STG312

# AWS re:INVENT

## Best Practices for Building a Data Lake on Amazon S3 & Amazon Glacier

with special guests: Viber and Airbnb

John Mallory, Business Development, Storage
PD Dutta, Sr. Product Manager, Amazon S3

November 30, 2017

# What to expect

- Defining the AWS data lake on Amazon S3 and Amazon Glacier
- Data cataloging
- Security, performance, and analytics best practices
- Special guests Viber and Airbnb

# Defining the AWS data lake

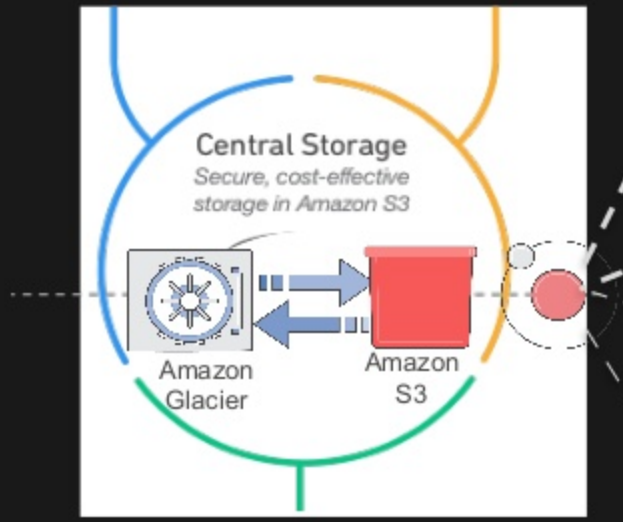
Data lake is an architecture with a virtually limitless centralized storage platform capable of categorization, processing, analysis, and consumption of heterogeneous data sets


## Key data lake attributes

- Decoupled storage and compute
- Rapid ingest and transformation
- Secure multi-tenancy
- Query in place
- Schema on read

# What can you do with a data lake?

## Batch processing




Amazon Athena
Clusterless SQL Query

Amazon Redshift
Data Warehouse

Amazon Glue
Clusterless ETL

Amazon EMR
Hadoop/Hive/Presto

**BI & Visualization**

Amazon QuickSight

looker

tableau

kibana

COGNOS

SAP BusinessObjects

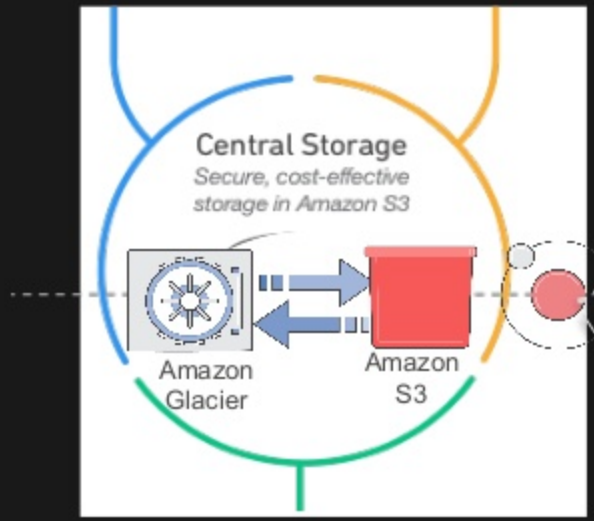
MicroStrategy




aws

# What can you do with a data lake?

## Streaming and real-time analytics




Amazon Elasticsearch Service

Apache Flink on EMR

Amazon Kinesis Analytics

Amazon ElastiCache

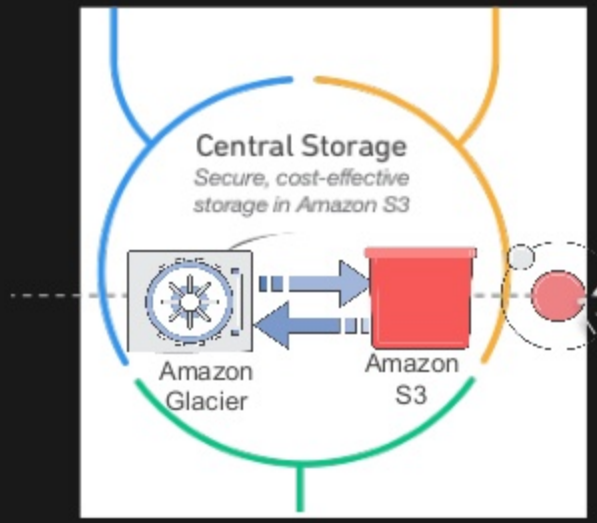
AWS Lambda

Apache Storm on EMR

Spark Streaming on EMR

# What can you do with a data lake?

## AI and machine learning




**Amazon Polly**
Life-like speech

**Amazon Rekognition**
Image analysis

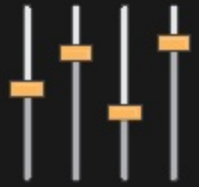
**Deep learning Frameworks**
MXNet, TensorFlow, Theano, Caffe, Torch

**Amazon Lex**
Conversational engine

# Reasons to choose Amazon S3 for data lake

Unmatched durability, availability, and scalability

Best security, compliance, and audit capability



Object-level control at any scale

Business insight into your data

Most ways to bring data in

Twice as many partner integrations

# Optimize costs with data tiering

Hot


HDFS

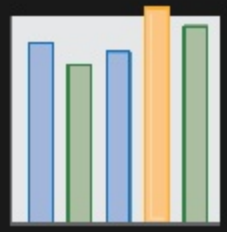
Amazon
S3 standard

Amazon S3—
infrequent access

Amazon
Glacier

Cold

- ✓ Use EMR/Hadoop with local HDFS for hottest data sets
- ✓ Store cooler data in S3 and Glacier to reduce costs
- ✓ Use S3 Analytics to optimize tiering strategy



S3 Analytics

# Multiple data lake ingestion methods

## AWS Snowball and AWS Snowmobile

- PB-scale migration

## Amazon Kinesis Firehose

- Ingest device streams
- Transform and store on Amazon S3

## AWS Storage Gateway

- Migrate legacy files

## AWS Direct Connect

- On-premises integration

## Native/ISV Connectors

- Ecosystem integration

## Amazon S3 Transfer Acceleration

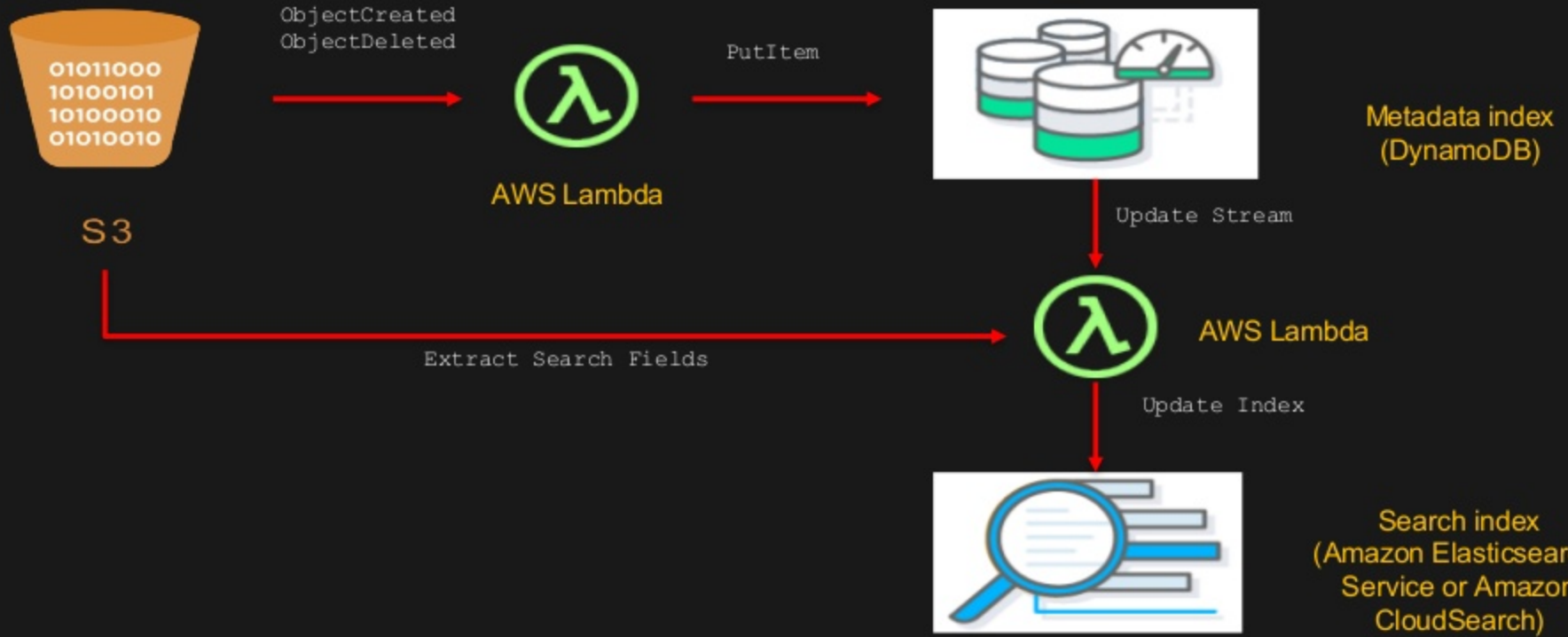
- Long-distance data transfer

AWS re:Invent

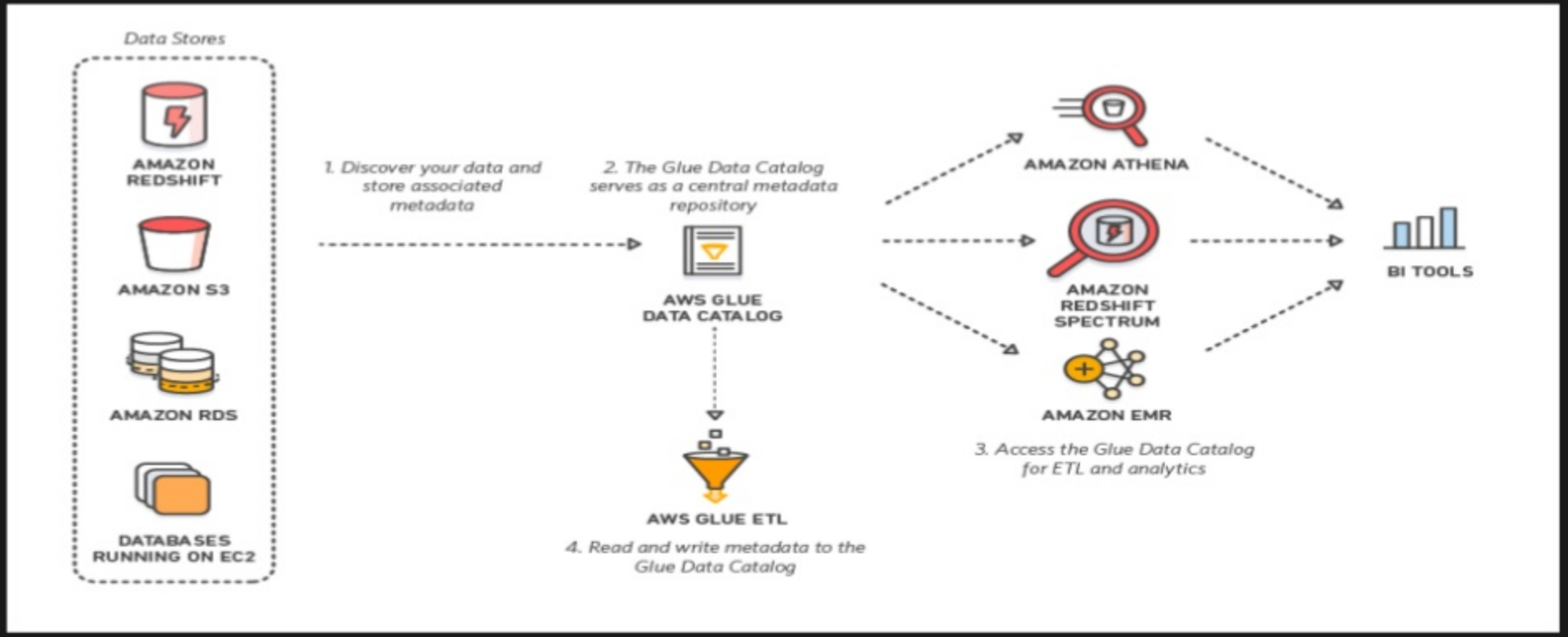


aws

# Catalog your S3 data



# AWS Glue analytics data catalog



# AWS Glue analytics data catalog

Manage table metadata through a Hive metastore API or Hive SQL. Supported by tools like Hive, Presto, Spark, etc.

We added a few extensions:

- **Search** over metadata for data discovery
- **Connection info**—JDBC URLs, credentials
- **Classification** for identifying and parsing files
- **Versioning** of table metadata as schemas evolve and other metadata are updated

Populate using Hive DDL, bulk import, or automatically through **crawlers**.

# Populating the AWS Glue data catalog

## Crawlers automatically build your data catalog and keep it in sync

Automatically discover new data, extracts schema definitions

- Detect schema changes and version tables
- Detect Hive style partitions on Amazon S3

Built-in classifiers for popular types; custom classifiers using Grok expressions

Run via Lambda triggers or scheduled; serverless—only pay when crawler runs

# Securing your data on Amazon S3

AWS re:Invent



aws

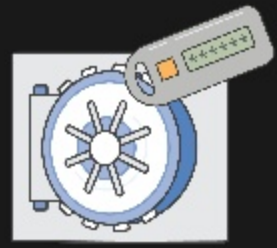
# AWS data lake security entitlements

## Identity and access

- Amazon Macie *New*
- Permission checks *New*
- AWS Config Rules *New*
- IAM & bucket policies
- Access control lists

## Encryption

- Default encryption *New*
- Server-side encryption
- Client-side encryption
- SSL endpoints
- Encryption status in inventory *New*
- CRR with KMS *New*



## Compliance

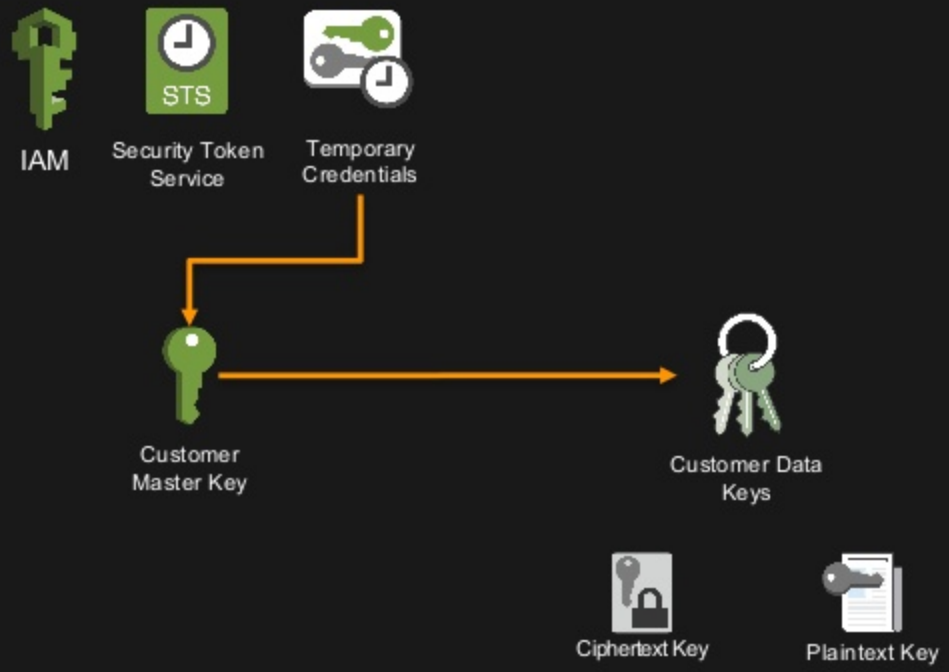
- Certifications—HIPAA, FedRAMP, PCI-DSS
- Cloud HSM integration
- Versioning & MFA deletes
- Audit logging

AWS re:Invent

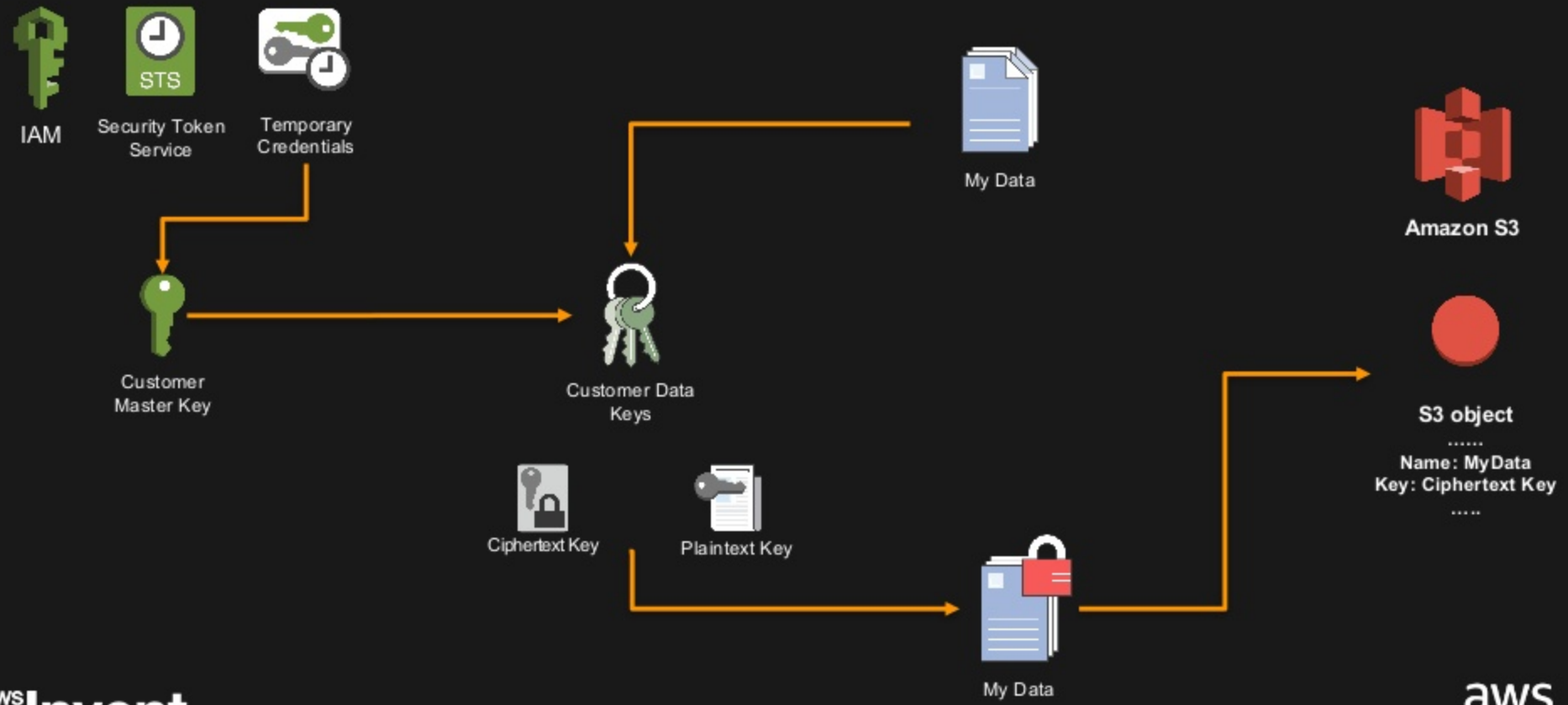


aws

# Security: Access to encryption keys



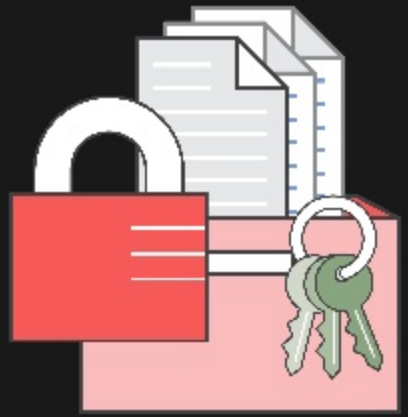
# Security: Access to encryption keys



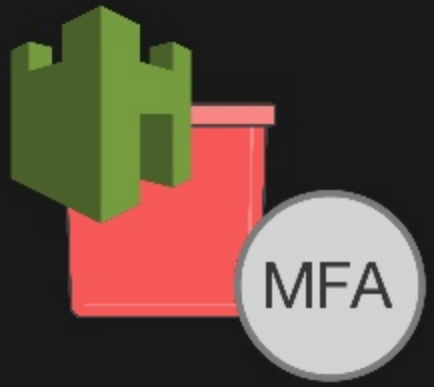
# Security for your data lake

Pro Tip

IAM best practices
SSL/TLS connections



Server-side encryption
Bucket policies




Versioning; recycle bin
MFA deletes

# Optimizing performance on Amazon S3

# Getting high throughput with Amazon S3

Most customers need not worry about introducing entropy in key names

Consider 3-4 character hash for higher requests per second

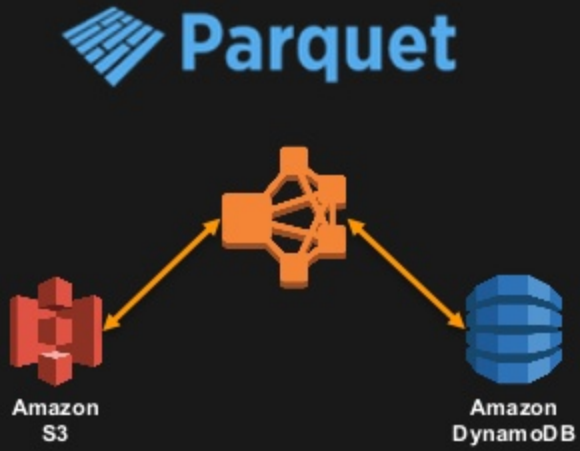
examplebucket/*232a*-*2017-26-05-15-00-00*/cust1234234/photo1.jpg
examplebucket/*7b54*-*2017-26-05-15-00-00*/cust3857422/photo2.jpg
examplebucket/*921c*-*2017-26-05-15-00-00*/cust1248473/photo2.jpg

**A bit more LIST friendly:**

examplebucket/*animations/232a*-2017-26-05-15-00-00/cust1234234/animation1.obj
examplebucket/*videos/ba65*-2017-26-05-15-00-00/cust8474937/video2.mpg
examplebucket/*photos/8761*-2017-26-05-15-00-00/cust1248473/photo3.jpg

Random hash should come before patterns such as dates and sequential IDs
Always first ensure that your application can accommodate

# Optimizing data lake performance


**Aggregate small files**

EMR: S3distcp

Amazon Kinesis Firehose

**S3 Select**

Big data cheaper, faster

Up to 400% faster

**Data Formats**

Columnar formats

EMRFS consistent view

# Big data analytics & query in place

# Amazon analytics end-to-end architecture

IAM

Amazon QuickSight

Amazon ML/MXNet

EMR

Athena

Amazon Redshift Spectrum

Data Catalog

RDS

Kinesis

Database Migration Service

Other Sources

AWS Glue

Amazon S3

# Introducing Amazon S3 Select *New*

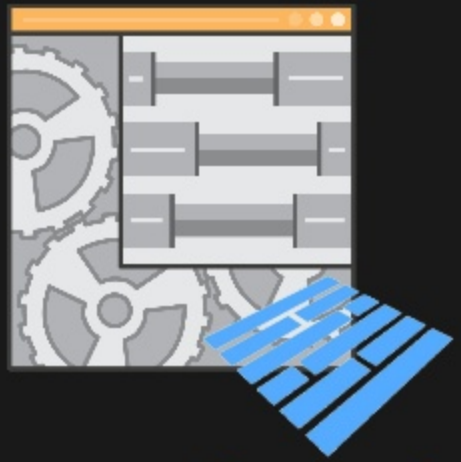
## Simple API to retrieve subset of data based on a SQL expression


Accelerate performance for data retrieval and processing by up to 400%

Simplify compute by retrieving subset of data in a common format

# Amazon EMR: Decouple compute & storage





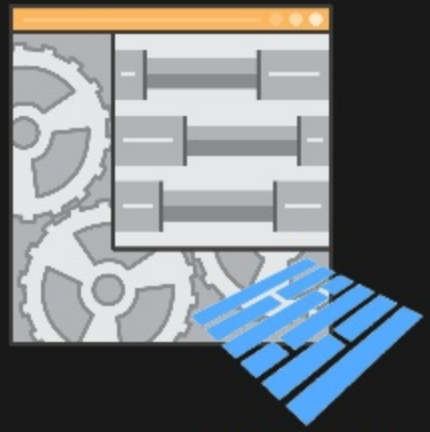

Highly distributed processing frameworks such as **Hadoop/Spark**

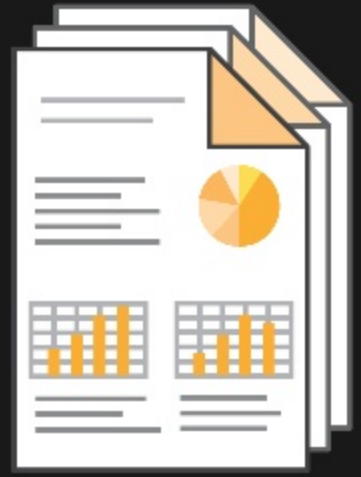
**Compress** datasets
**Columnar** file formats

**Aggregate small files**
S3distcp “group-by” clause

Pro Tip

# Amazon Redshift Spectrum: Exabyte Scale query-in-place



Structured data w/ joins

Multiple on-demand clusters-scale concurrency

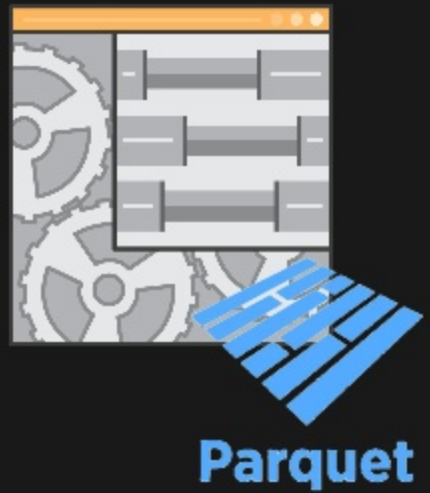
Columnar file formats

Data partitioning

Better query performance with predicate pushdown

# Amazon Athena: Query without ETL

Pro Tip




Serverless service

Schema on read

Compress datasets

Columnar file formats

Optimize file sizes

Optimize querying (Presto backend)

# Use the right data formats

- Pay by the amount of data scanned per query
- Use compressed columnar formats
  - Parquet
  - ORC
- Easy to integrate with wide variety of tools

```
SELECT elb_name,
       uptime,
       downtime,
       cast(downtime as DOUBLE)/cast(uptime as DOUBLE) uptime_downtime_ratio
FROM
    (SELECT elb_name,
        sum(case elb_response_code
        WHEN '200' THEN
        1
        ELSE 0 end) AS uptime, sum(case elb_response_code
        WHEN '404' THEN
        1
        ELSE 0 end) AS downtime
    FROM elb_logs_raw_native
    GROUP BY  elb_name)
```

| Dataset | Size on Amazon S3 | Query Run time | Data Scanned | Cost |
|---|---|---|---|---|
| **Logs stored as text files** | 1 TB | 237 seconds | 1.15TB | $5.75 |
| **Logs stored in Apache Parquet format*** | 130 GB | 5.13 seconds | 2.69 GB | $0.013 |
| **Savings** | **87% less with Parquet** | **34x faster** | **99% less data scanned** | **99.7% cheaper** |

Special guest: Viber

# Viber data lake

Amir Ish-Shalom

Chief Architect, Viber

# Rakuten Viber

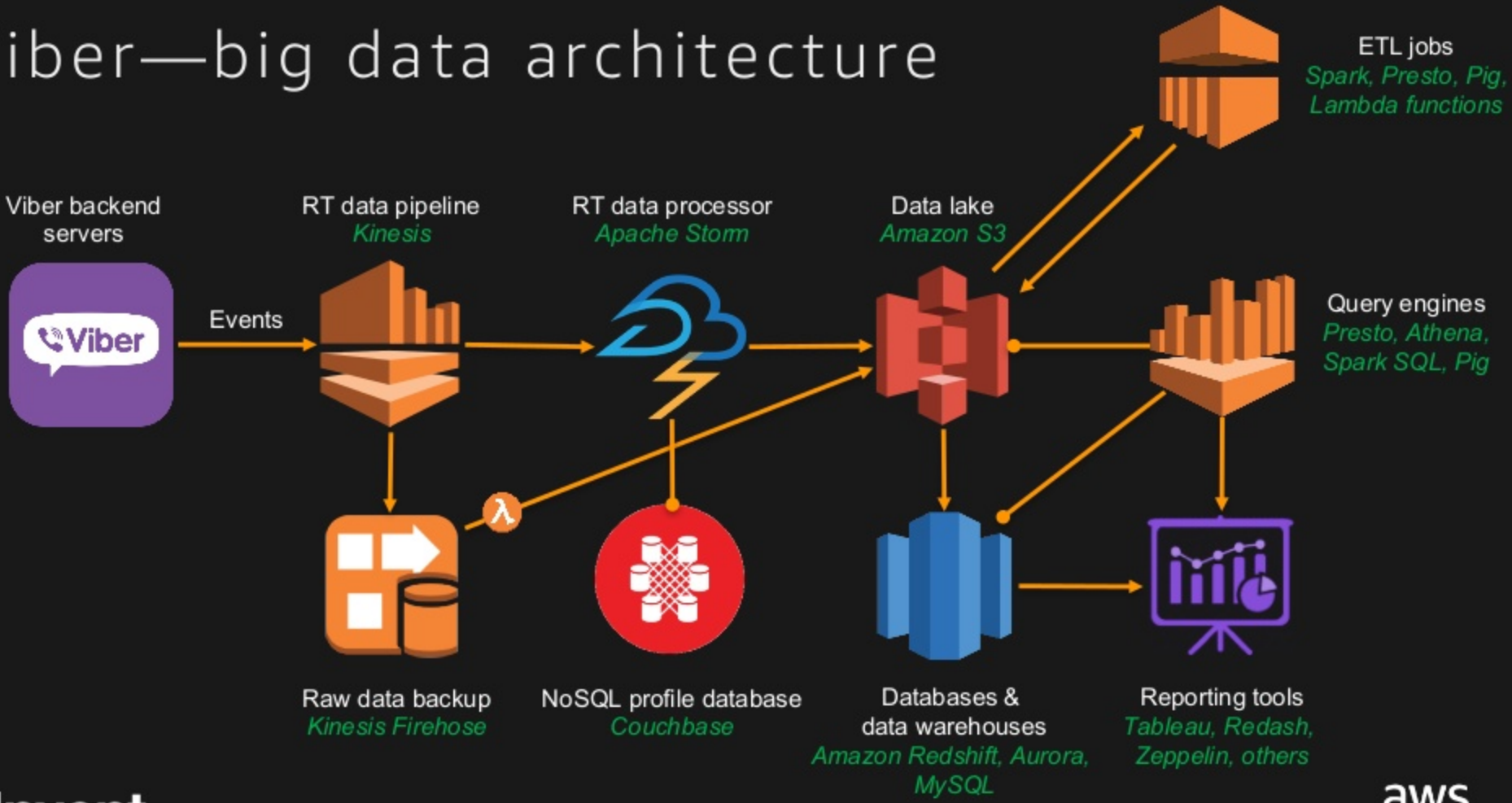
- Messaging (including group)
- Secure end-to-end encryption
- Rich media and chat extensions
- Full multiple device support
- HD video and voice calls
- Viber out and Viber in
- Public chats and accounts
- Chatbots



# Big data @ Viber

- Close to 1 billion users worldwide
- Globally used in 230 countries
- 10-15 billion events daily (2 TB)
- 300,000 events per second (peak hours)
- 5 PB of data stored on Amazon S3/Amazon Glacier
- NoSQL DB (Couchbase) performing
  2 million TPS on 20 TB of data
  with 35 billion keys

# Viber—big data architecture



# Data lake challenges

Use case #1: S3 performance

Use case #2: Data access rights

Use case #3: Encrypted data storage

Use case #4: Storage of data from third parties

# Use case: S3 performance

**Challenge:**

- Over 300 different event types with large throughput variance
- Storm data processor created many small files, especially for lower throughput events
- Events were stored in Hive partitioned folders (Y/M/D/H), which are not optimal for Amazon S3
- Running a query over these events using Presto could generate up to 15K tps on a single S3 bucket, resulting in 5xx errors and throttling the whole bucket for other processes

# Use case: S3 performance

**Solution:**

- Concatenate small files into large files, optimally 100 MB+
- Convert files into columnar file format such as Parquet or ORC

# Use case: S3 performance

**Future solution:**

- Concatenate and convert files in a single process (Glue?)
- Use better partitioned hive directory format (H/D/M/Y instead of Y/M/D/H)
- Use even larger files for high-throughput events

# Use case: Data access rights

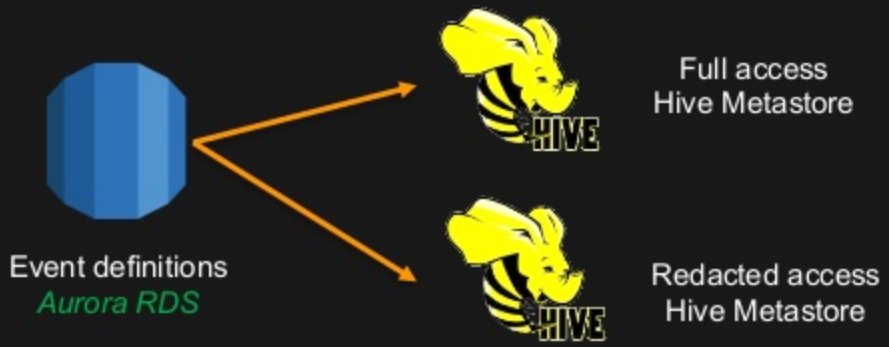
**Challenge:**

- Events can contain sensitive personal data
- Allow access to events without exposing personal data

# Use case: Data access rights

**Solution #1:**

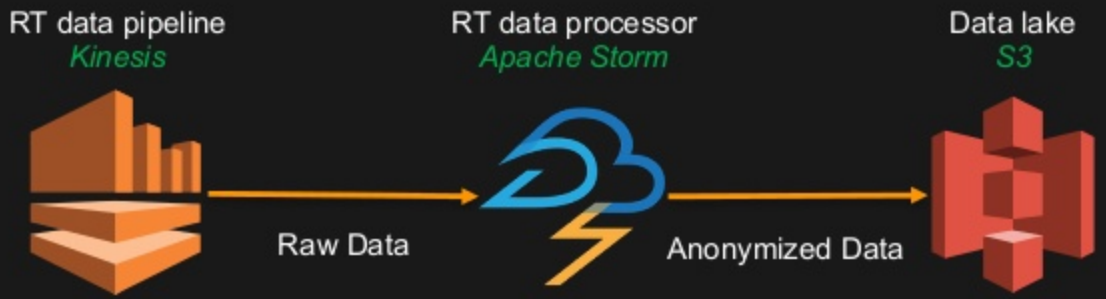
- Create separate Hive metastores for full and redacted data
- Reporting tools will select relevant metastore based on current user



# Use case: Data access rights

**Solution #2:**

- Anonymize sensitive personal data
- Legal compliancy issues
- Limits data science capabilities



# Use case: Encrypted data storage

**Challenge:**

- Store daily backups in S3
- Backup must be encrypted
- Strict access control
- Regional replication
- Complex data retention

# Use Case: Encrypted data storage

## Solution


**Security**—encrypt using SSE-KMS

**Access**—require permissions to both S3 bucket & KMS key

**Tagging**—use S3 object-level tagging to apply different lifecycle policies to certain objects

**Multiple regions—u**se CRR-KMS

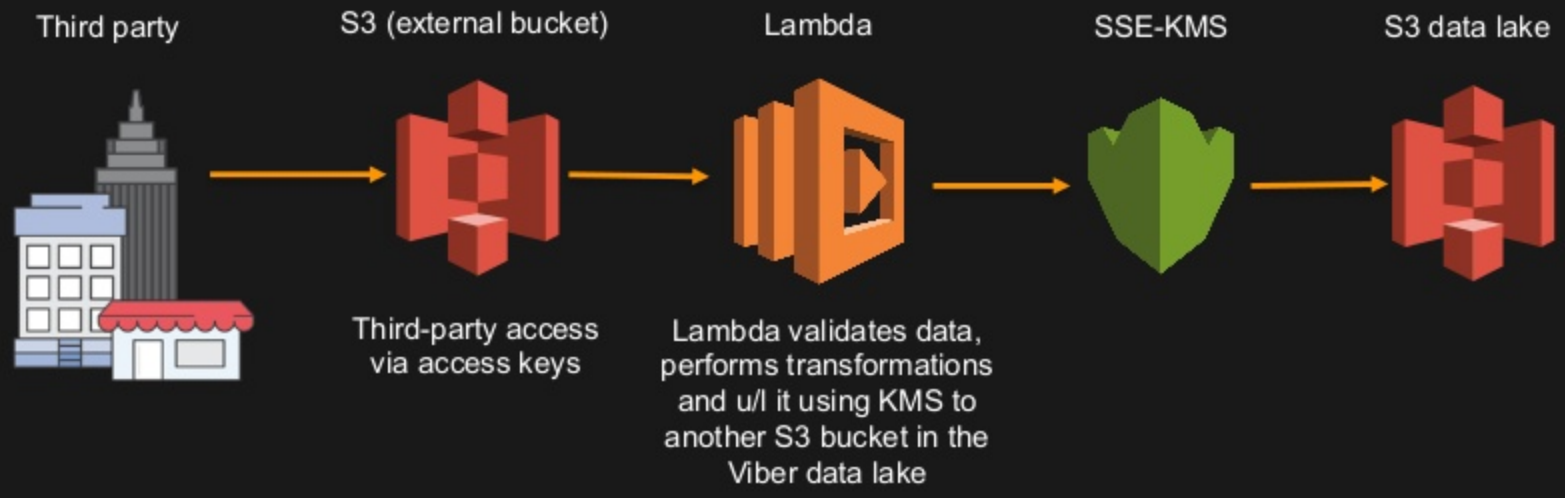
# Use case: Storage of data from third parties

**Challenge:**

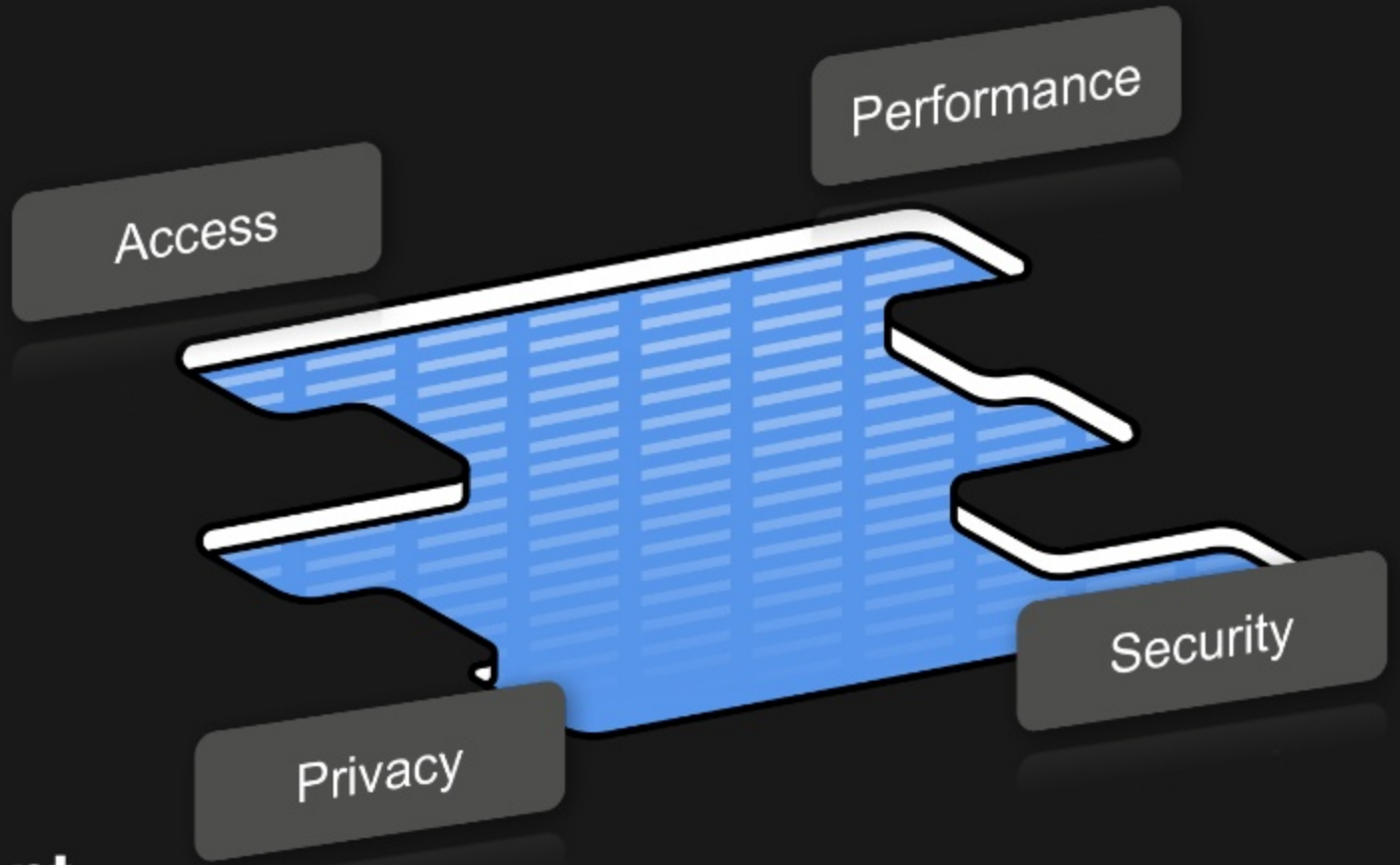
- Securely store data from third parties in data lake
- Validate data before storing
- Allow optional data transformation

# Use case: Storage of data from third parties

**Solution:**



# Viber data lake—summary



Special guest: Airbnb

# Airbnb—tiered storage system

Hongbo Zeng

Software Engineer, Airbnb

AWS re:Invent



aws

# Agenda

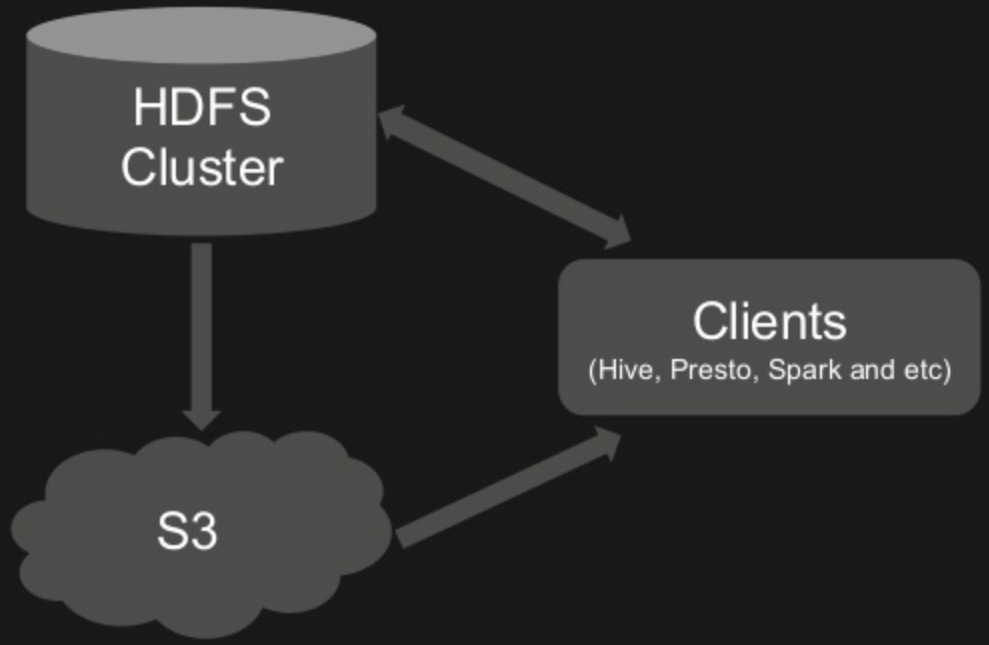
- Challenges
- Tiered storage system
- S3A+ file system

# Motivations

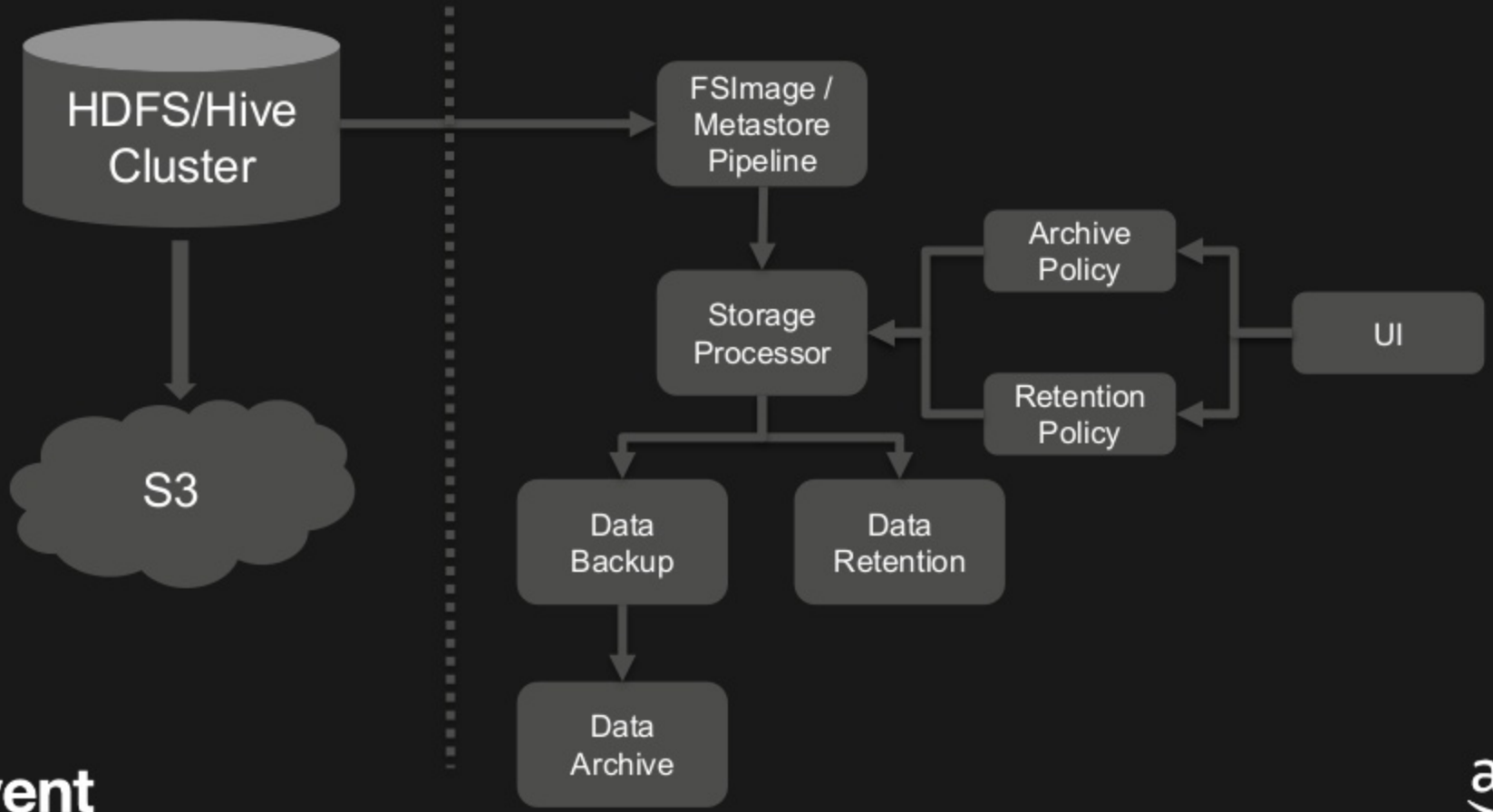
- 3x+ YoY data growth
- HDFS bottlenecks
    - Namenode scalability
    - Cost
- S3 is an object store
    - Eventual consistency
    - Metadata retrieval performance
    - Read/write performance

# Tiered storage system

- HDFS + S3
  - Hot data on HDFS
  - Warm and cold data on S3
- Bring the best of both together
  - Performance
  - Scalability
  - Cost



# Architecture



# Backup

# Archive

- Metadata validation
    - Is there a successful backup?
    - Is the backup location valid?
    - Anything changed since the latest backup?
- Data validation
    - File count
    - File size
- Archive
    - Update the location of partitions

# The journey of a partition

# Problem solved?

- HDFS bottlenecks
    - Namenode scalability
    - Cost
- S3 is an object store
    - Eventual consistency
    - Metadata retrieval performance
    - Read/write performance

# Problem solved … partly

- HDFS bottlenecks
  - ~~Namenode scalability~~
  - ~~Cost~~
- S3 is an object store
  - ~~Eventual consistency~~
  - **Metadata retrieval performance**
  - **Read/write performance**

# S3A+ file system

- Cache metadata
- Leverage S3 multipart API
- Prefetch data for reads

# Metadata cache in MySQL

| Path | Is dir | Is empty | Length |
|---|---|---|---|
| | | | |
| s3a://bucket/foo/bar/baz/data/a0 | 0 | 0 | 100 |
| | | | |
| s3a://bucket/foo/bar/baz/data/a2 | 0 | 0 | 300 |

# Metadata cache in MySQL

| Path | Is dir | Is empty | Length |
|---|---|---|---|
| | | | |
| s3a://bucket/foo/bar/baz/data/a0 | 0 | 0 | 100 |
| | | | |
| s3a://bucket/foo/bar/baz/data/a2 | 0 | 0 | 300 |

30x Speed Up

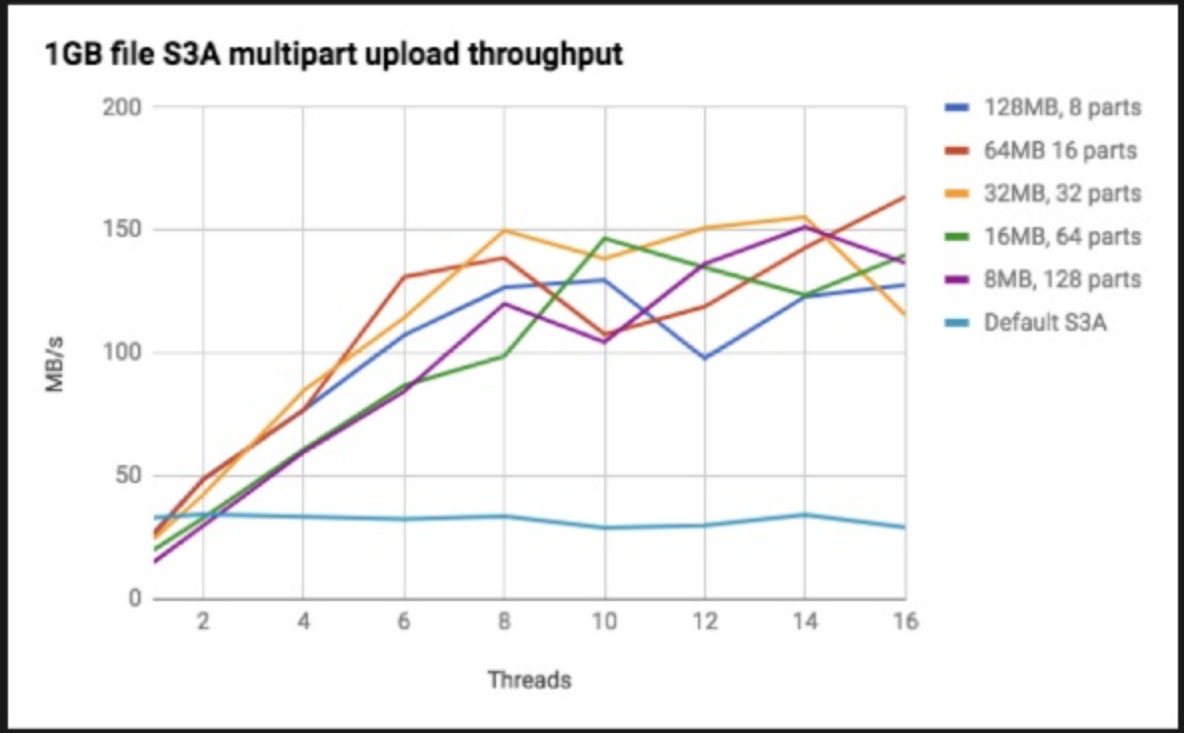
# S3 multipart API

- Improved throughput
- Quick recovery from any network issues
- Pause and resume object uploads
- Begin an upload before you know the final object size

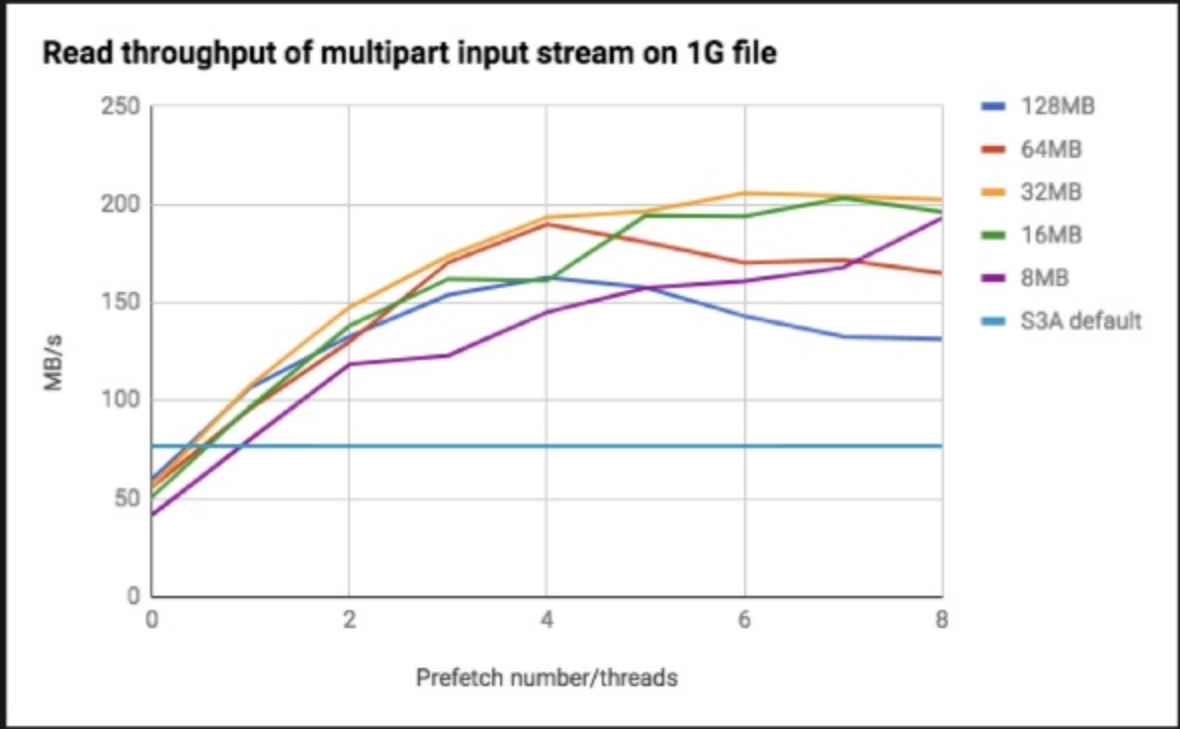
# S3 multipart API

- **Improved throughput**
- Quick recovery from any network issues
- Pause and resume object uploads
- Begin an upload before you know the final object size

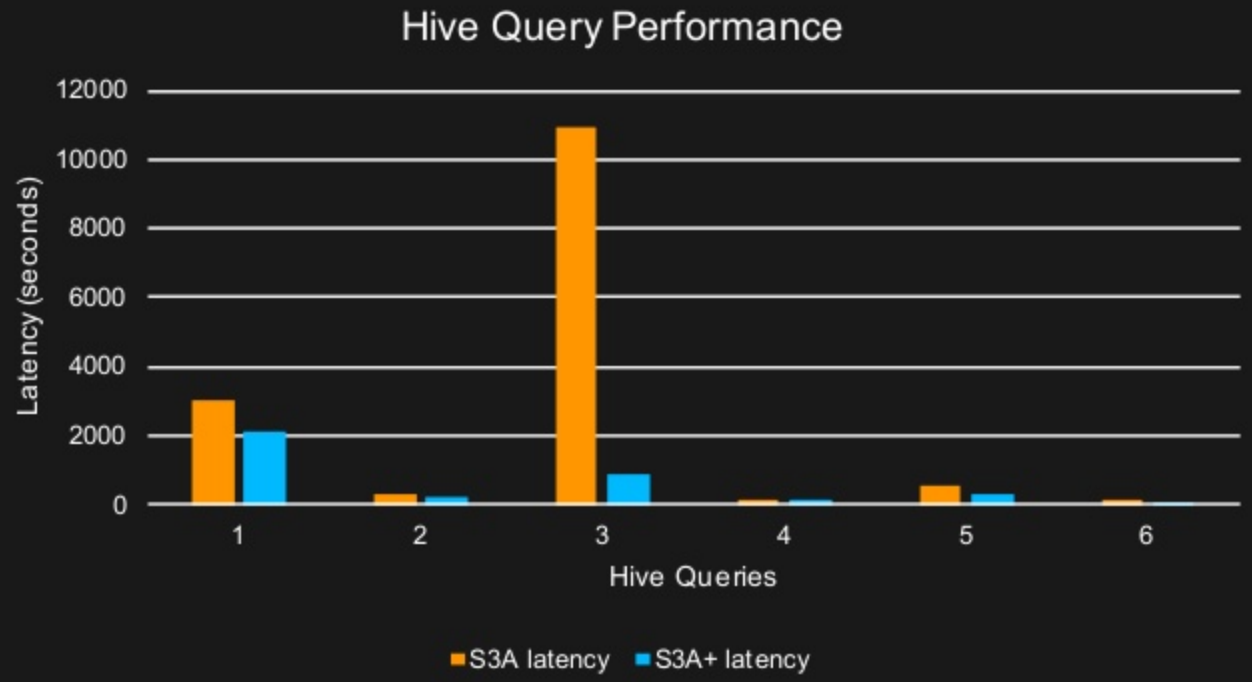
# S3A multipart API



# Read prefetch



# Performance



# Putting it all together

# To summarize

- ✓ Always store a copy of the raw input
- ✓ Use automation with S3 events to enable trigger-based workflows
- ✓ Implement the right security controls
- ✓ Use a format that supports your data, rather than forcing your data into the format
- ✓ Partition data to improve performance
- ✓ Apply compression to lower network load and cost

# New storage training

aws training and certification

For Enterprise Storage Engineers

- Learn how to architect and manage highly available solutions on AWS storage services
- Advance toward AWS certifications
- Help your organization migrate to the cloud faster

Online at www.aws.training

- Access 100+ new digital training courses including advanced training on storage
- Deep dives on Amazon S3, EFS, and EBS
- Migrating and tiering storage to AWS (hybrid solutions)

At re:Invent

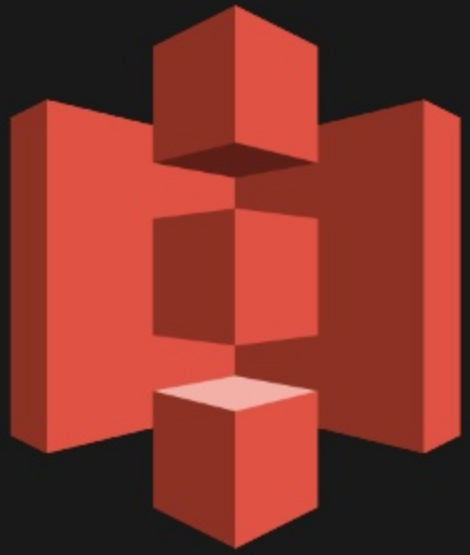
- Visit Hands-on Labs at the Venetian
- Attend a proctored “Introduction to EFS” Spotlight Lab on Thursday at 3pm at the Venetian
- Meet storage experts at the Ask the Experts in Hands-on Labs room at the Venetian

AWS re:Invent



aws

# Q&A



Amazon S3

Amazon Glacier


AWS re:Invent



aws