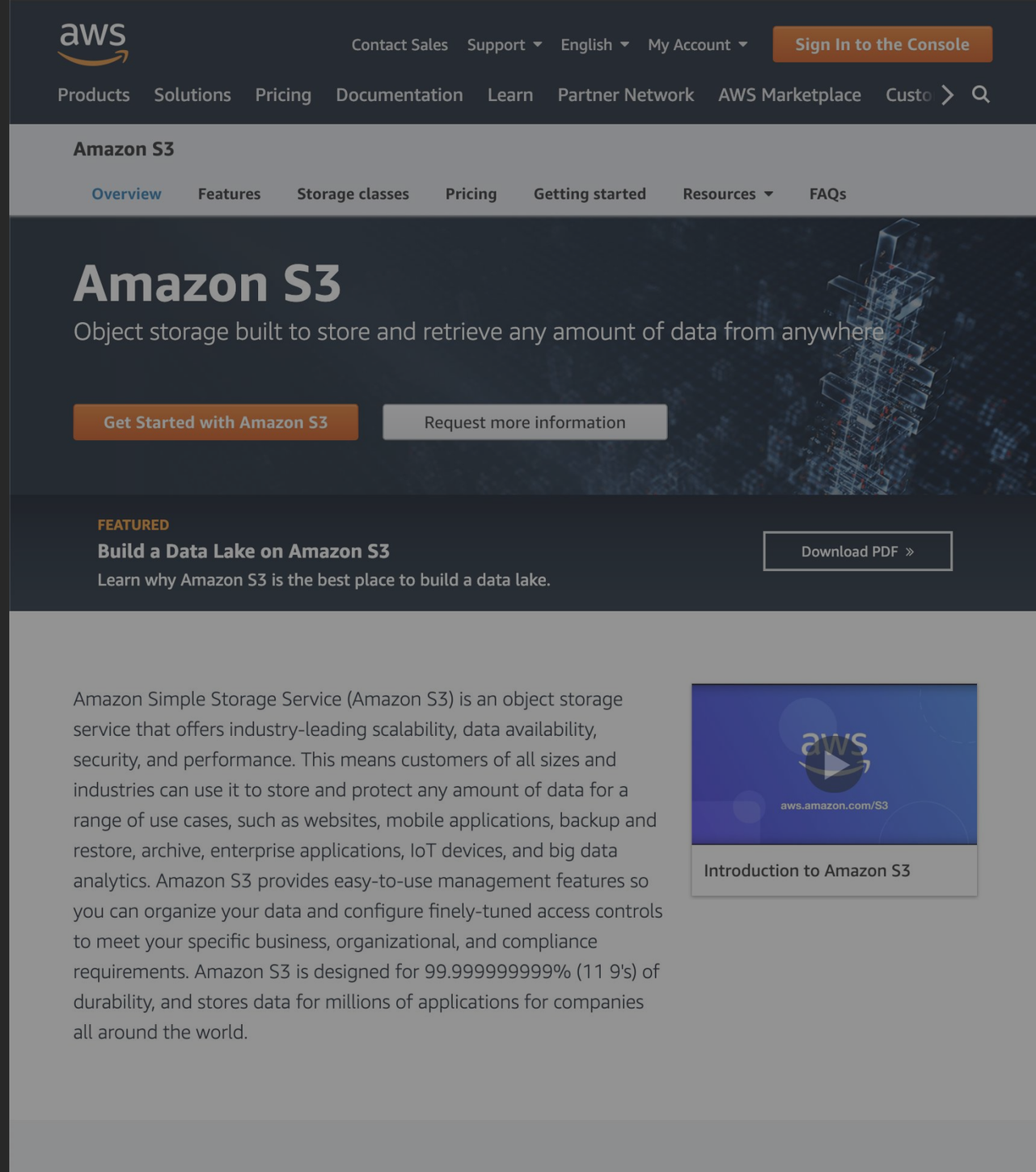
Amazon S3

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Amazon S3

Amazon S3 is an object storage service that offers industry-leading scalability, data availability, security, and performance

docs.aws.amazon.com/AmazonS3/latest/gsg/GetStartedWithS3

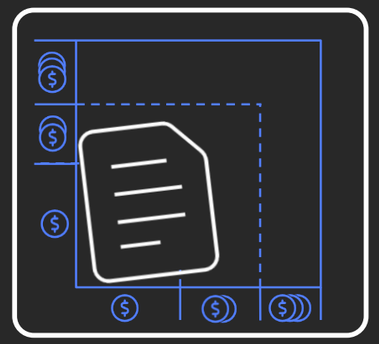




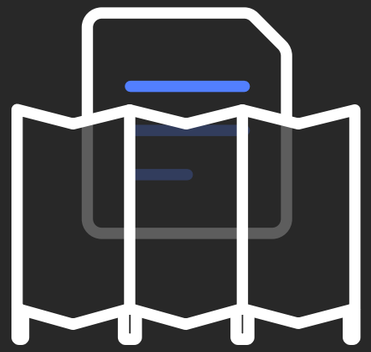
Data organization

Data organization



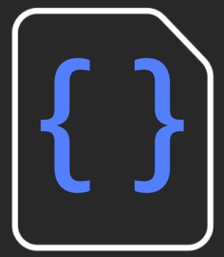


Storage classes

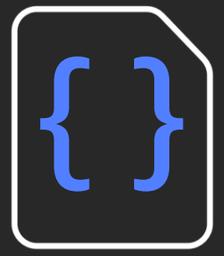


PII-free naming

Data formatting



BSON



JSON



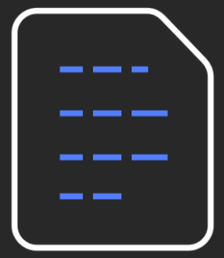
CSV



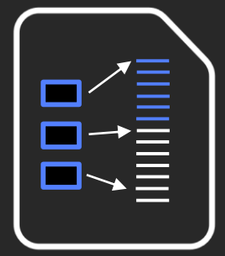
TSV



Avro



Parquet

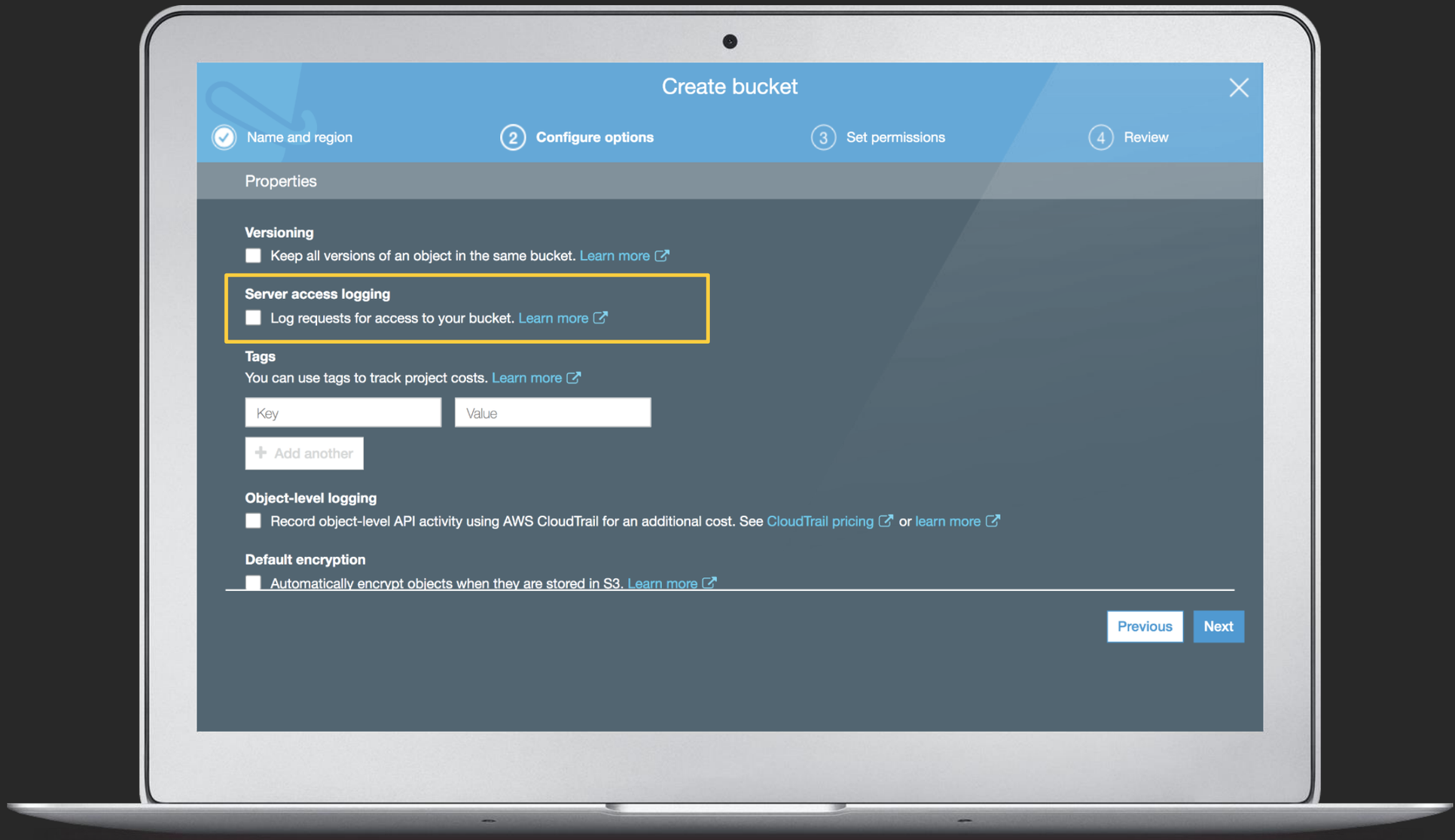


ORC

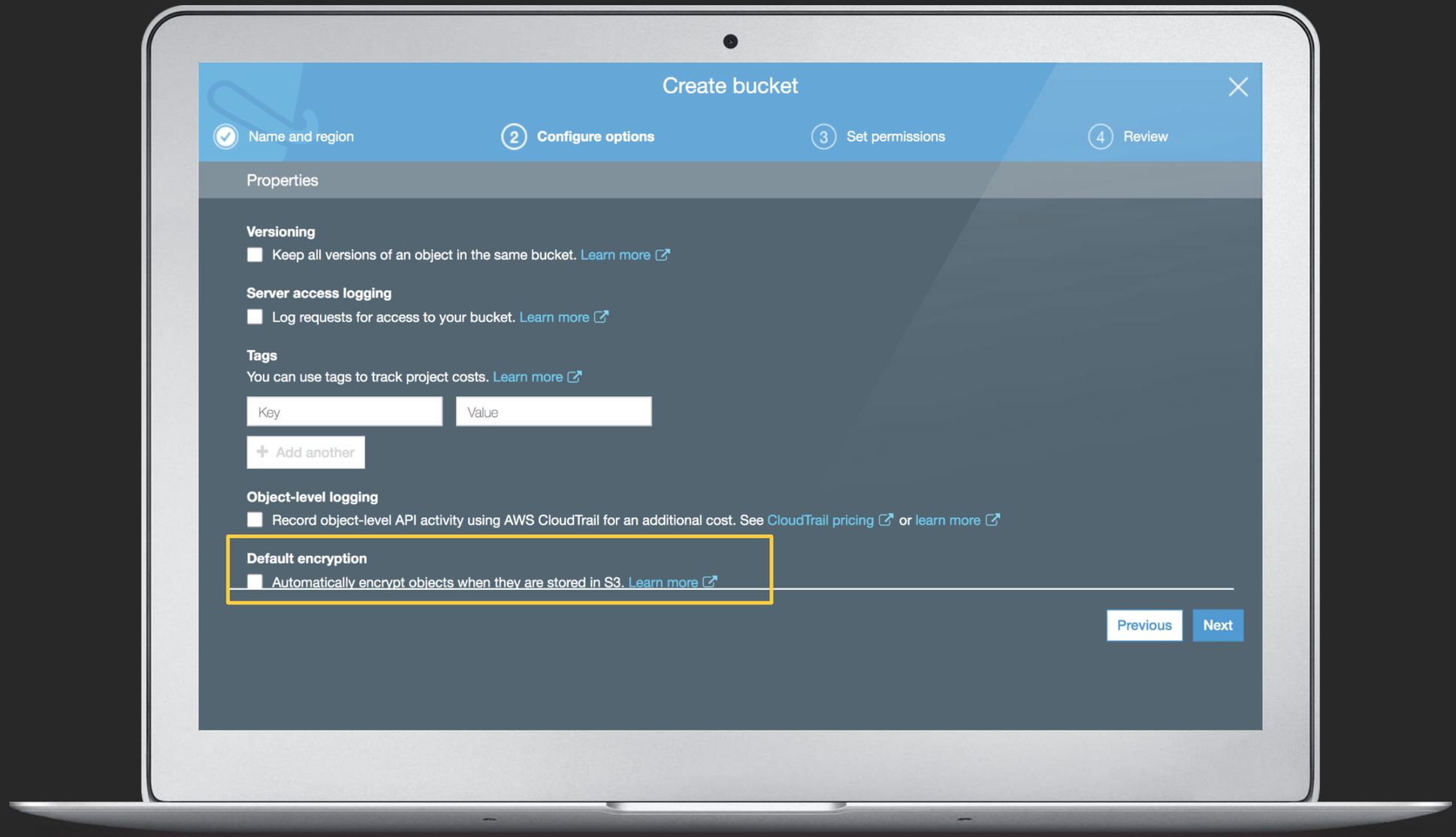


XLSX

Amazon S3 Server access logging



Amazon S3 encryption at rest



AWS Identity and Access Management (IAM) service

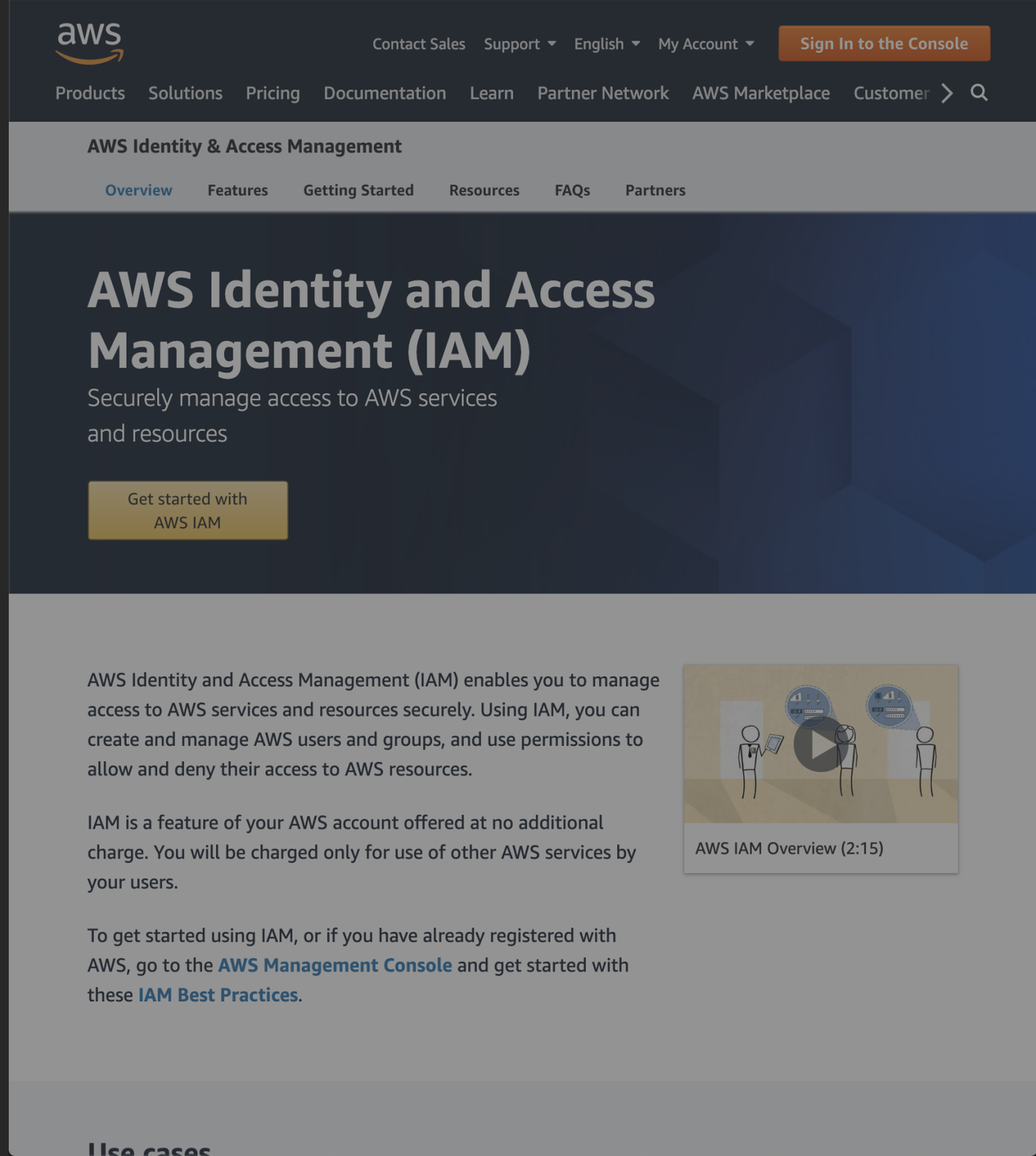
© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

IAM

An AWS service that helps you securely control access to AWS resources.

IAM allows you to control who is authenticated (signed in) and authorized (has permissions) to use resources.

docs.aws.amazon.com/IAM/latest/UserGuide/best-practices



IAM roles



Role policy:
actions +
resources



Trust
relationships

Role policy: Actions + resources



```
{
    "Version": "2012-10-17",
    "Statement": [ {
        "Effect": "Allow",
        "Action": [
            "s3:ListBucket",
            "s3:GetObject",
            "s3:GetObjectVerstion"
        ],
        "Resource": [
            "arn:aws:s3:::mongodb-atlas-data-lake-demo",
            "arn:aws:s3:::mongodb-atlas-data-lake-demo/*"
        ]
    }]
}
```

Trust relationships

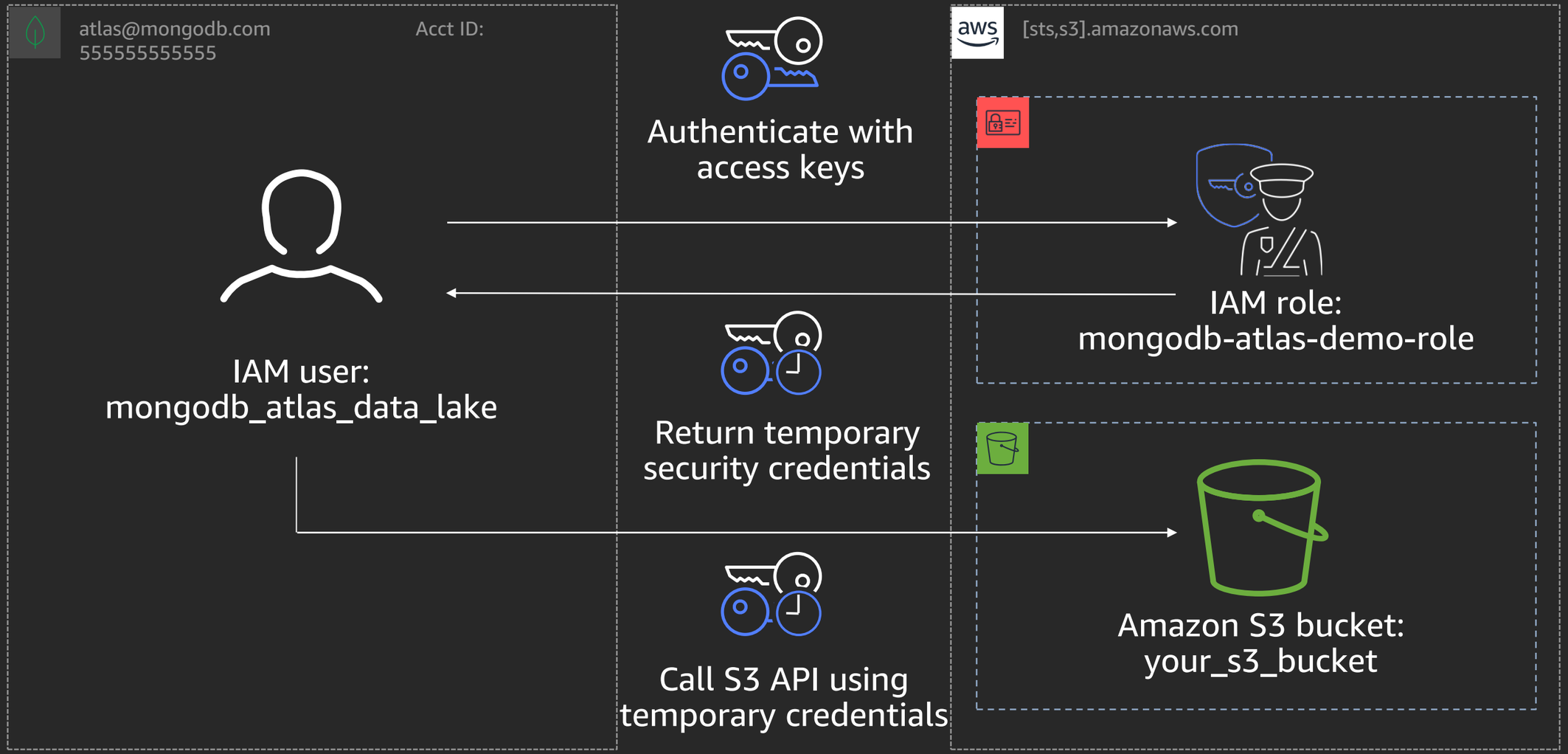


Policy Document

```
{
    "Version": "2012-10-17",
    "Statement": [ {
        "Effect": "Allow",
        "Principal": { "arn:aws:iam::555555555555:user/atlas-data-lake" },
        "Action": "sts:AssumeRole",
        "Condition": {
            "StringEquals": { "sts:ExternalId": ”i8df9289-374f-43e3-959f-629d6343df7c" }
        }
    } ]
}
```

IAM role assumption

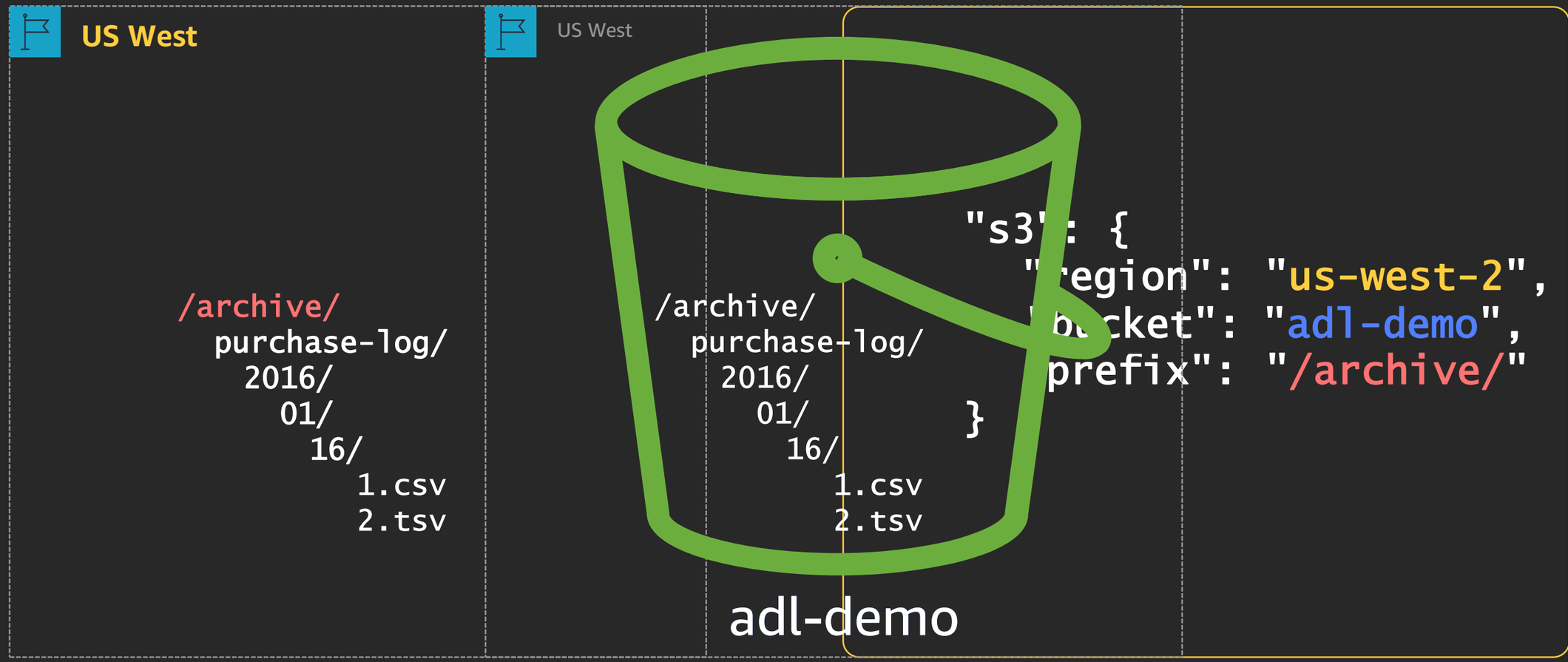




Atlas Data Lake

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Data Lake architecture



Data Lake configuration



```
"stores": [
 {
    "s3": {
        "region": "us-west-2",
        "name":   "demo-store",
        "bucket": "adl-demo",
        "prefix": "/archive/"
    }
 }
]
```

S3 Store: "demo-store"

```
/archive/
  purchase-log/
    2016/
      01/
        16/
           1.csv
           2.tsv
```

Data Lake configuration



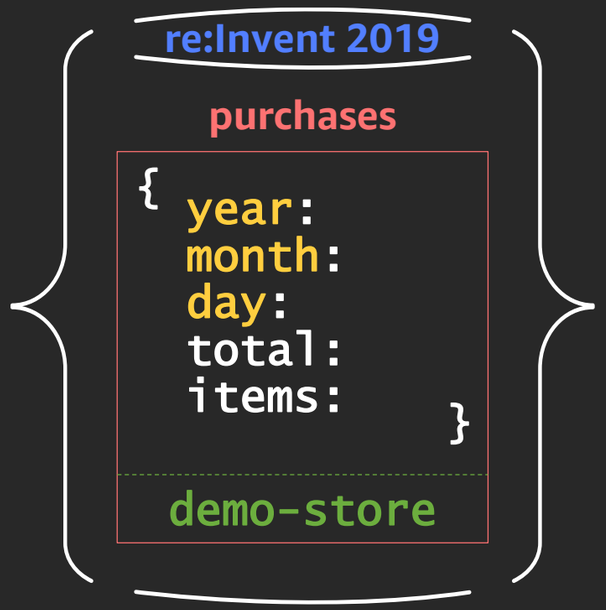
```
"stores": [ ... ]
 {
    "s3": {
        "region": "us-west-2"
        "name":   "demo-store"
        "bucket": "adl-demo"
        "prefix": "/archive/"
    }
 }
]
```

S3 Store: "demo-store"

```
/archive/
  purchase-log/
    2016/
      01/
        16/
           1.csv
           2.tsv
```

Data Lake configuration



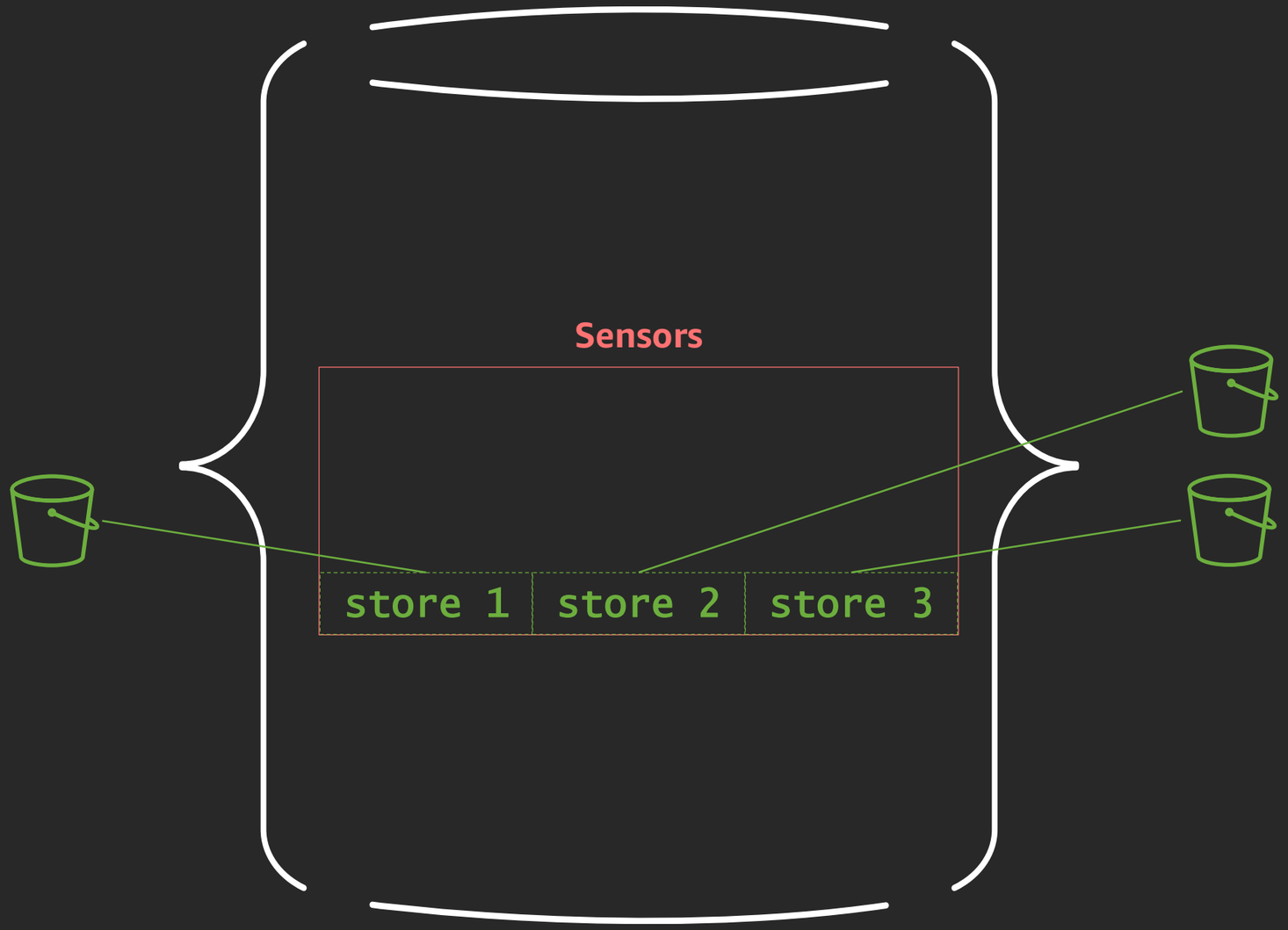


S3 Store: "demo-store"

```
/archive/
  purchase-log/
    2016/
      01/
        16/
           1.csv
           2.tsv
```

```
"stores": [ ... ],
"databases": {
  "reInvent2019": {
    "purchases": [ {
      "store": "demo-store",
      "definition": "/purchase-log/
{year int}/{month int}/{day int}"
    } ]
  },
  // ...
}
```

Data Lake federation



Atlas Data Lake recap

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

S3 data



- S3 Bucket "adl-demo"

/archive/fitness-data

```
20160312.json
20160313.json
20160314.json
```

/archive/purchase-log

```
2016
    03
        01
            1.parquet
            2.parquet
2018
    12
        04
            1.parquet
2017.json.gz
2016.json.gz
```

Sample documents



```
// Purchases

{
    "token":            "25asdfsadx249asdfasdfwq713",
    "item":             "Butterhead Lettuce",
    "price":            13.00,
    "timestamp":        "2017-12-13T02:13:41Z"
}

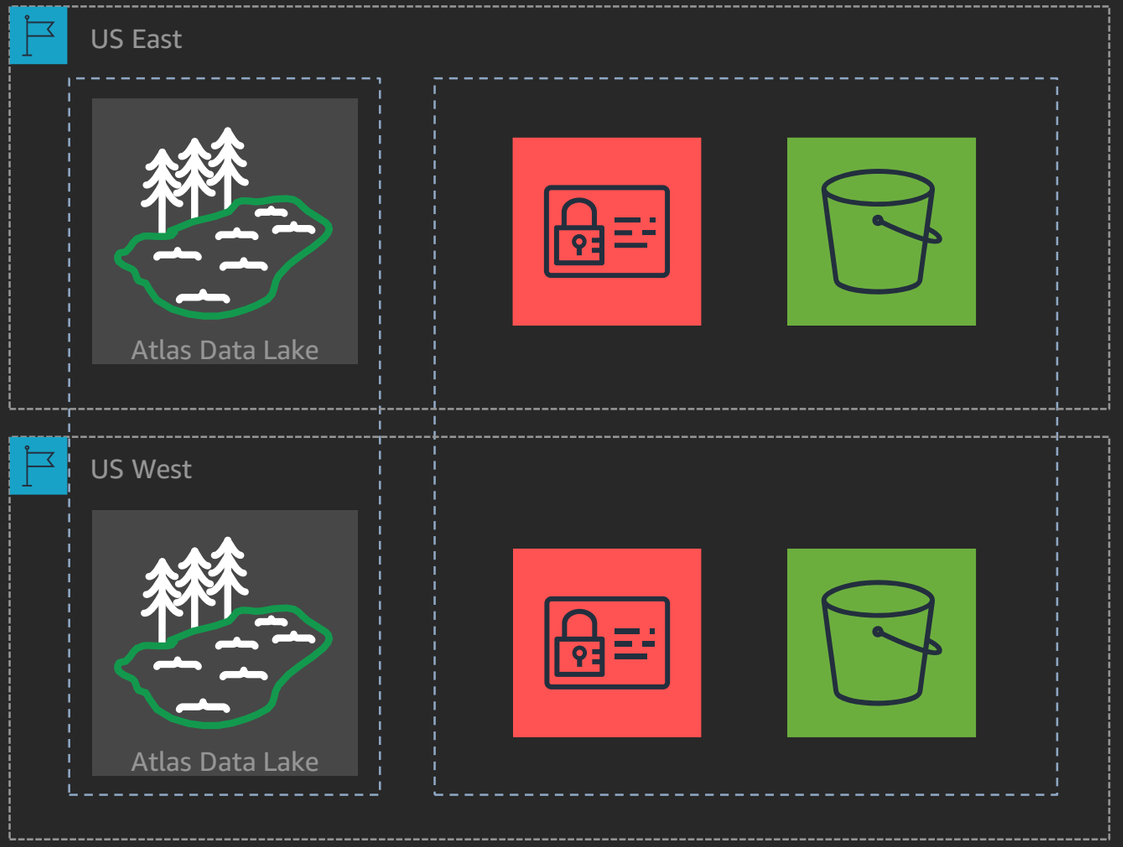
// Fitness data

{
    "id":               1503960366,
    "activity_hour":    "2013-11-20T23:00:00Z",
    "step_total":       20
}
```

Data Lake architecture



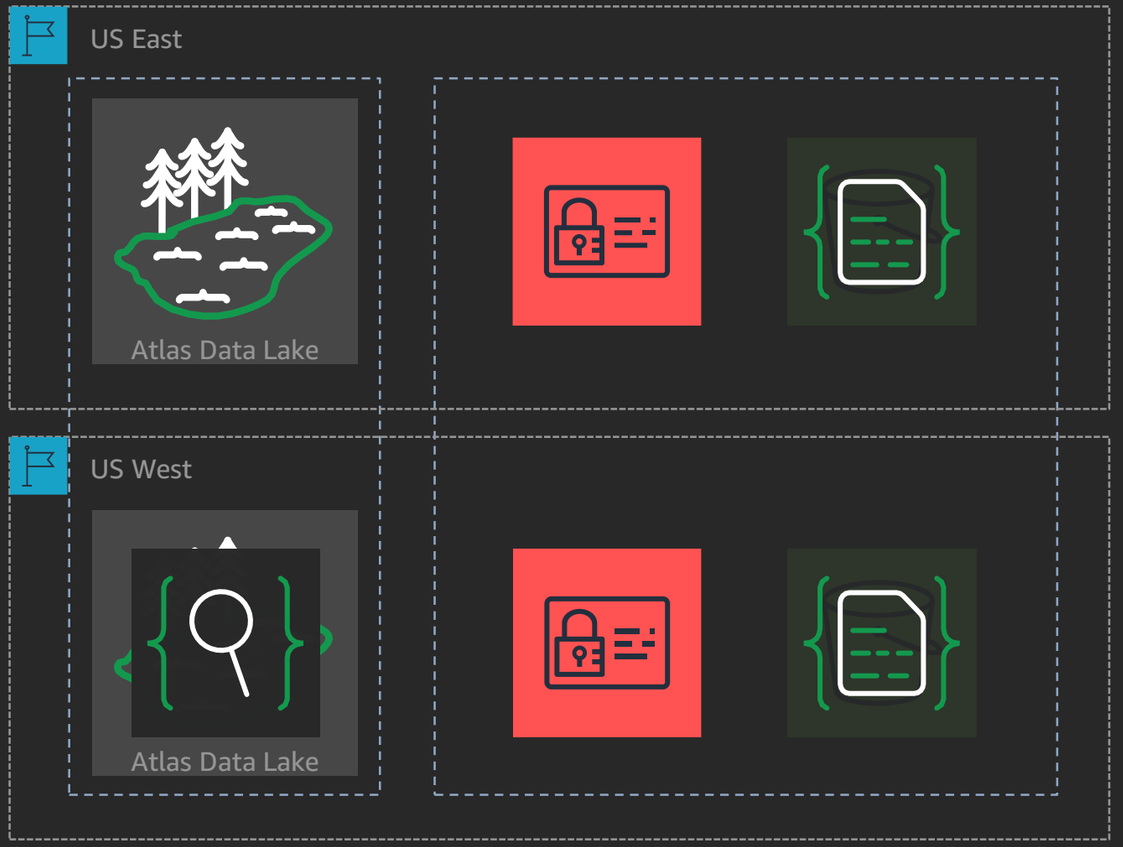
MQL



Data Lake architecture



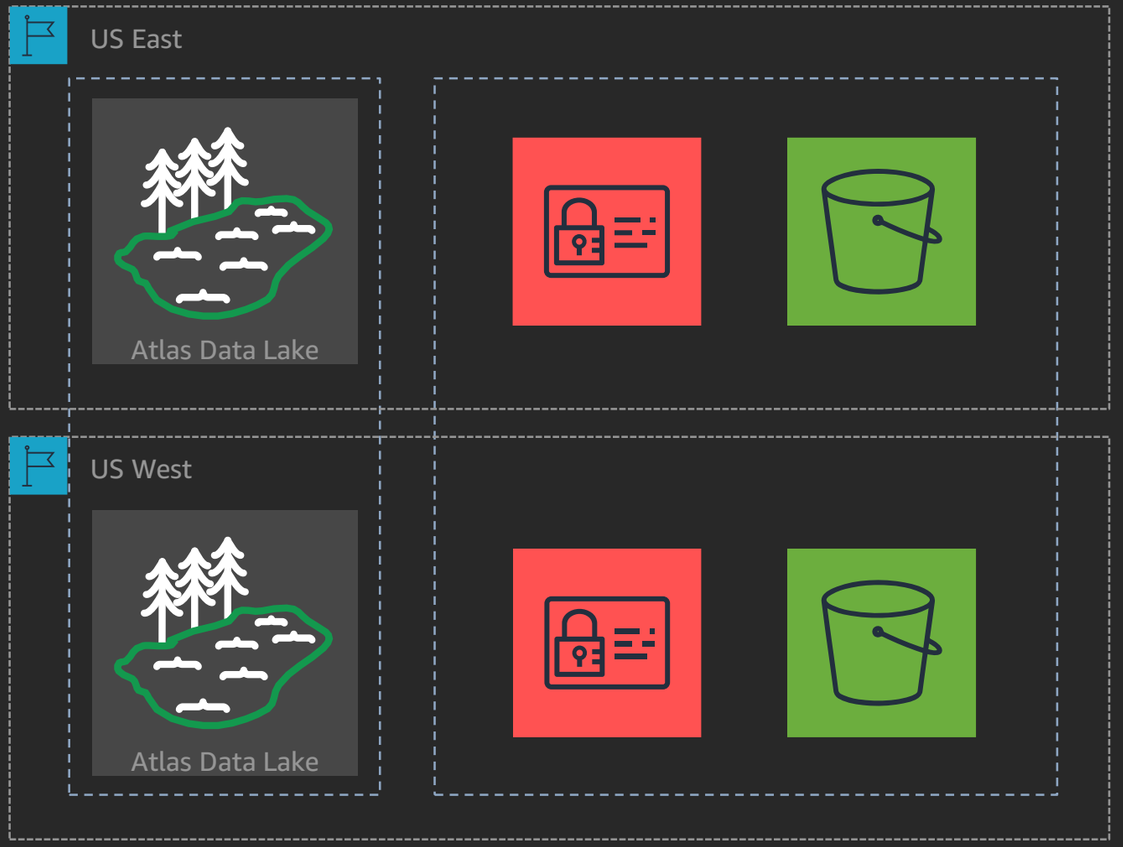
MQL



Data Lake architecture

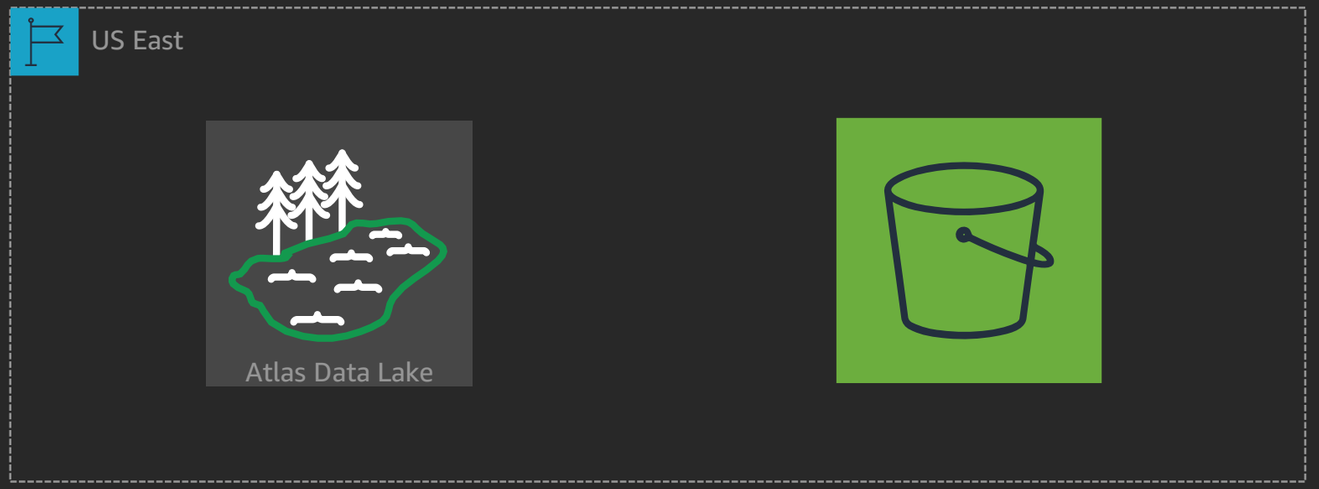


MQL



Data Lake architecture

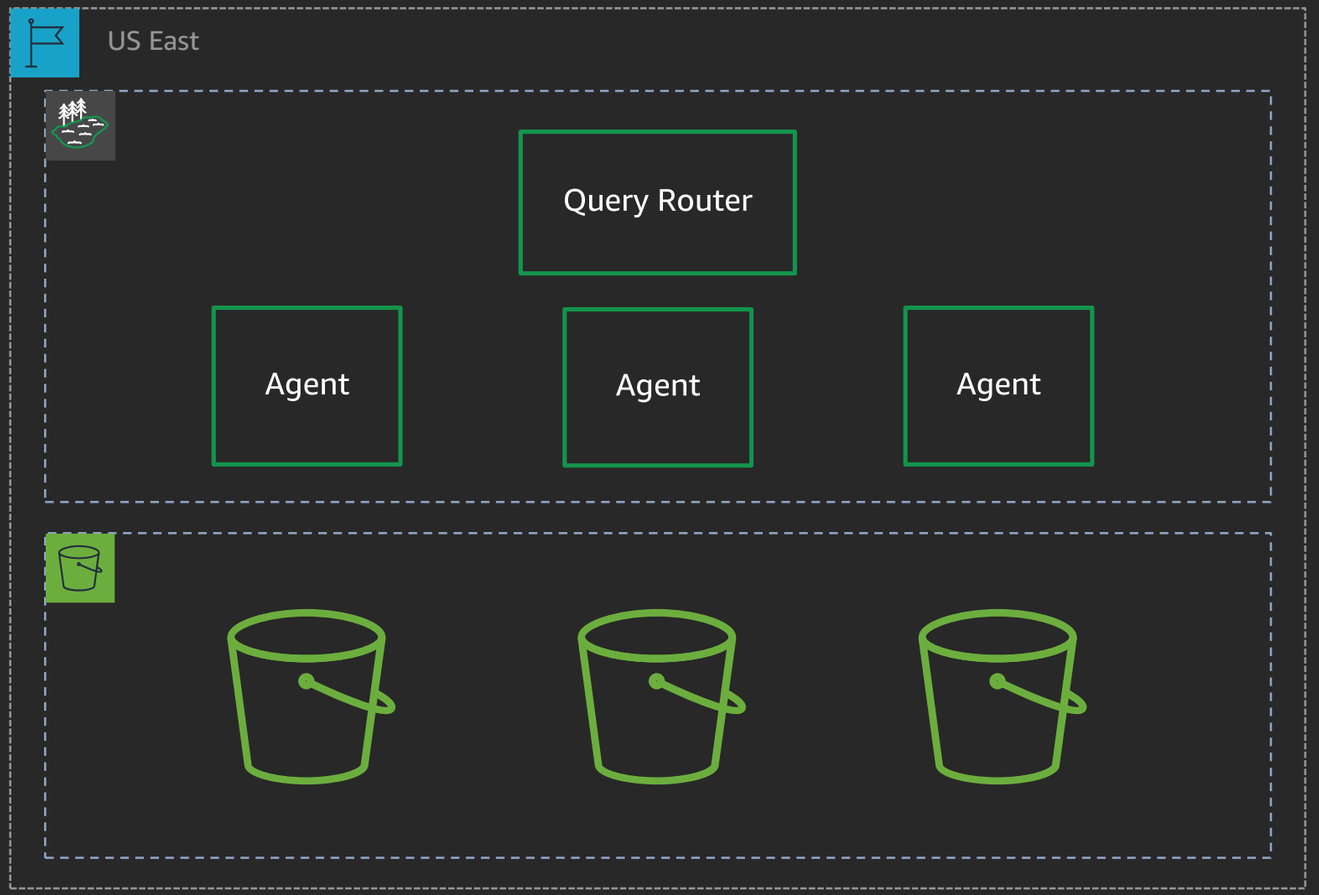




Data Lake architecture



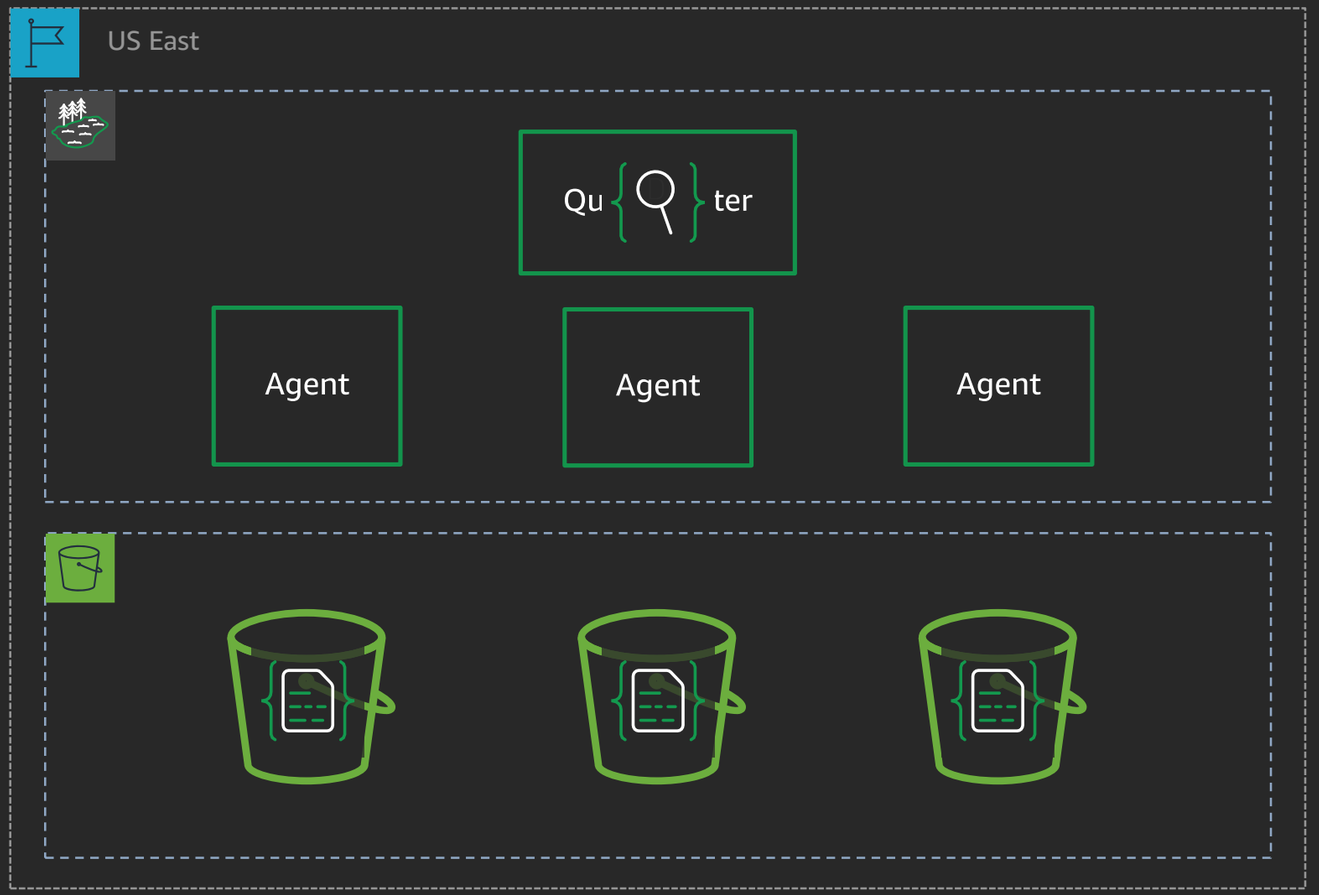
MQL



Data Lake architecture



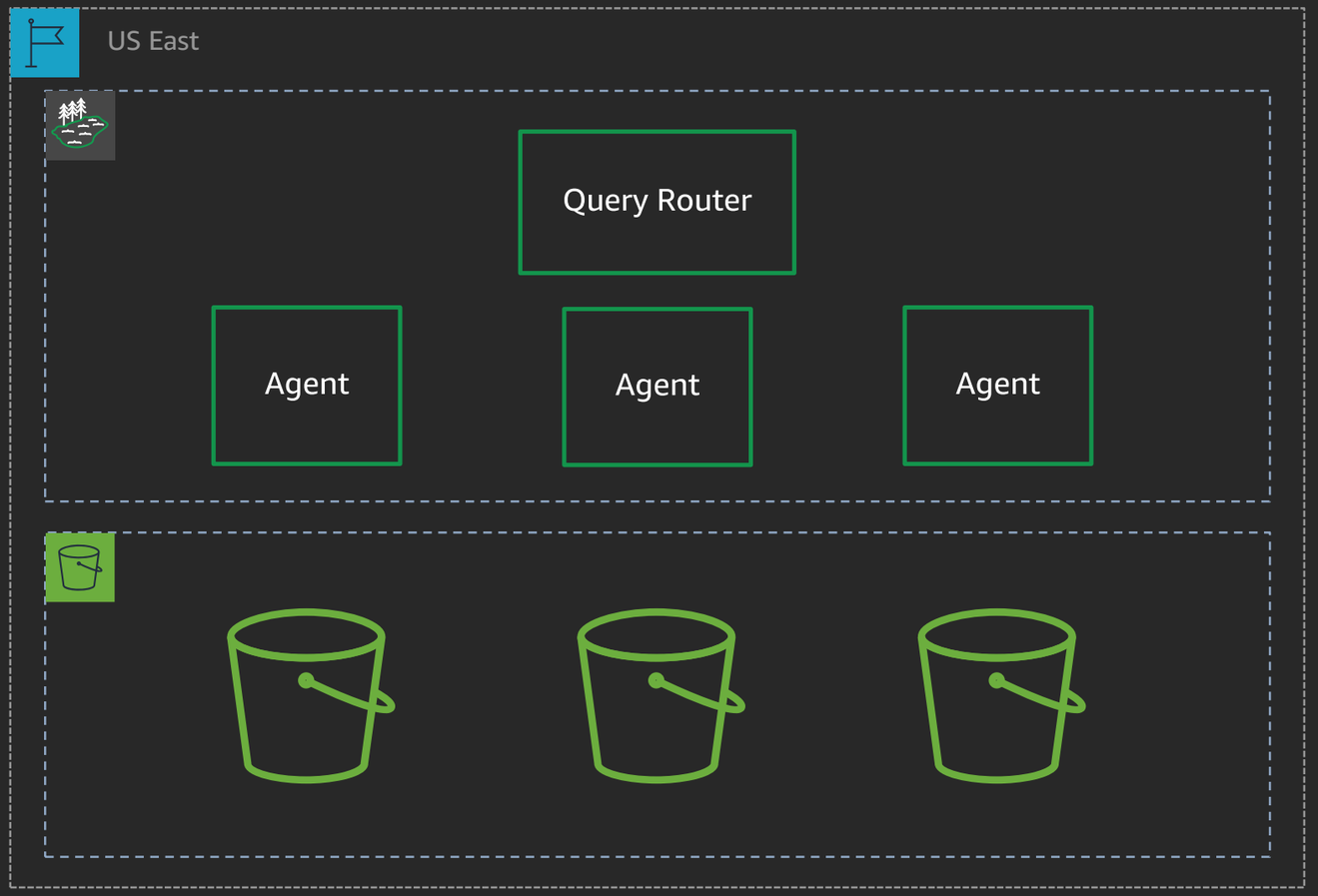
MQL



Data Lake architecture



MQL



Building an Atlas Data Lake

AWS re:Invent

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

aws

Demo

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Atlas Data Lake Internals

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Atlas Data Lake backend



MQL -> Distributed MQL

- Parse
- Parallelize
- Distribute

Lifetime of a query: $group



```
{ $match: {year: 2016, month: 10, day: 31}
{ $group: { _id: "$item", avg_price: {$avg: "$price"}}}
```



Map:

```
{ $match: {year: 2016, month: 10, day: 31}
{ $group: { _id: "$item",
   avg_price_sum: { $sum: "$price" }, avg_price_count: { $sum: 1 }
} }
```

Reduce:

```
{ $group: { _id: "$_id",
   avg_price_sum: { $sum: "$avg_price_sum" }, avg_price_count: { $sum: "$avg_price_count" }
} }
```

Finalize:

```
{ $project: {
   item: "$_id", avg_price: { $divide: ["$avg_price_sum", "$avg_price_count"] }
} }
```

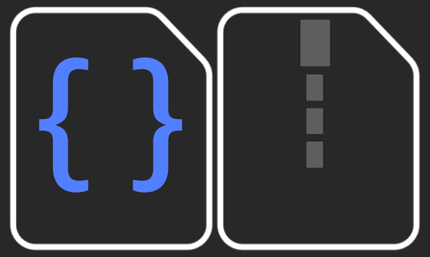
Atlas Data Lake security



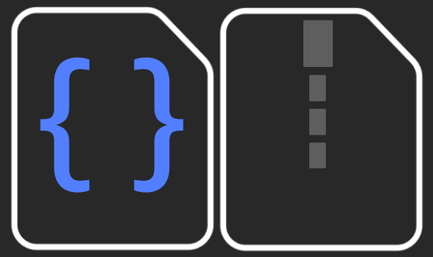
- To make it behave like MongoDB, we:
  - Implemented commands for a read-only server
  - Used the server's aggregation engine
- Same role-based access controls as MongoDB
  - Authentication
  - Authorization
- Requires the use of TLS + SNI
  - SNI = Server Name Indicator
- Fully integrated with Atlas

Atlas Data Lake supported file formats

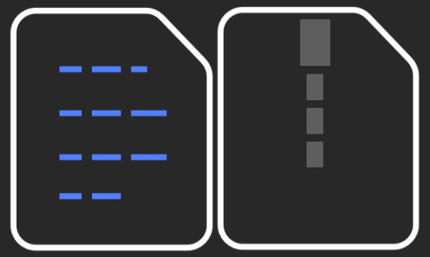




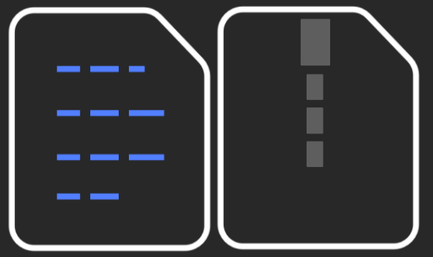
BSON
(gzipped)



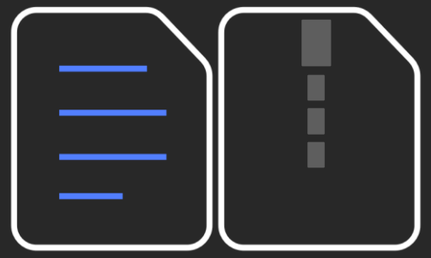
JSON
(gzipped)



CSV
(gzipped)



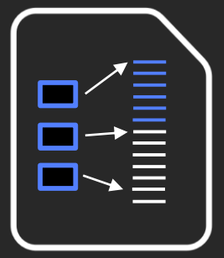
TSV
(gzipped)



Avro
(gzipped)



Parquet



ORC



XLSX

Future

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Data tiering



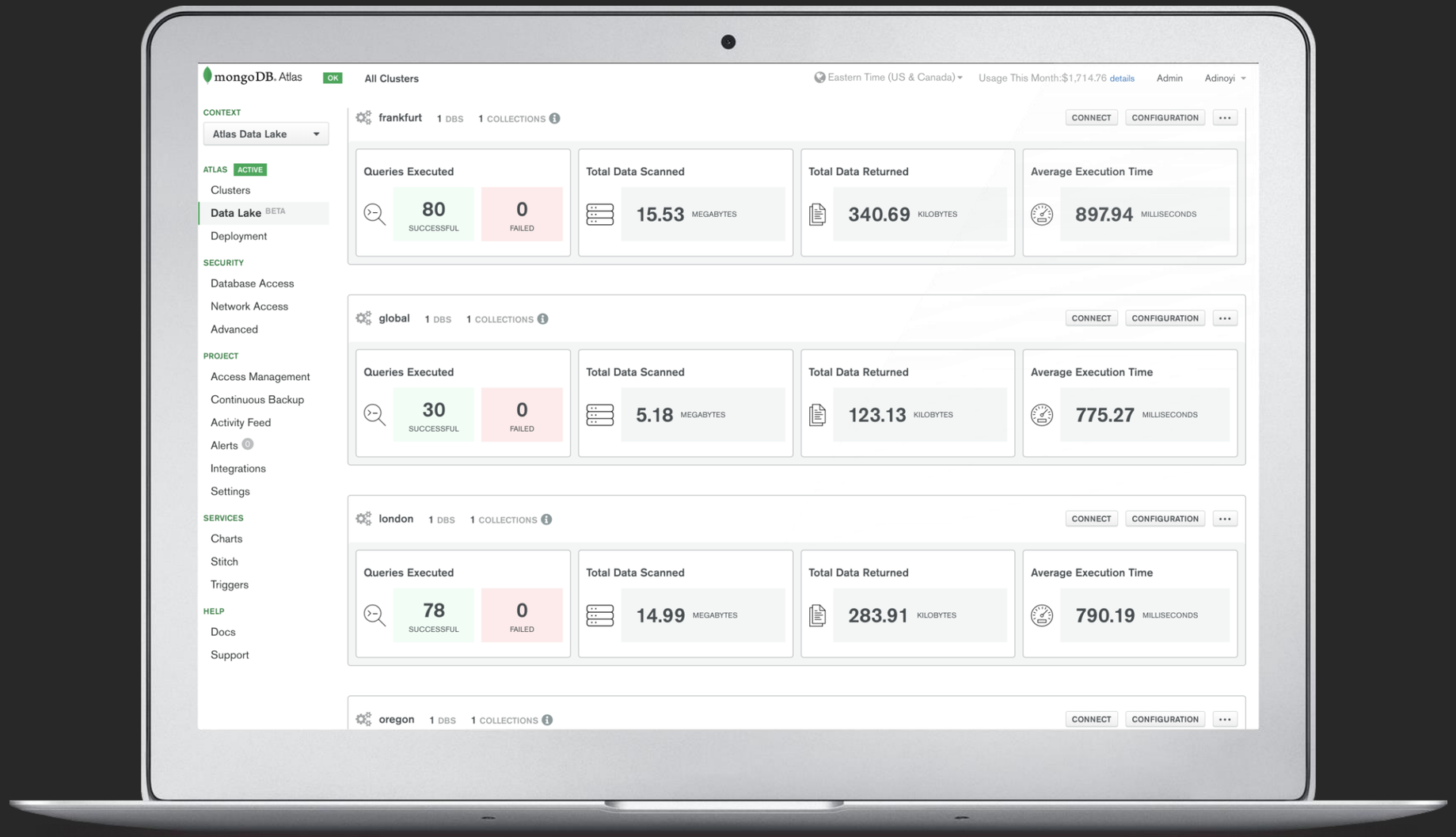
```
reInvent2019: {
  fitnessData: [ {
      store: "mongodb-atlas-data-lake-demo",
      definition: "/fitness-data/{year int:\\d{4}}{monthint:\\d{2}}{day int:\\d{2}}"
    }, {
      store: "atlas",
      cluster: "my-atlas-cluster",
      db: "customers",
      collection: "invoices"
  } ]
}
```

What's next



- Query Execution Optimizations
- SQL Support
- Data Tiering
- View Creation
- …

MongoDB Atlas Data Lake



mongodb.com/reinvent19-data-lake

Q&A

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Thank you!

Sahir Azam
sahir@mongodb.com

Wisdom Omuya
wisdom@mongodb.com

AWS re:Invent

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

aws



Please complete the session survey in the mobile app.

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.