AWS re:INVENT

Big Data: Data Lakes and Data Oceans

Lex Crosett, Principal Solutions Architect
Sajee Mathew, Principal Solutions Architect
Erick Dame, Enterprise Solutions Architect
John Mallory, Business Development Manager

November 28, 2017

AWS re:Invent

aws

What to Expect from the Workshop

- Data lake and analytics review (30 min.)
- Set up a data lake using an AWS solution (30 min.)
- Add information to the data lake (30 min.)
- Perform lightweight analysis with AWS big data tools (1 hour)

Introduction to Data Lake Concepts

Unlocking Data

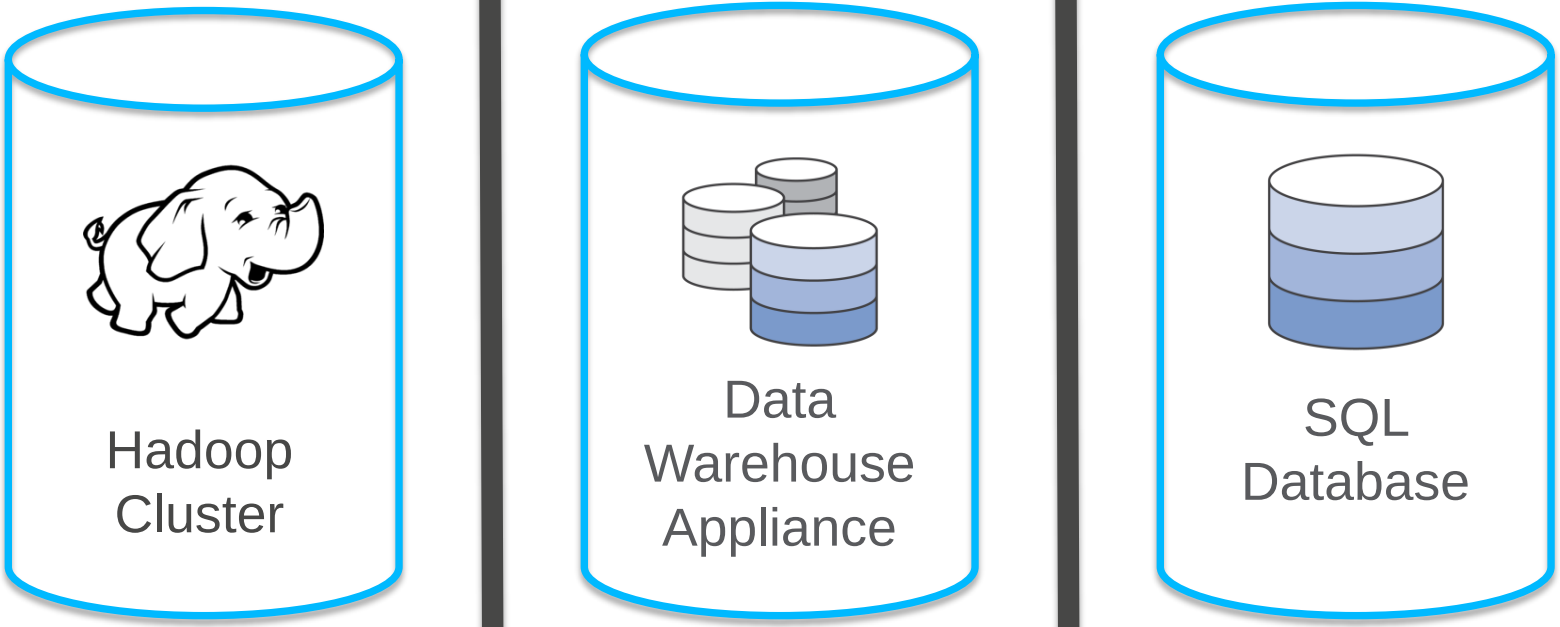
Most companies and organizations are embarking on ambitious innovation initiatives to unlock their data

The data already exists but goes unused or is locked away from complimentary data sets in isolated data silos

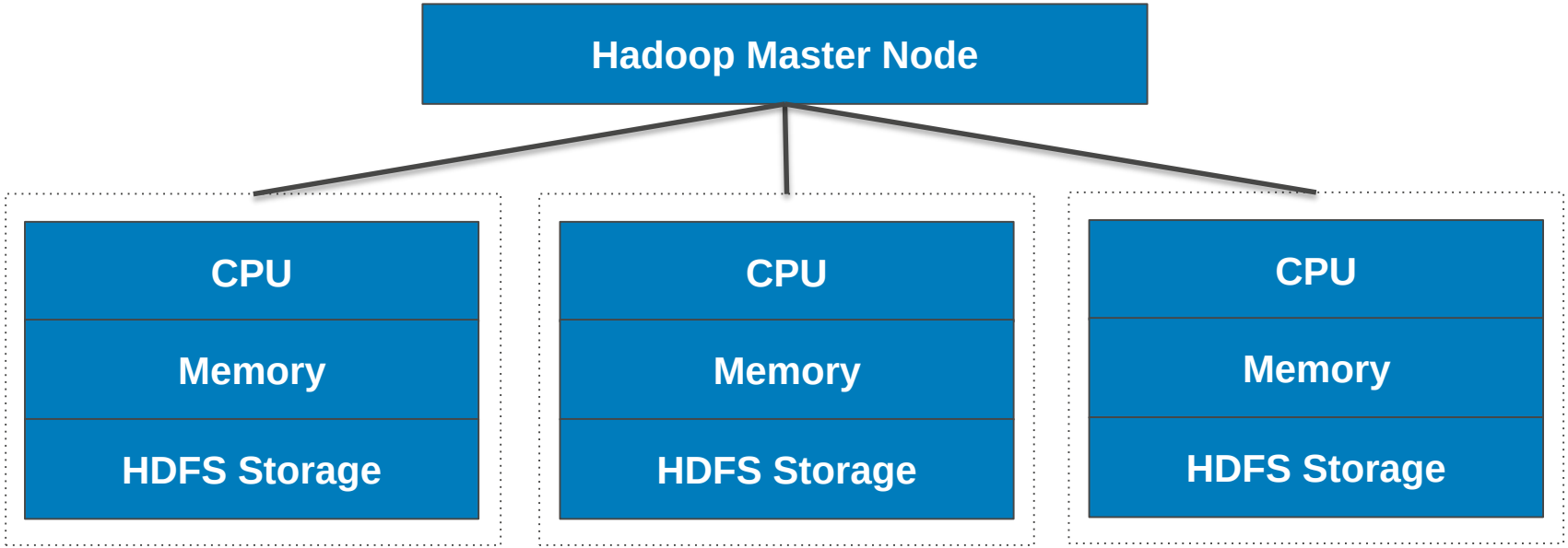
Challenges with Legacy Data Architectures

- Can’t move data across silos
- Can’t deal with dynamic data and real-time processing
- Can’t deal with format diversity and change rate
- Complex ETL processes
- Difficult to find people with adequate skills to configure and manage these systems
- Can’t integrate with the explosion of available social and behavior tracking data

Legacy Data Architectures Exist as Isolated Data Silos



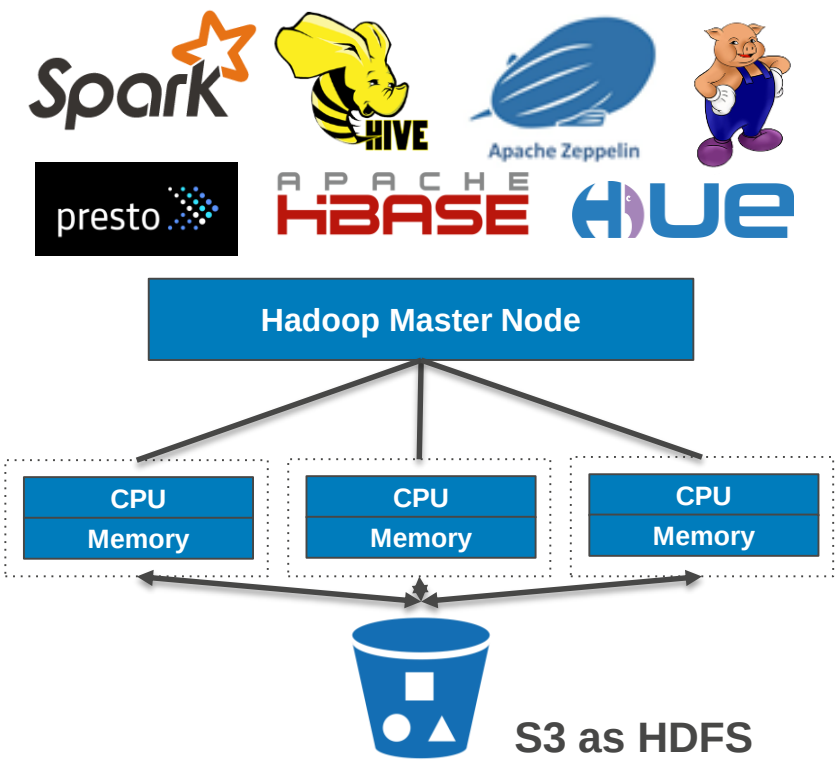
Legacy Data Architectures Are Monolithic



Multiple layers of functionality all on a single cluster

Evolution of Data Architectures

2009: Decoupled EMR architecture



Improvements

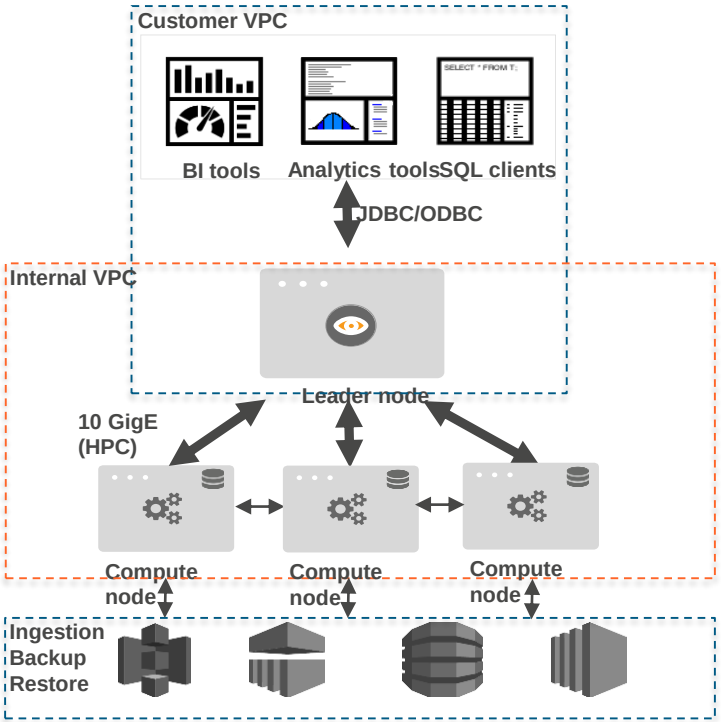
- Decoupled storage and compute
- Scale CPU and memory resources independently and up and down
- Only pay for the 500 TB dataset (not 3X)
- Multi-physical facility replication via Amazon S3
- Multiple clusters can run in parallel against shared data in Amazon S3
- Each job gets its own optimized cluster. For example, Spark on memory intensive, Hive on CPU intensive, HBase on I/O intensive, and so on

Constraints

- Still have a cluster to provision and manage
- Must expose EMR cluster to SQL users via Hive, Presto, and so on

Evolution of Data Architectures

2012: Amazon Redshift—cloud DW



Improvements

- Automated installation, patching, backups
- No servers to manage and maintain
- MPP columnar relational database
- $1,000/TB/year
- Accessible to any ODBC or JDBC BI Tool

Constraints

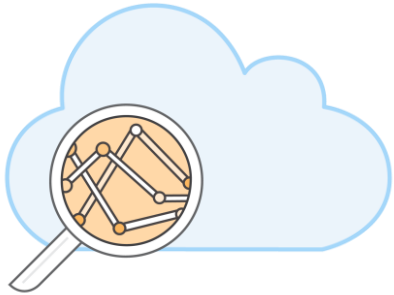
- Still have to load data into a schema

Enter Data Lake Architectures

Data lake is a new and increasingly popular architecture to store and analyze massive volumes and heterogeneous types of data

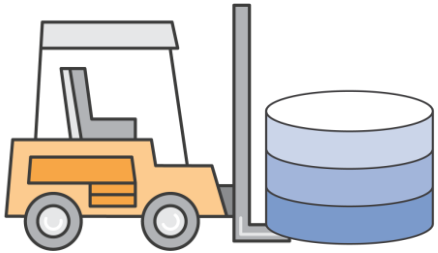
Benefits of a Data Lake—All Data is in One Place

“Why is the data distributed in many locations? Where is the single source of truth?”

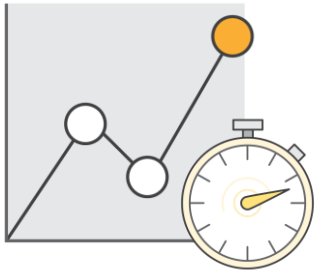


Analyze all of your data, from all of your sources, in one stored location

Benefits of a Data Lake—Quick Ingest

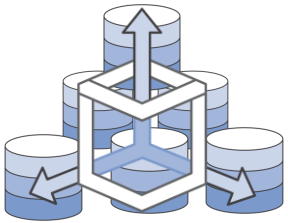


“How can I collect data quickly from various sources and store it efficiently?”

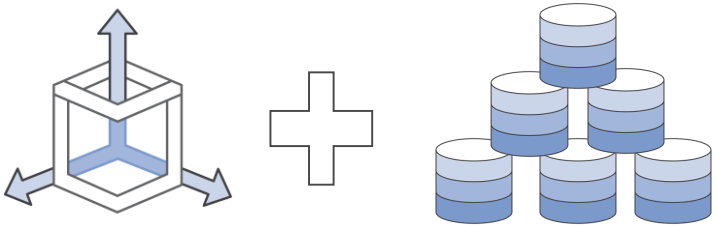


Quickly ingest data without needing to force it into a predefined schema

Benefits of a Data Lake—Storage vs. Compute

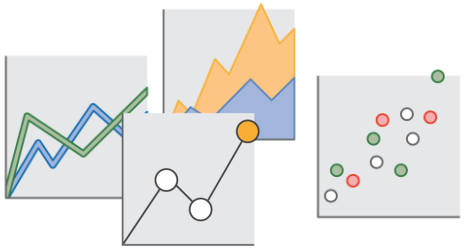


"How can I scale up with the volume of data being generated?"



Separating your storage and compute allows you to scale each component as required

Benefits of a Data Lake—Schema on Read



“Is there a way I can apply multiple analytics and processing frameworks to the same data?”

A data lake enables ad-hoc analysis by applying schemas on read, not write

AWS Approach to Data Lake

Amazon S3 is the Data Lake

Why Amazon S3 for a Data Lake?

Durable

Designed for 11 9s of durability

Available

Designed for **99.99**% availability

High performance

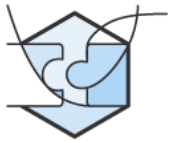
- Multiple upload
- Range GET
- Scalable throughput

Easy to use

- Simple REST API
- AWS SDKs
- Simple management tools
- Event notification
- Lifecycle policies

Scalable

- Store as much as you need
- Scale storage and compute independently
- No minimum usage commitments



Integrated

- Amazon EMR
- Amazon Redshift Spectrum
- Amazon DynamoDB
- Amazon Athena
- AWS Glue
- Amazon Rekognition
- Amazon Macie

AWS re:Invent

Benefits of an AWS Amazon S3 Data Lake

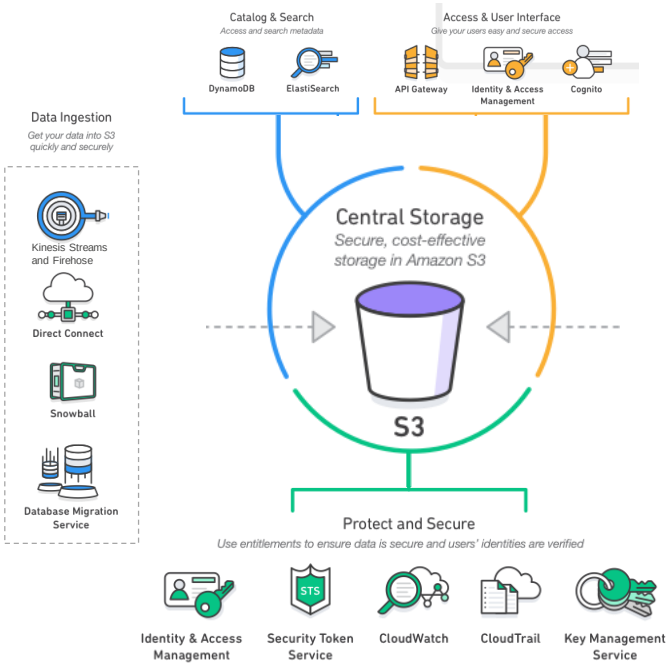
Fixed cluster data lake

- Limited to only the single tool contained on the cluster (for example, Hadoop or data warehouse or Cassandra). Use cases and ecosystem tools change rapidly.
- Expensive to add nodes to add storage capacity
- Expensive to replicate data against node loss
- Complexity in scaling local storage capacity
- Long refresh cycles to add additional storage equipment

AWS Amazon S3 data lake

- Decouple storage and compute by making S3 object based storage, not a fixed tool to manage the data lake
- Flexibility to use any and all tools in the ecosystem. ***The right tool for the job.***
- Catalog, transform, and query in place
- Future-proof your architecture. As new use cases and tools emerge you can plug and play current best of breed.

Building a Data Lake on AWS



Processing and Analytics

Real-time

- Elasticsearch Service
- Kinesis Analytics, Kinesis Streams
- Spark Streaming on EMR
- Apache Flink on EMR
- AWS Lambda
- Apache Storm on EMR

Batch

- EMR Hadoop, Spark, Presto
- Amazon Athena Query Service
- Amazon Redshift Data Warehouse

AI and Predictive

- Amazon Lex Speech recognition
- Amazon Polly Text to speech
- Amazon Rekognition
- Machine Learning Predictive analytics

Transactional and RDBMS

- DynamoDB NoSQL DB
- Aurora Relational Database

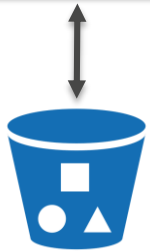
BI and Data Visualization

- Amazon QuickSight
- tableau
- MicroStrategy
- kibana

Further Evolution of Data Lake Architectures

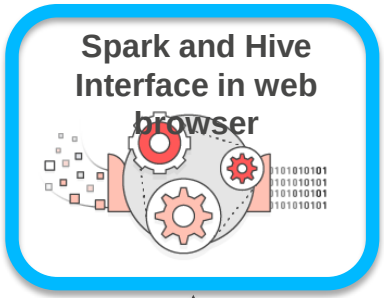
Today: clusterless

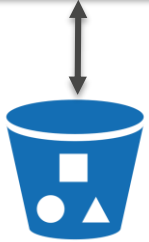
Amazon Athena for SQL



S3 Data Lake

AWS Glue for ETL





S3 Data Lake

Improvements

- No cluster/infrastructure to manage
- Business users and analysts can write SQL without having to provision a cluster or touch infrastructure
- Pay by the query
- Zero administration
- Process data where it lives

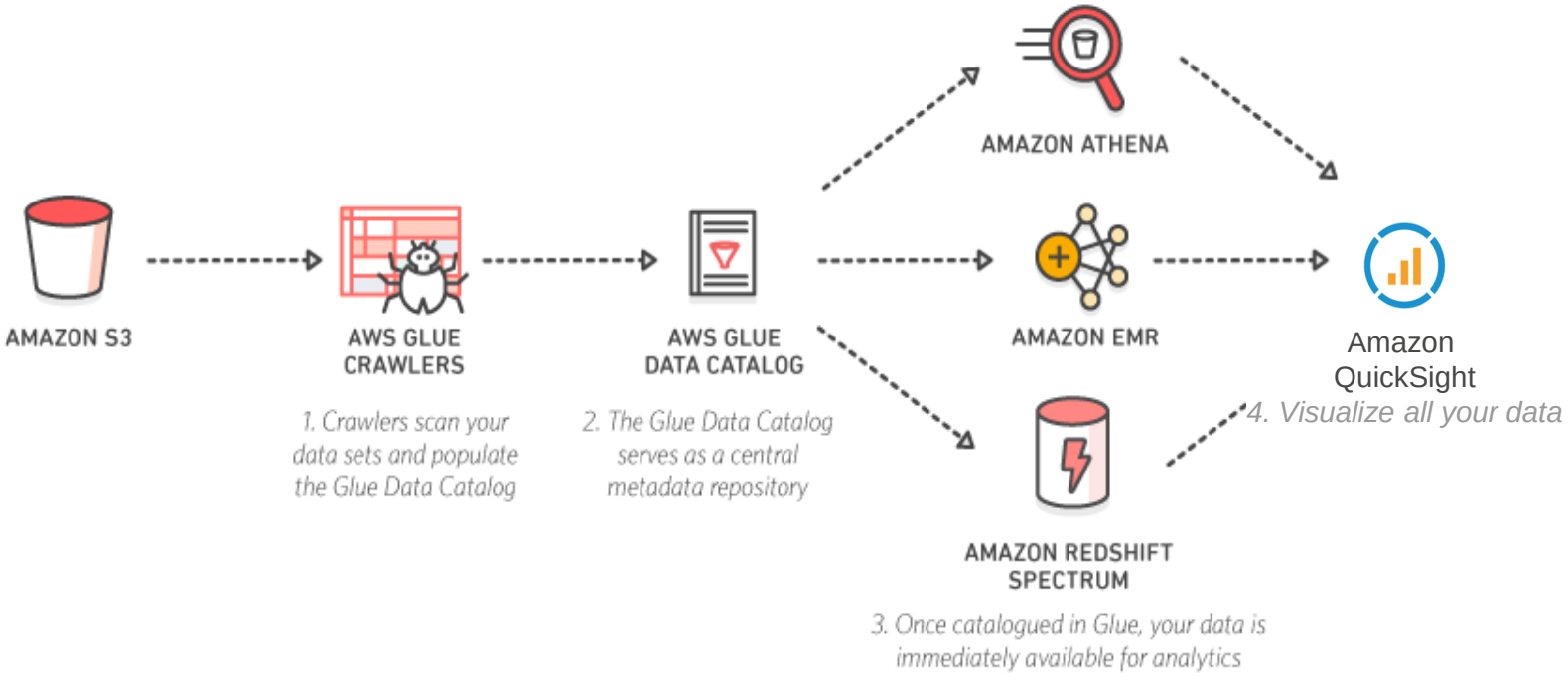
Constraints

- Limited to SQL, Hive, and Spark jobs today
- More frameworks to come!

Open Data Formats—Free Your Data!

- Parquet, ORC, Avro, JSON, CSV, others
- Allows multiple analytic tools to operate on same data
- Proprietary data = countdown to next migration

Use an Optimal Combination of Highly Interoperable Se



Summary of AWS Analytics, Database, and AI Tools

Amazon Redshift
Enterprise Data Warehouse

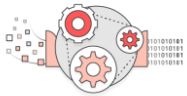
Amazon Elasticsearch Service
Elasticsearch

Amazon EMR
Hadoop/Spark

Amazon DynamoDB
Managed NoSQL Database

Amazon Athena
Clusterless SQL

Amazon ElastiCache
Redis In-memory Datastore



AWS Glue
Clusterless ETL

Amazon QuickSight
Business Intelligence/Visualization

Amazon Aurora
Managed Relational Database

Amazon Rekognition
Deep Learning-based Image Recognition

Amazon Machine Learning
Predictive Analytics

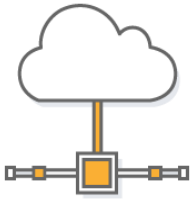
Amazon Lex
Voice or Text Chatbots

AWS re:Invent

aws

AWS Innovations to Produce More Efficient Data Lake Architectures

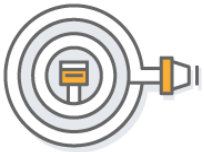
Data Ingestion into Amazon S3



AWS Direct Connect

AWS Snowball

ISV Connectors



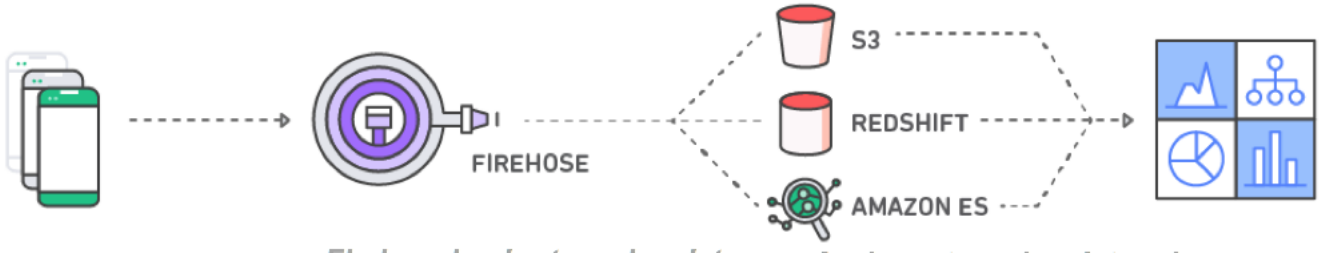
Amazon Kinesis Firehose

Amazon S3 Transfer Acceleration

AWS Storage Gateway

Amazon Kinesis Firehose

Load massive volumes of streaming data into Amazon S3, Amazon Redshift. and Amazon Elasticsearch



Capture and submit streaming data to Firehose

Firehose loads streaming data continuously into Amazon S3, Amazon Redshift, and Amazon Elasticsearch

Analyze streaming data using your favorite BI tools

Zero administration: capture and deliver streaming data into Amazon S3, Amazon Redshift, and Amazon Elasticsearch **without writing an application or managing infrastructure**

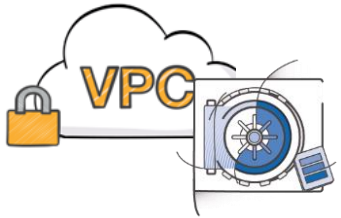
Direct-to-data store integration: **batch, compress**, and **encrypt** streaming data for delivery into data destinations **in as little as 60 seconds** using simple configurations

Seamless elasticity: seamlessly scales to match data throughput without intervention

Implement the Right Cloud Security Controls

Encryption

- SSL endpoints
- Server-side encryption (SSE-S3)
- S3 server-side encryption with provided keys (SSE-C, SSE-KMS)
- Client-side encryption



Security

- Identity and access Management (IAM) policies
- Bucket policies
- Access Control Lists (ACLs)
- Private VPC endpoints to Amazon S3
- Amazon S3 object tagging to manage access policies

Compliance

- Buckets access logs
- Lifecycle management policies
- Access Control Lists (ACLs)
- Versioning and MFA deletes
- Certifications—HIPAA, PCI, SOC 1/2/3, etc.

Data Lake Best Practices

- Use Amazon S3 as the storage repository for your data lake, instead of a Hadoop cluster or data warehouse
- Decoupled storage and compute is cheaper and more efficient to operate
- Decoupled storage and compute allow us to evolve to clusterless architectures like Amazon Athena and AWS Glue
- Do not build data silos in Hadoop or the Enterprise DW
- Use granular encryption, roles, and access controls to build a secure, multi-tenant centralized data platform
- Gain flexibility to use all the analytics tools in the ecosystem around Amazon S3 and future-proof the architecture

Workshop Process

- Log in to your AWS account
- Apply account credit
- Open the workshop guide
- Find and run the Data Lake Foundation on AWS
- Once complete, load the Behavioral Risk Factor Surveillance system (BRFSS) data into Amazon S3 following the workshop instructions
- Follow the analysis steps in the workshop guide
- Run Delete Stack to remove resources when done

Questions?