SUMMIT ONLINE

Best practices for implementing a data lake in Amazon S3

Kumar Nachiketa

Senior Partner Solutions Architect

Amazon Web Services

AWS SUMMIT ONLINE

Agenda

Data at scale and data lakes

Data lake foundation best practices

Data lake performance & security best practices

AWS Lake Formation demo

Data at scale and data lakes

60 seconds in Viber

3M messages sent

140,000 calls

1.2M users log in

2,000 new users join

300,000 stickers sent

10,000 group chat likes

1.5M photos sent

A day in FINRA

75 billion market events

50,000 files daily

200 rules to format files

Half a trillion validations each day

Data at scale

Growing exponentially

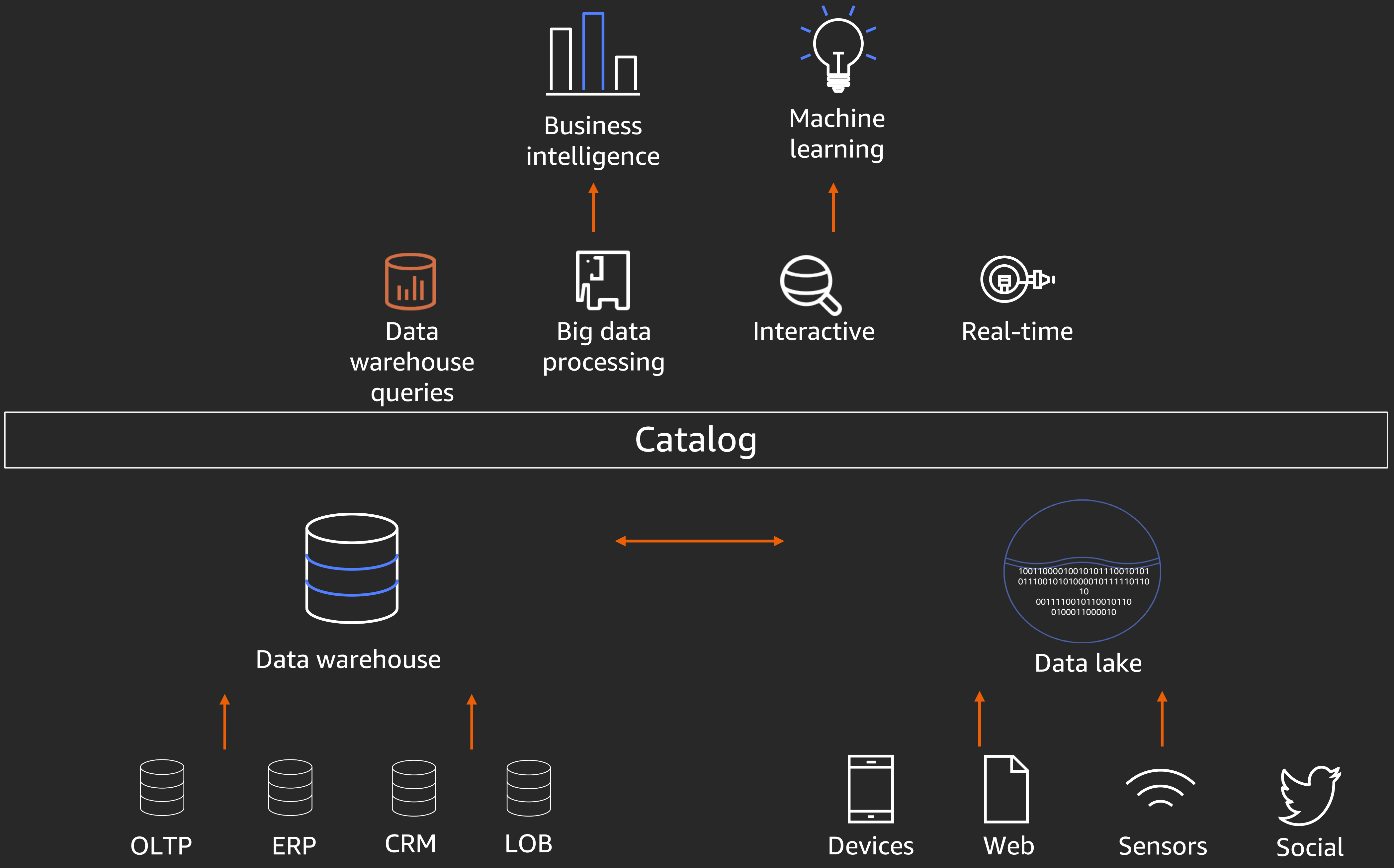
From new sources

Increasingly diverse

Used by many people

Analyzed by many applications

Defining the data lake



Business intelligence

Machine learning

Data warehouse queries

Big data processing

Interactive

Real-time

Catalog

Data warehouse

Data lake

OLTP

ERP

CRM

LOB

Devices

Web

Sensors

Social

Amazon S3 is the foundation of any data lake

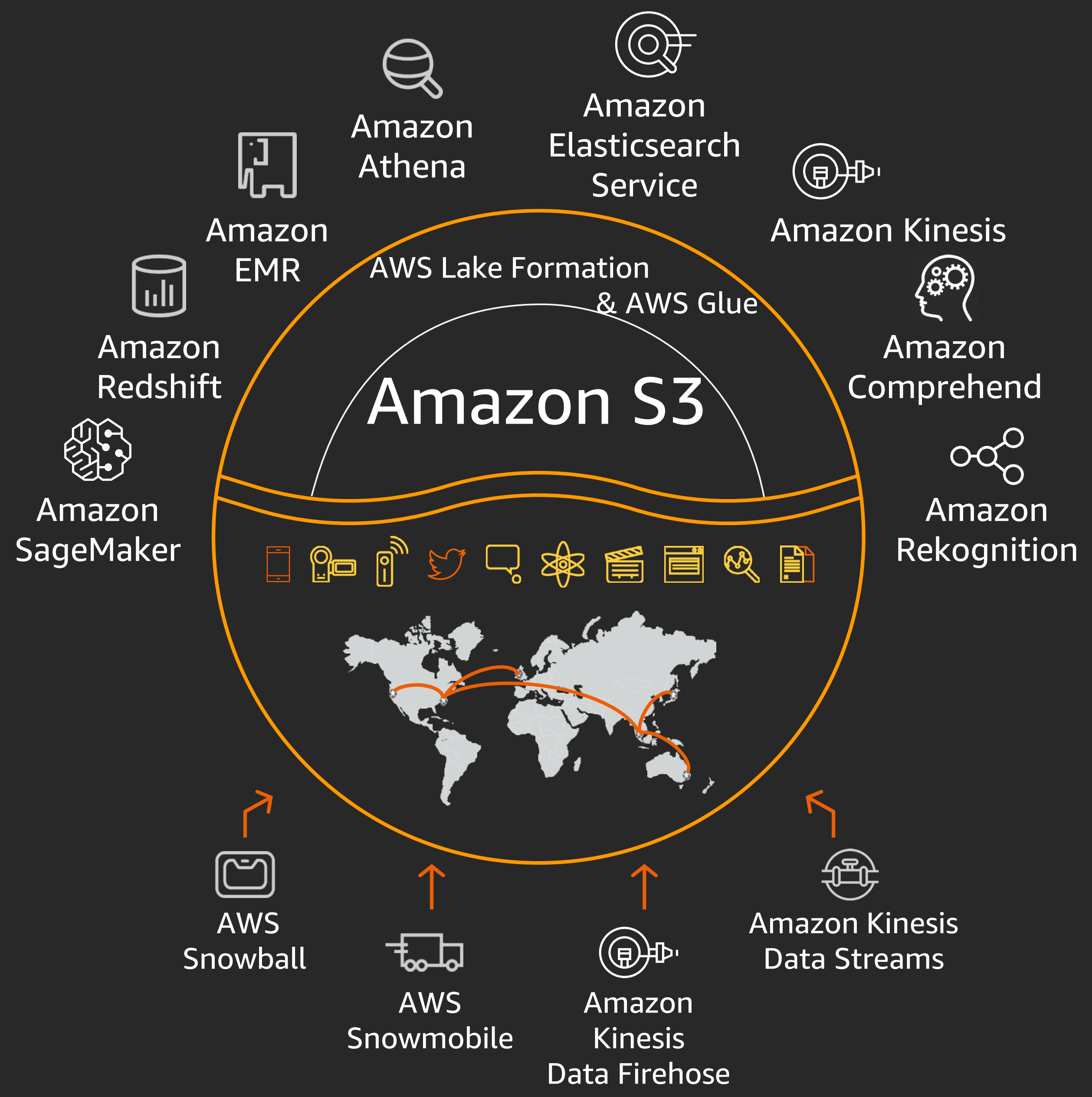
Multiple data input sources

Storage scales on demand

Supports many unique users and teams

Analyzed by many applications

Amazon S3 as the foundation for data lakes



Durable, available, exabyte-scalable

Secure, compliant, auditable

High performance

Low-cost storage and analytics

Broad network integration

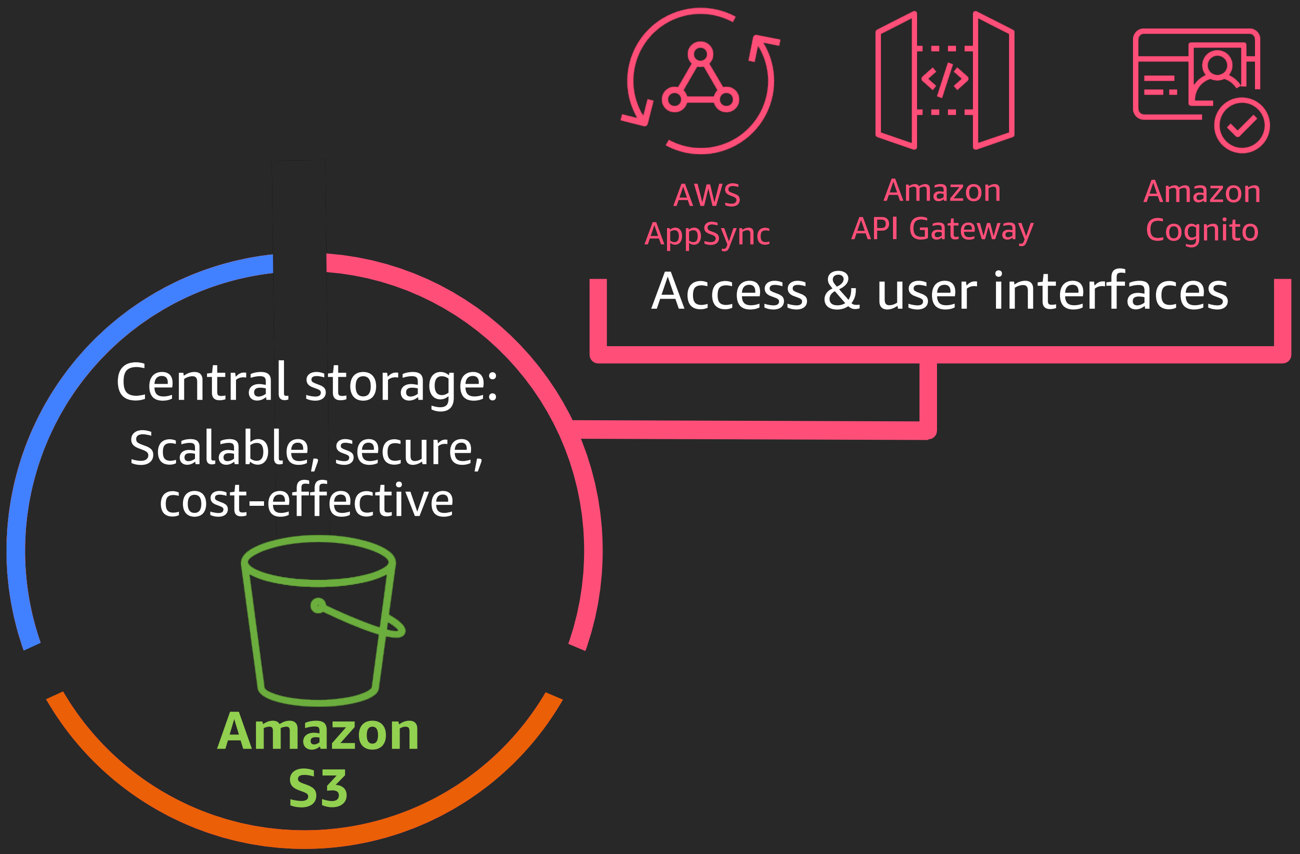
Data lake foundation best practices

Data lake on AWS – S3 at the core

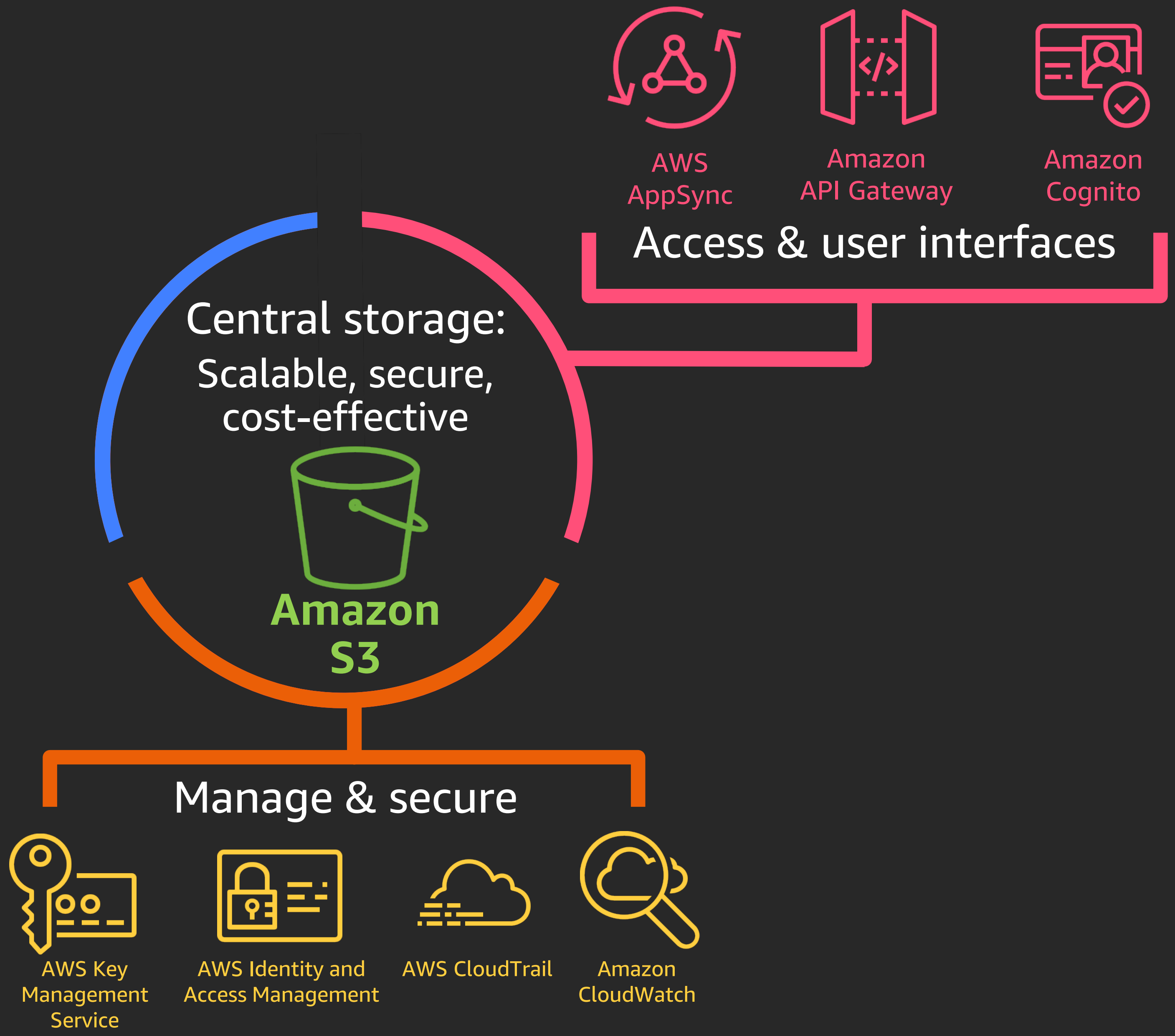
Central storage:

Scalable, secure, cost-effective

Data lake on AWS – access and user interface



Data lake on AWS – manage and secure



Central storage:
Scalable, secure, cost-effective

Amazon S3

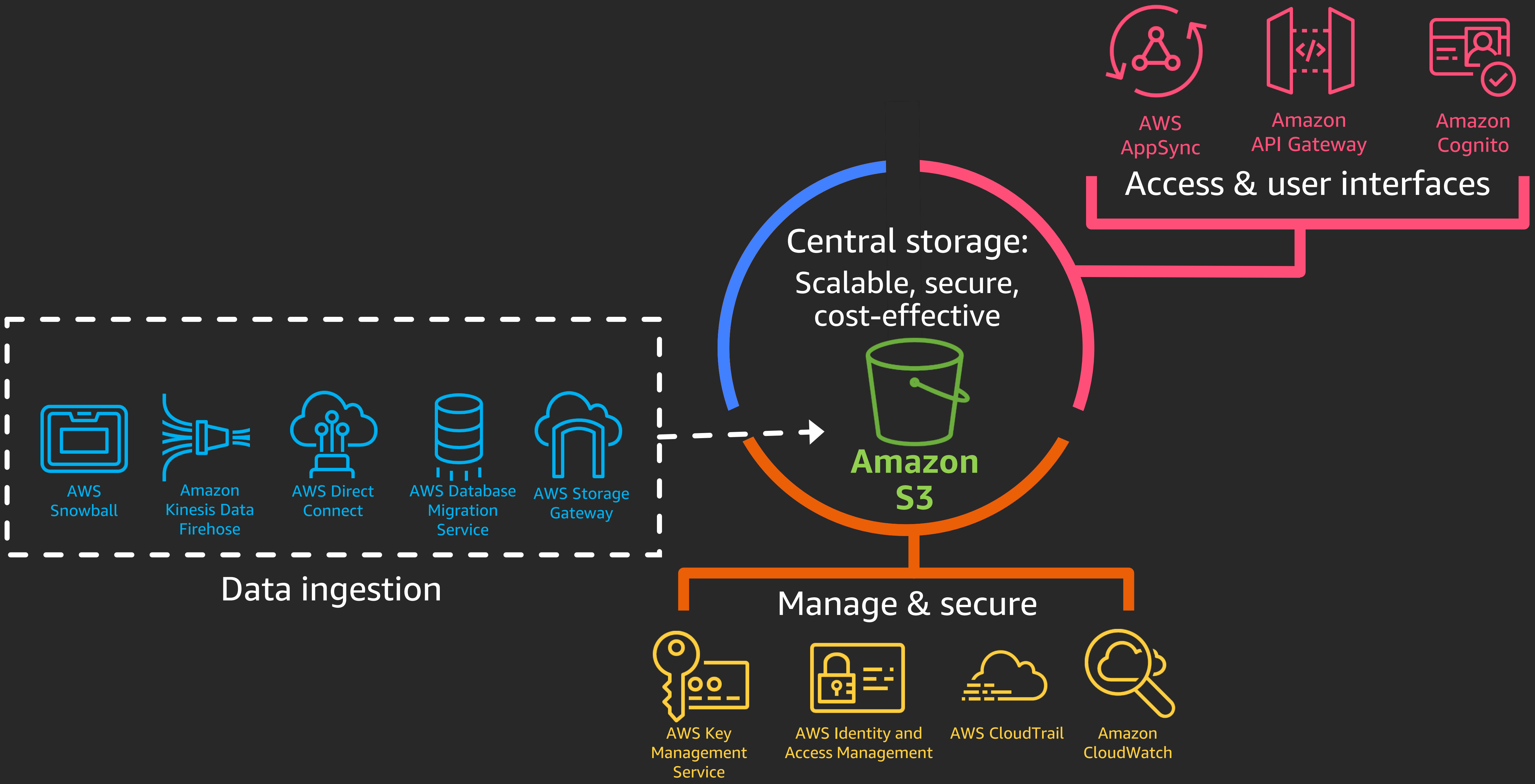
Access & user interfaces

- AWS AppSync
- Amazon API Gateway
- Amazon Cognito

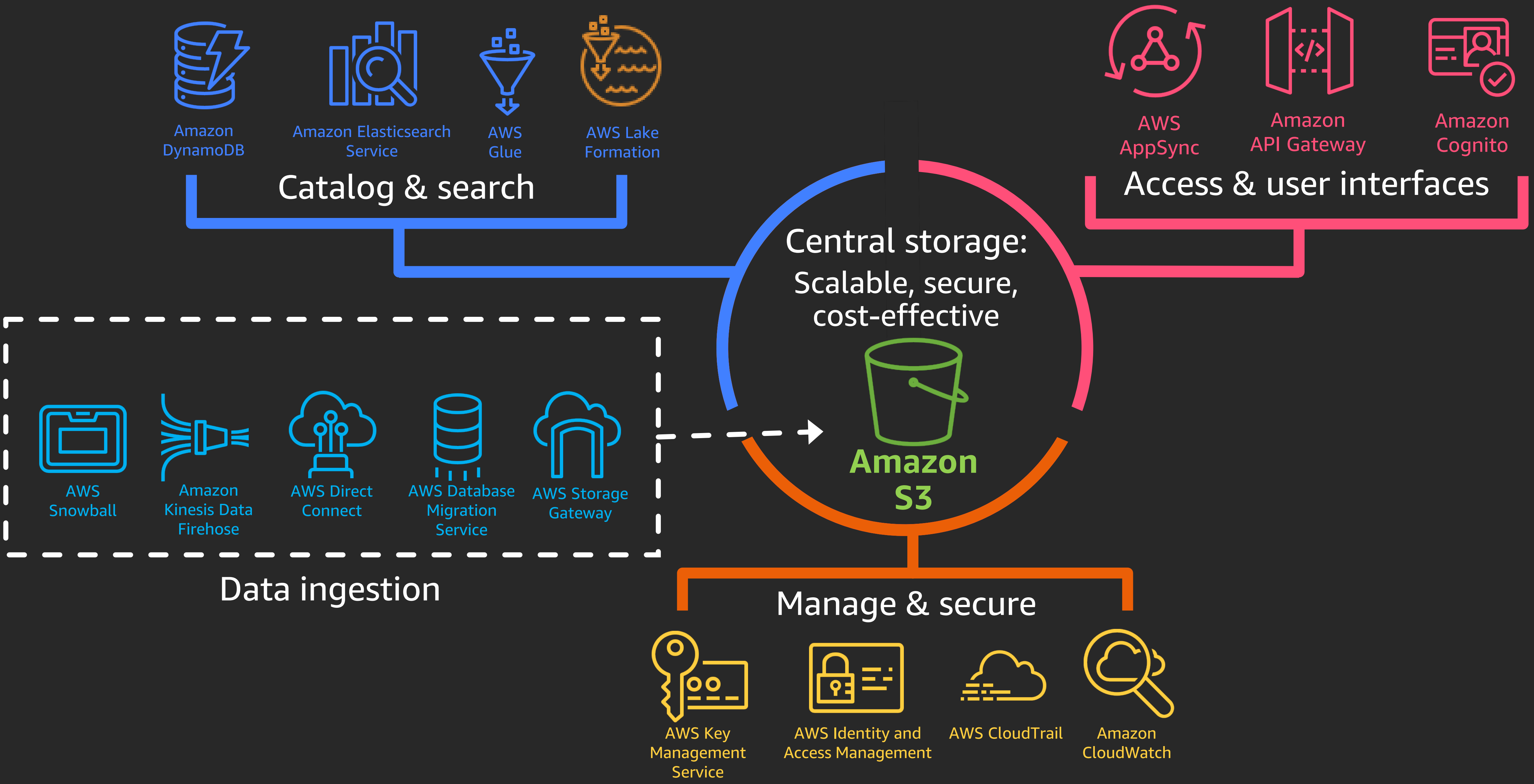
Manage & secure

- AWS Key Management Service
- AWS Identity and Access Management
- AWS CloudTrail
- Amazon CloudWatch

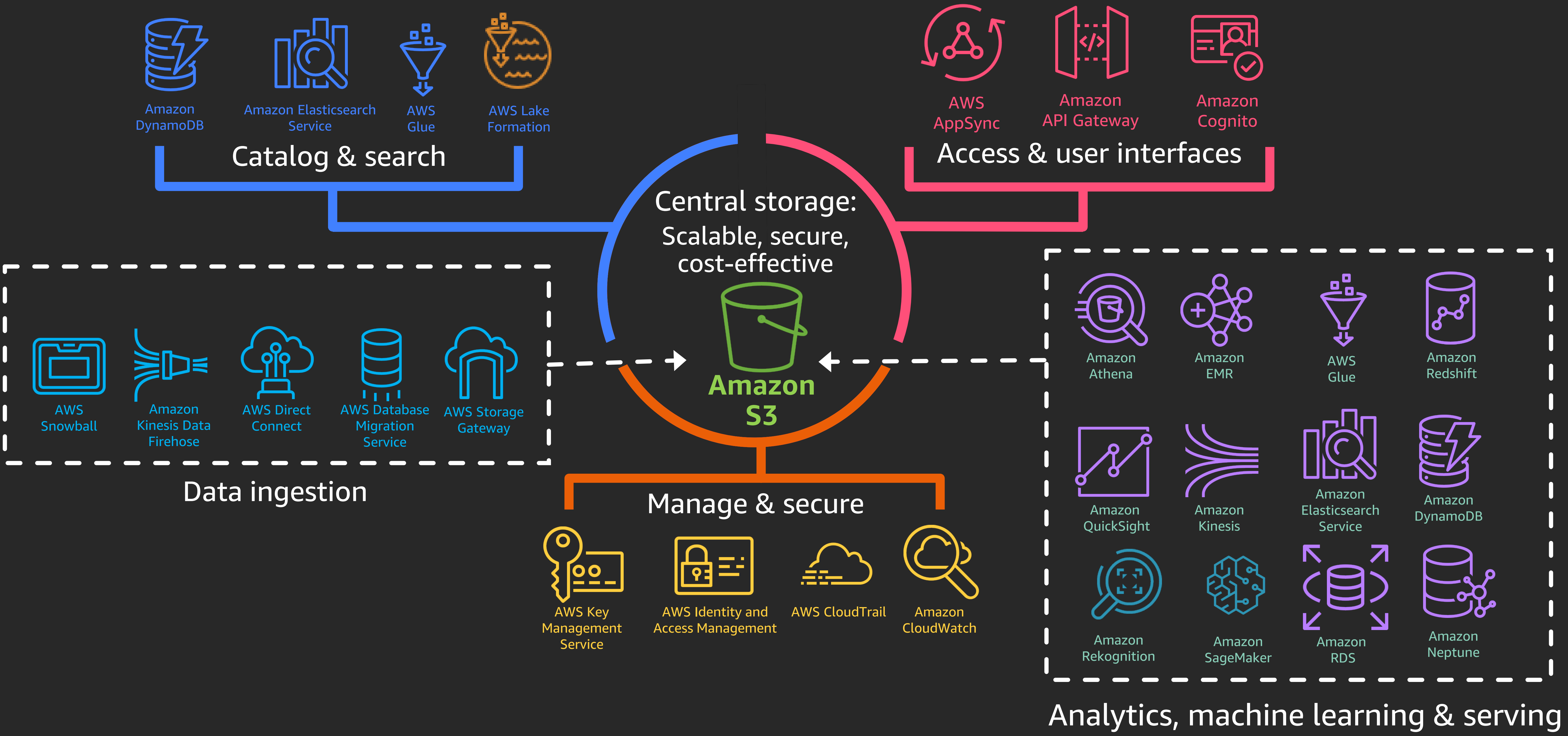
Data lake on AWS – data ingestion



Data lake on AWS – catalog and search



Data lake on AWS – analytics, ML, and serving



Amazon DynamoDB

Amazon Elasticsearch Service

AWS Glue

AWS Lake Formation

Catalog & search

AWS AppSync

Amazon API Gateway

Amazon Cognito

Access & user interfaces

Central storage:

Scalable, secure, cost-effective

Amazon S3

AWS Snowball

Amazon Kinesis Data Firehose

AWS Direct Connect

AWS Database Migration Service

AWS Storage Gateway

Data ingestion

Manage & secure

AWS Key Management Service

AWS Identity and Access Management

AWS CloudTrail

Amazon CloudWatch

Amazon Athena

Amazon EMR

AWS Glue

Amazon Redshift

Amazon QuickSight

Amazon Kinesis

Amazon Elasticsearch Service

Amazon DynamoDB

Amazon Rekognition

Amazon SageMaker

Amazon RDS

Amazon Neptune

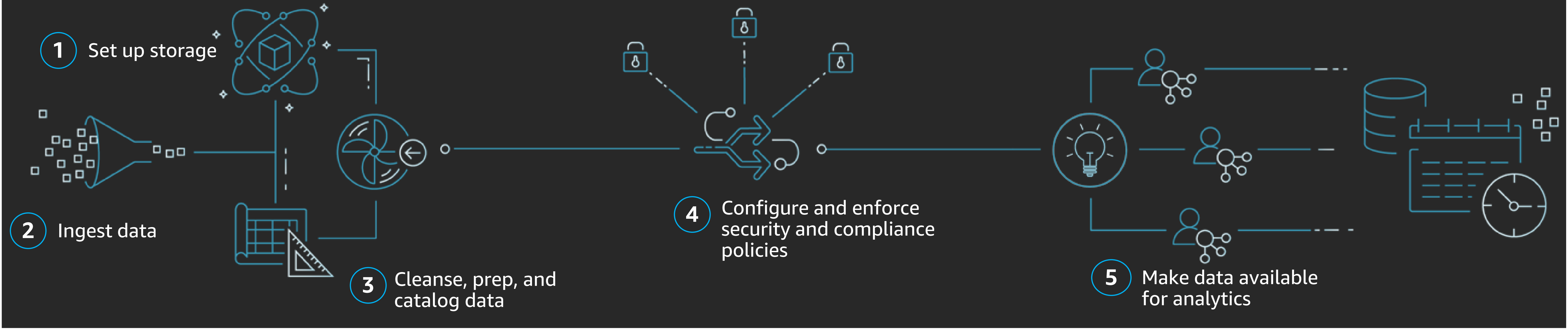
Analytics, machine learning & serving

Data lake design pattern example

aws SUMMIT ONLINE

Data lake workflow pattern

Typical steps in building a data lake



1 Set up storage

2 Ingest data

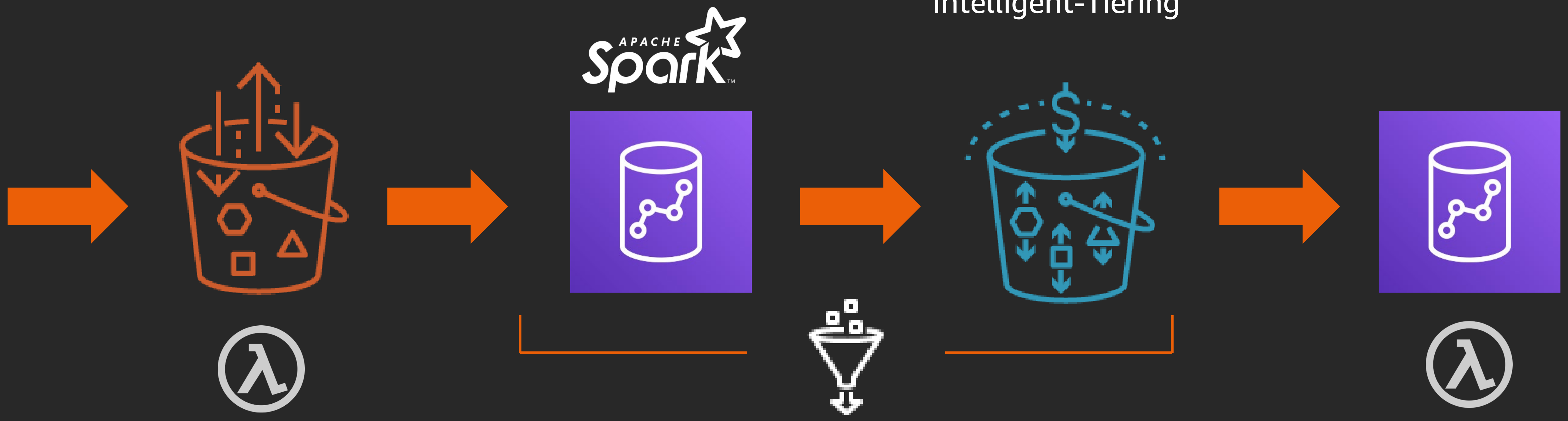
3 Cleanse, prep, and catalog data

4 Configure and enforce security and compliance policies

5 Make data available for analytics

Data lake ingest and transform patterns

Pipelined architectures improve governance, data management, and efficiency



Raw data
Amazon S3
Standard

ETL
AWS Glue or Amazon EMR

Production data (data lake)
Amazon S3
Intelligent-Tiering

Data warehouse
Amazon Redshift

Triggered code
AWS Lambda

ETL and catalog management
AWS Glue and AWS Lake Formation

Triggered code
AWS Lambda

Viber: Processing events on a data lake

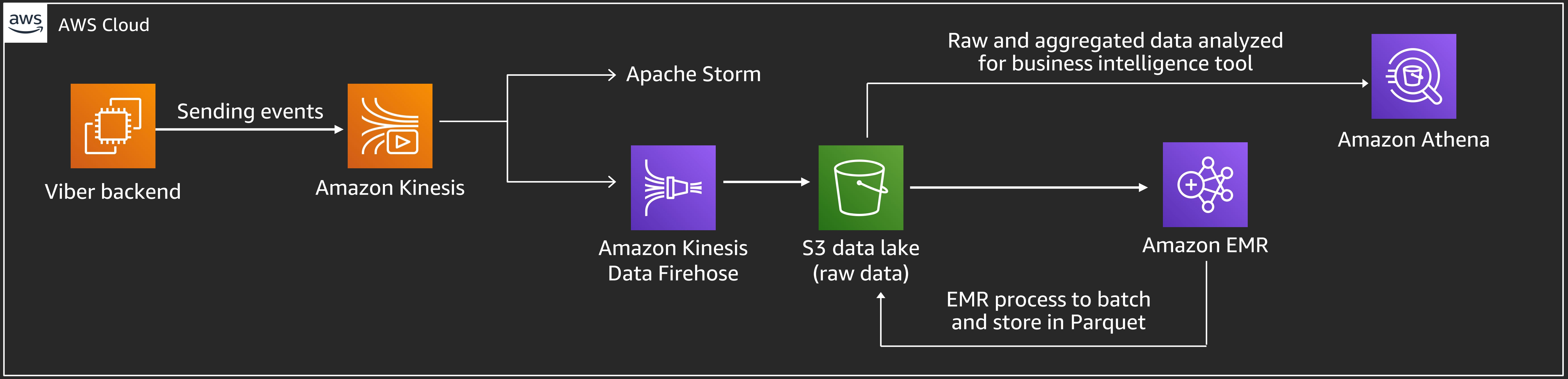
Workload: Communications platform serving a billion users worldwide

- Processing over 10–15 billion events per day
- Peaking at 300,000 events per second
- Storing many petabytes of data
- Running over 200 events on a single Amazon Kinesis stream

Viber: Processing events on a data lake

Workload: Communications platform serving a billion users worldwide

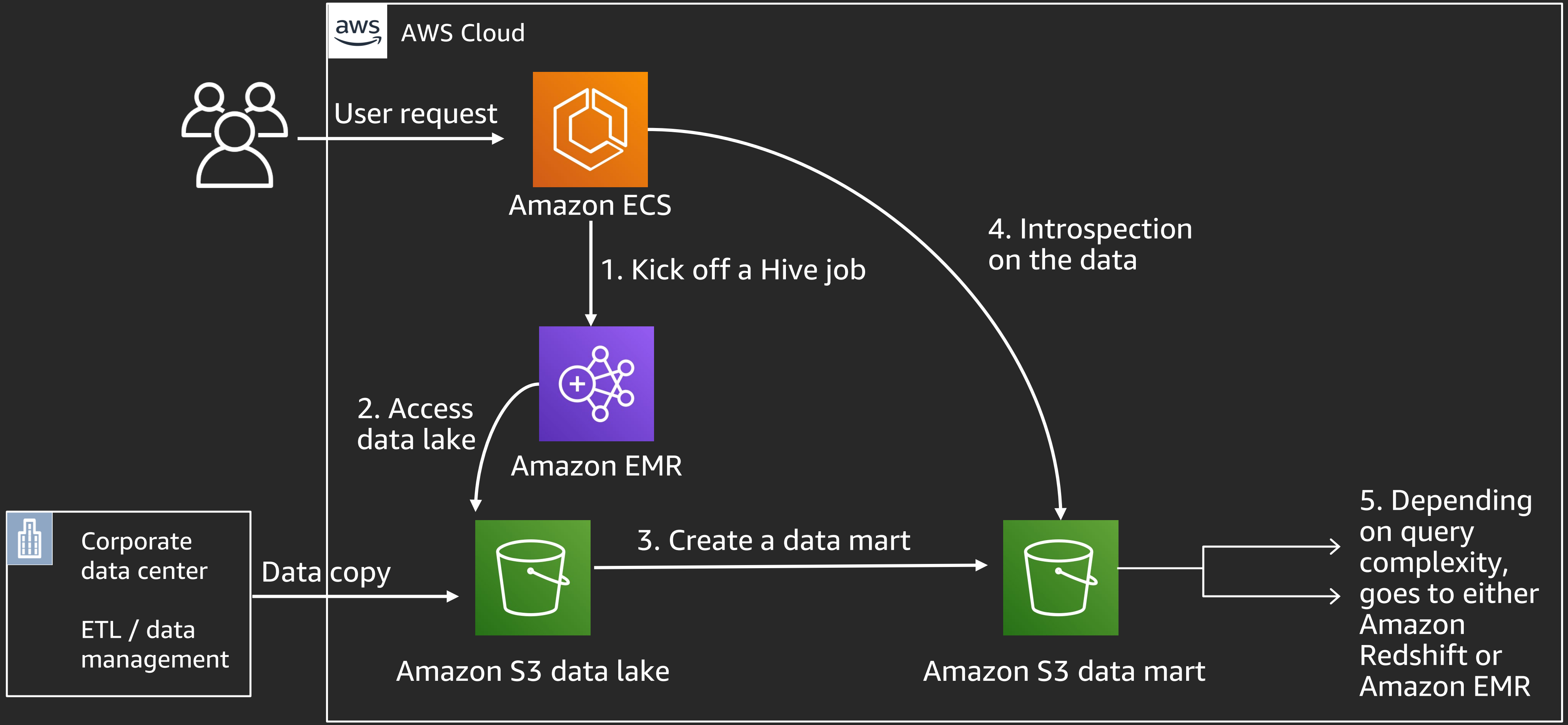
- Processing over 10–15 billion events per day
- Peaking at 300,000 events per second
- Storing many petabytes of data
- Running over 200 events on a single Amazon Kinesis stream


AWS Cloud
Viber backend
Sending events
Amazon Kinesis
Apache Storm
Amazon Kinesis Data Firehose
S3 data lake (raw data)
Raw and aggregated data analyzed for business intelligence tool
Amazon Athena
Amazon EMR
EMR process to batch and store in Parquet

FINRA: Petabyte-scale data analysis

Workload: Financial regulatory authority providing users with ability to access PBs of data for analytics

Architecting on AWS for scale



"With the new architecture, we uncovered more needs of the end user. This led us to move from one large Amazon Redshift cluster to a blend of querying engines."

Running analytics on AWS data lakes

	Lift-and-shift	AWS Managed Services
	cloudera databricks Qubole Informatica	Amazon Redshift AWS Glue Amazon EMR Amazon Athena
What	• Run third-party analytics tools on Amazon EC2 • Use Amazon EBS and Amazon S3 as data stores • Self-managed environments	• AWS managed and serverless platforms • AWS Glue, Amazon Athena, Amazon EMR, Amazon Redshift • More options to process data in place
Why	• Simplify on-premises migrations • Use existing tools, code, and customizations • Minimize application changes	• Focus on data outcomes, not infrastructure • Speed adoption of new capabilities • More tightly integrated with AWS security
Consider	• You provision, manage, and scale • You monitor and manage availability • You own upgrades and versioning	• Utilizing AWS Lake Formation • Flexibility and choice with open data formats • Leverage AWS pace of innovation

Amazon S3 is the storage foundation for both approaches

Amazon S3: Data management, performance, and security

Choosing the right data lake storage class

Select storage class by data pipeline stage

Raw data

Amazon S3 Standard

- Small log files
- Overwrites if synced
- Short-lived
- Moved & deleted
- Batched & archived

ETL

Amazon S3 Standard

- Data churn
- Small intermediates
- Multiple transforms
- Deletes <30 days
- Output to data lake

Production data lake

Amazon S3 Intelligent-Tiering

- Optimized sizes (MBs)
- Many users
- Unpredictable access
- Long-lived assets
- Hot to cool

Online cool data

Amazon S3 Standard Infrequent Access (S3-IA/ZIA)

- Replicated DR data
- Infrequently accessed
- Infrequent queries
- ML model training

Historical data

Amazon S3 Glacier or S3 Glacier Deep Archive

- Historical assets
- ML model training
- Compliance/audit
- Data protection
- Planned restores

Optimize costs for all stages of data lake workflows

Data management at scale: Best practices

Utilize Amazon S3 object tagging

Granularly control access, analyze usage,
manage life cycle policies, and replicate objects

Implement life cycle policies

Automated, policy-driven archive and data expiration

Utilize batch operations

Manage millions to billions of objects with a single request

Plan for rapid growth and automation of management at any scale

Recommendation: Consider performance design patterns

Structure key namespace to scale

Most workloads fit in the S3 3,500 PUT / 5,500 GET TPS per key name partition

Amazon S3 automatically creates partitions as data lake use increases

Extremely bursty workloads might require customized key name design

Consider object format and size

Use Parquet or optimized columnar format

Aim for 16–256 MB minimum object size (might require aggregation during ingest)

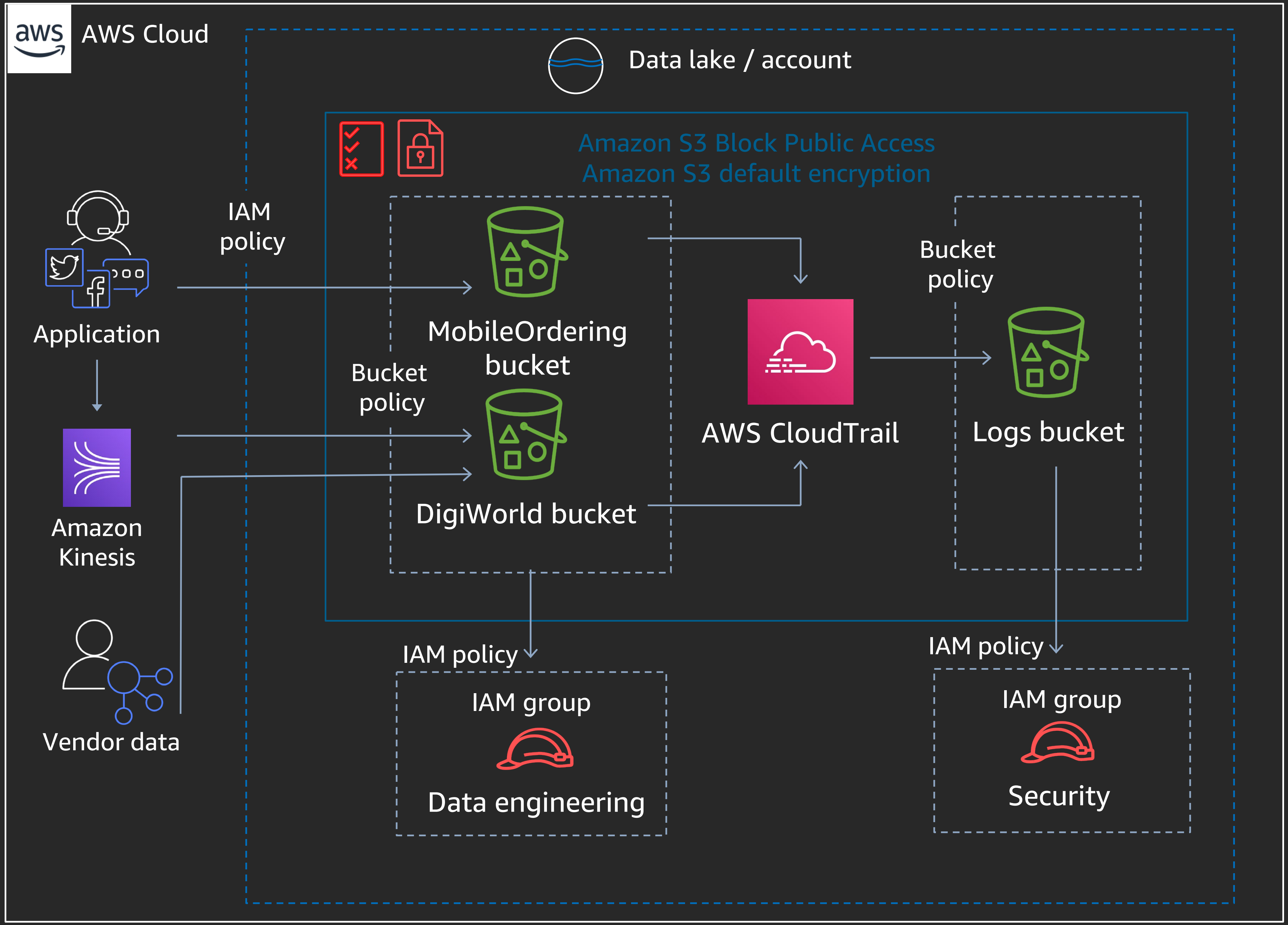
Perform parallel byte range access (included in AWS SDKs)

Use latest SDKs and software versions

AWS SDKs include support for the latest features and optimizations

Amazon EMR 5.18 and above supports S3 Select for Hive, Presto, and Spark

Secure your data lake



Deny access by default

Encrypt your data

Secure multiple data input sources

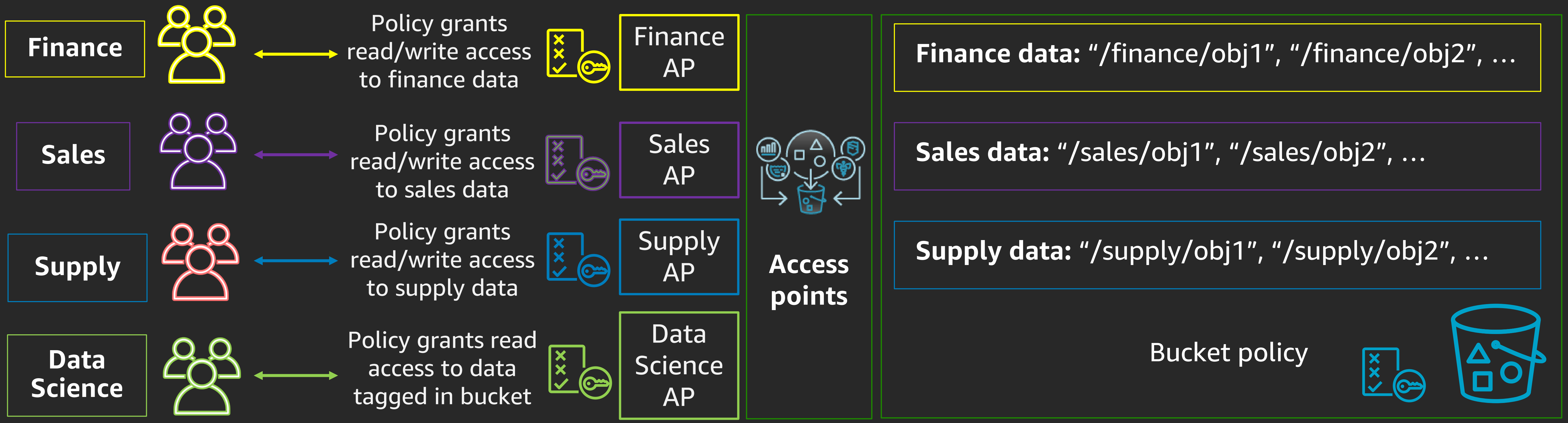
Provide specific access where appropriate

Support multiple unique users and teams

New!

Recommendation: Use Amazon S3 access points to manage your application set

Granular control for hundreds of teams accessing your data lake



Finance

Policy grants read/write access to finance data

Finance AP

Sales

Policy grants read/write access to sales data

Sales AP

Supply

Policy grants read/write access to supply data

Supply AP

Data Science

Policy grants read access to data tagged in bucket

Data Science AP

Access points

Finance data: "/finance/obj1", "/finance/obj2", …

Sales data: "/sales/obj1", "/sales/obj2", …

Supply data: "/supply/obj1", "/supply/obj2", …

Bucket policy

AWS Lake Formation

aws SUMMIT ONLINE

AWS Lake Formation

Build a secure data lake in days

Build data lakes quickly

Move, store, catalog, and clean your data faster

Transform to open formats like Parquet and ORC

ML-based deduplication and record matching

Simplify security management

Centrally define security, governance, and auditing policies

Enforce policies consistently across multiple services

Integrates with IAM and KMS

Provide self-service access to data

Build a data catalog that describes your data

Enable analysts and data scientists to easily find relevant data

Analyze with multiple analytics services without moving data

Demo

aws SUMMIT

Overarching takeaways

- Amazon S3 is the foundation for data lakes
- Leverage pipelined architectures to improve governance, data management, and efficiency
- Improve performance by parallelizing access and scaling horizontally
- Privatize your data lake, encrypt everything, and secure specific access to and from that data lake
- Simplify control for shared bucket access by many teams by using S3 access points

Learn storage with AWS Training and Certification

Resources created by the experts at AWS to help you build cloud storage skills

45+ free digital courses cover topics related to cloud storage, including:

- Amazon S3
- AWS Storage Gateway
- Amazon S3 Glacier
- Amazon Elastic File System (Amazon EFS)
- Amazon Elastic Block Store (Amazon EBS)

Classroom offerings, such as Architecting on AWS, feature AWS expert instructors and hands-on activities

Visit the storage learning path at https://aws.training/storage

Thank you!

Kumar Nachiketa

aws SUMMIT ONLINE