
AWS
re:Invent
NOV. 28 – DEC. 2, 2022 | LAS VEGAS, NV

ARC313-R

Building modern data architectures on AWS

Raghavarao Sodabathina

Principal Solutions Architect
AWS

Agenda

Why modern data architecture?

Modern data strategy

Building modern data architectures

Reference architectures for common scenarios

Best practices & key takeaways

Why modern data architecture?

Create better business outcomes with data

Make better, faster decisions

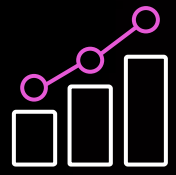
Improve customer experience and loyalty

Stay ahead of the competition

Prepare for the future

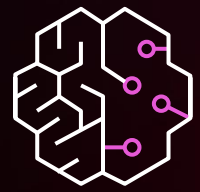
Reduce costs and reimagine processes

However, challenges are in the way



More data than ever is being generated

Data of all types is stored in silos across multiple data stores



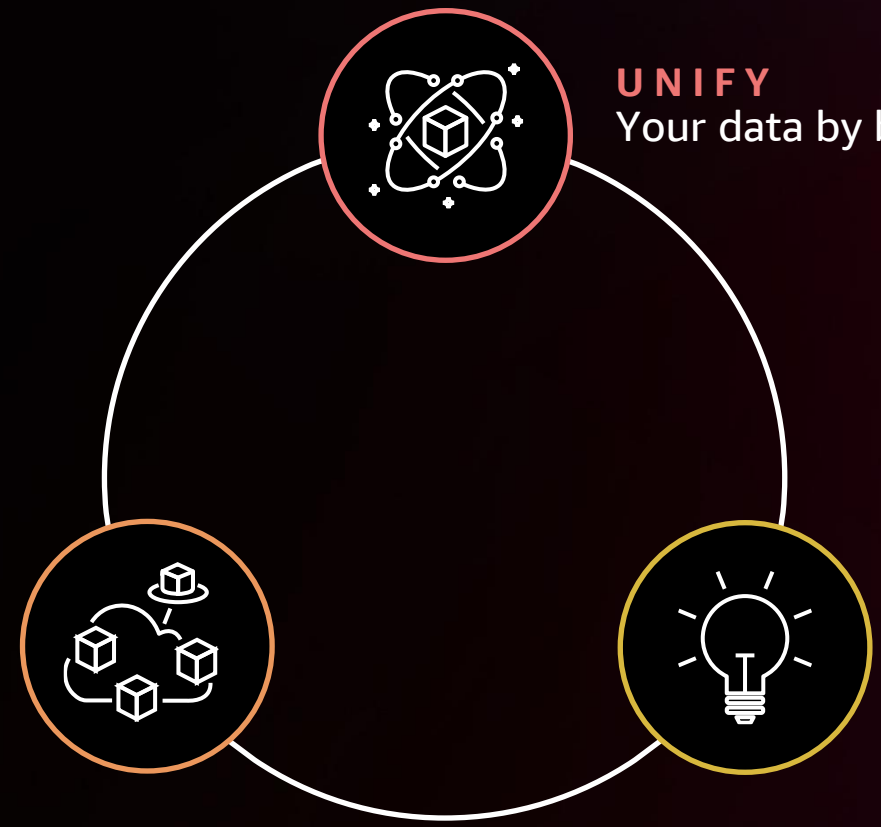
Machine learning adoption is challenged by lack of skills and organizational inertia

Data security, privacy, and compliance regulations are increasingly important

Modern data strategy

Modern data strategy for better business outcomes

START ANYWHERE



UNIFY
Your data by breaking down silos

MODERNIZE
Your data infrastructure to a scalable, trusted, and secure cloud provider

INNOVATE
By inventing new experiences and reimagining existing processes

Modernize

MODERNIZE DATA INFRASTRUCTURE FROM A LEGACY SOLUTION TO A SCALABLE, TRUSTED, AND SECURE CLOUD PROVIDER

- Reduce operational overhead with purpose-built, cloud-based databases
- Modernize analytics tools to handle structured, unstructured, and streaming data – at scale
- Standardize on a modern ML infrastructure to harness the ML benefits at scale

Unify

BREAK DOWN SILOS, SO DATA CAN BE PUT TO WORK ACROSS DATABASES, DATA LAKES, ANALYTICS, AND ML SERVICES

- Unify your data and make data accessible and shared in a secure way
- Ensure that data can easily get to wherever it’s needed, with the right controls
- Enable analysis and insights through analytics, visualization, and ML tools

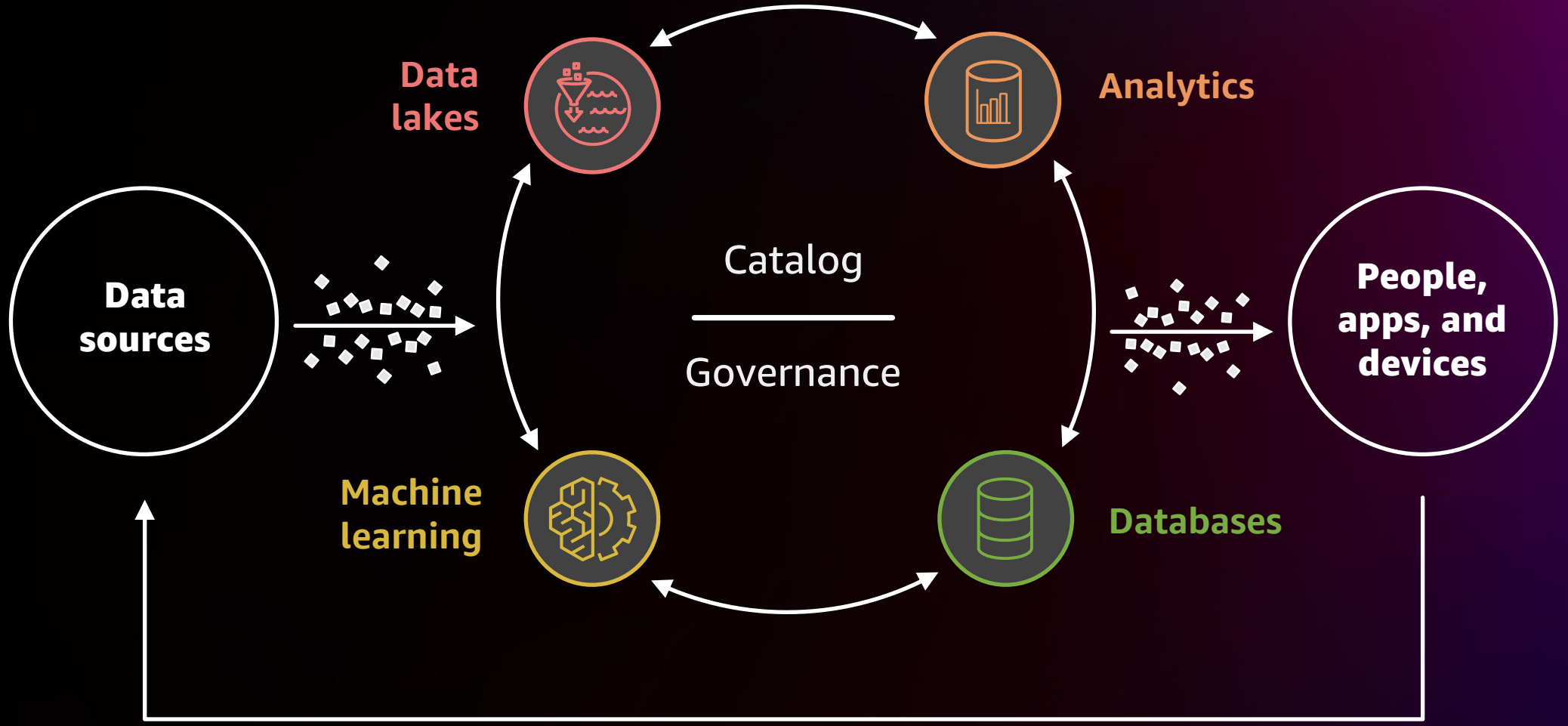
Innovate

INVENT NEW EXPERIENCES AND REIMAGINE PROCESSES WITH PURPOSE-BUILT DATABASES, ADVANCED ANALYTICS AND ML

- As the types of data and workloads evolve, the databases, analytics tools, and ML services need to evolve
- ML is driving unprecedented levels of innovation
- Create better customer experiences with insights and predictions enabled by ML

Building modern data architectures

Modern data architecture



Modern data architecture on AWS

Modern data architecture pillars

Data at any scale

The best price performance

Seamless data access

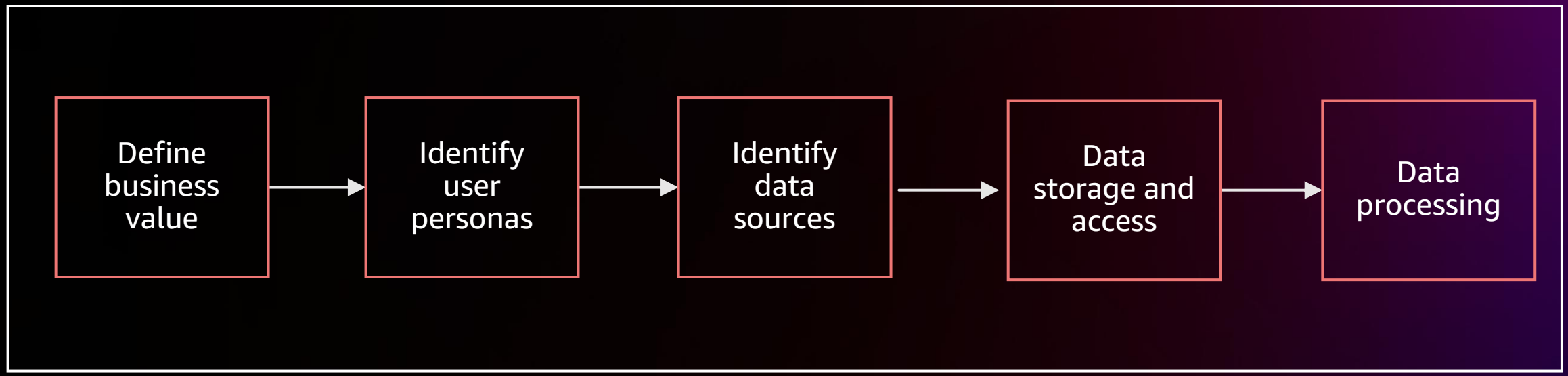
Unified governance

AI and ML to solve business challenges

Data discovery

FIRST STEP IN BUILDING MODERN DATA ARCHITECTURES

The data discovery process consists of a number of interactive sessions with various stakeholders within an organization



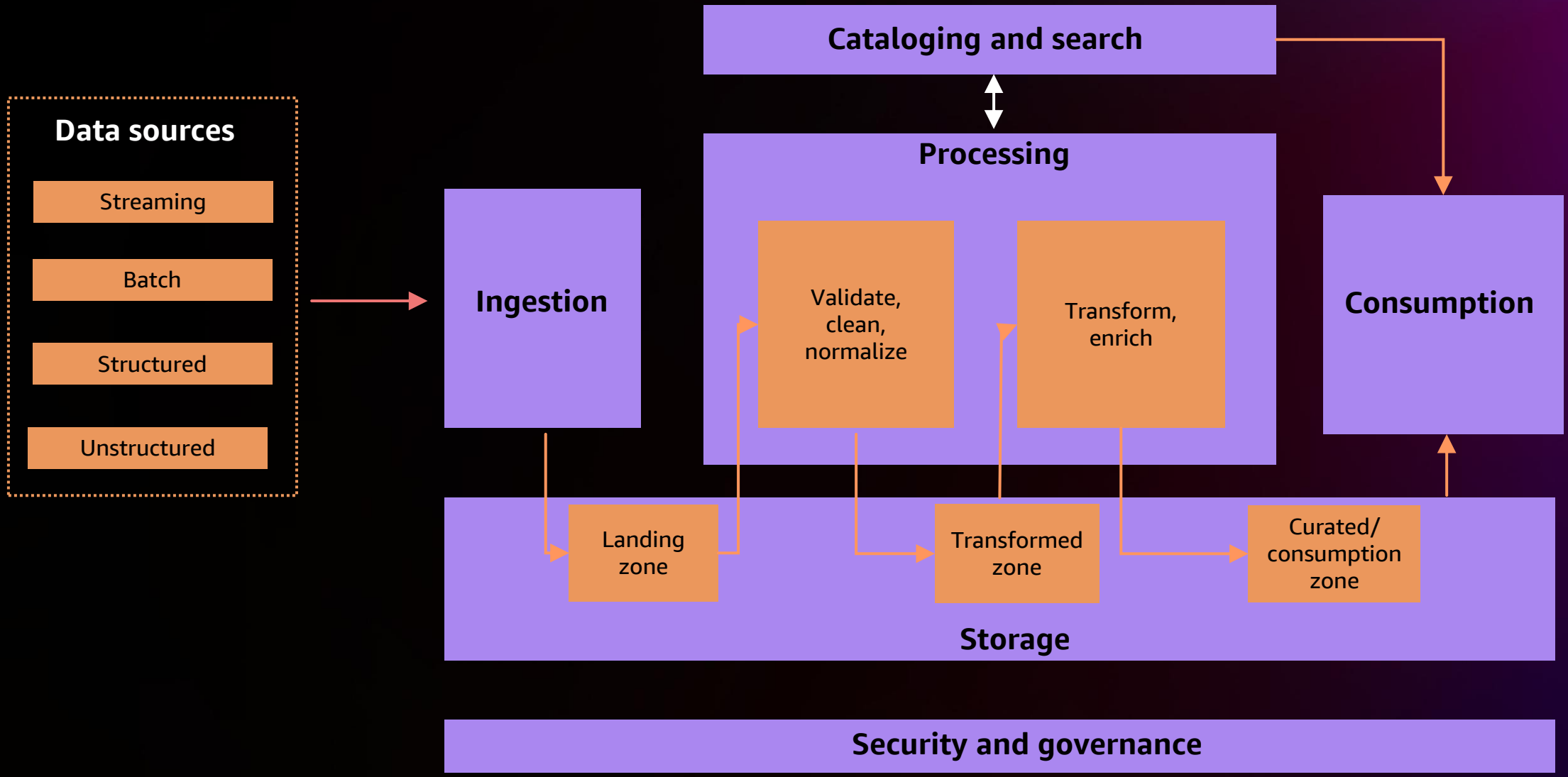
https://docs.aws.amazon.com/wellarchitected/latest/analytics-lens/data-discovery.html

Building modern data architecture

ENVISION A MODERAN DATA ARCHITECTURE AS A STACK OF SIX LAYERS

1. Data ingestion	2. Data storage	3. Data cataloging	4. Data processing	5. Data consumption	6. Security and governance
Bring the data into your data platform	Store your structured and unstructured data	Store your metadata	Create data processing pipelines	Enable your user personas for purpose-built analytics and machine learning	Protect your data across the layers and data access management

Layered modern data architecture



Data sources

Modern data architecture enables you to bring a wide variety of data from various data sources

TYPICAL DATA SOURCES IN AN ORGANIZATION

Data type	Data sources
Structured data	ERP applications, CRM applications, CMS applications , SAAS applications, SAP applications and line-of-business (LOB) applications, SQL databases
Semi-structured data	Web applications, NoSQL databases, EDI (Electronic data interchange), CSV, XML, JSON documents, etc.
Unstructured data	Video files, audio files, images, IoT data, sensors data, invoices, etc.
Batch	Most of of internal applications generate structured data at regular defined schedules

Building modern data architecture

ENVISION A MODERAN DATA ARCHITECTURE AS A STACK OF SIX LAYERS

1. Data ingestion	2. Data storage	3. Data cataloging	4. Data processing	5. Data consumption	6. Security and governance
Bring the data into your data platform	Store your structured and unstructured data	Store your metadata	Create data processing pipelines	Enable your user personas for purpose-built analytics and machine learning	Protect your data across the layers and data access management

Data ingestion layer

Ingest data from a wide variety of data sources to support unique data sources and data types

The typical list of data sources

Database data sources

Files shares

SaaS applications

Partner data feeds

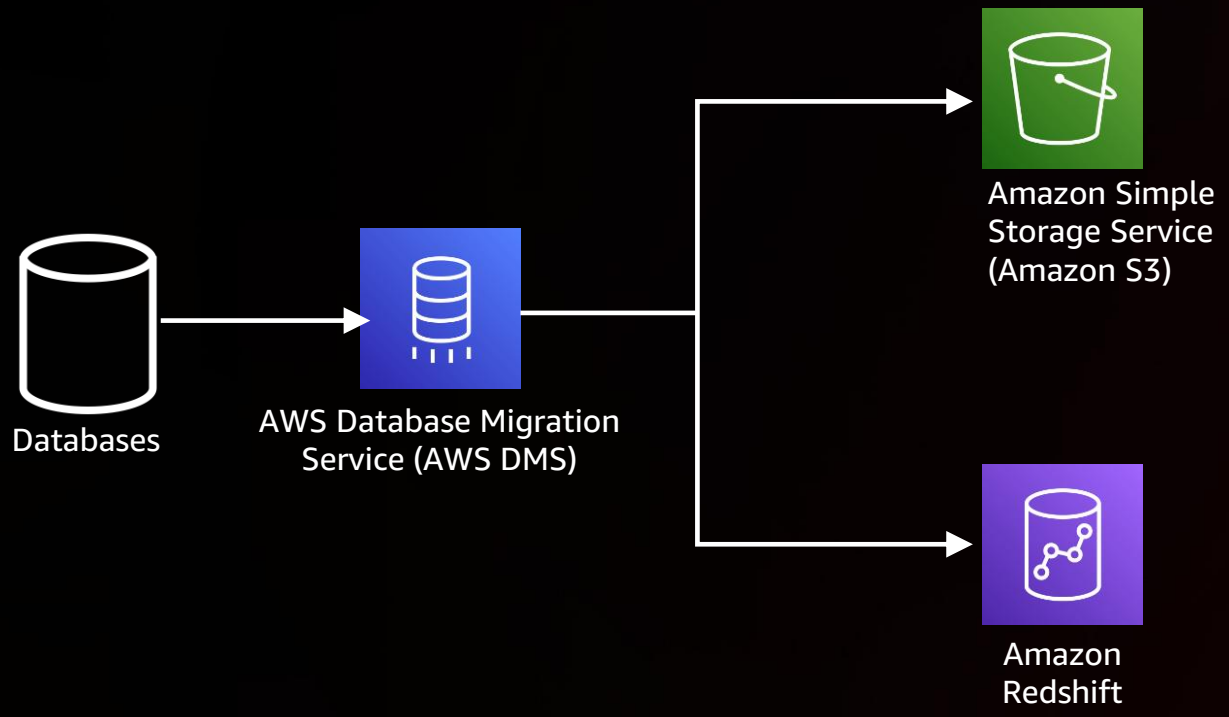
Third-party data products

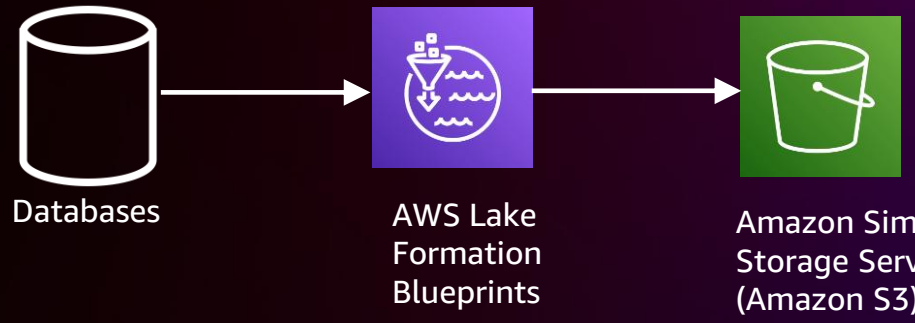
Custom data sources

Streaming data sources

Database data sources

We provide AWS Database Migration Service (AWS DMS) and AWS Lake Formation blueprints by generating AWS Glue crawlers, jobs, and triggers that discover and ingest database data into storage layer



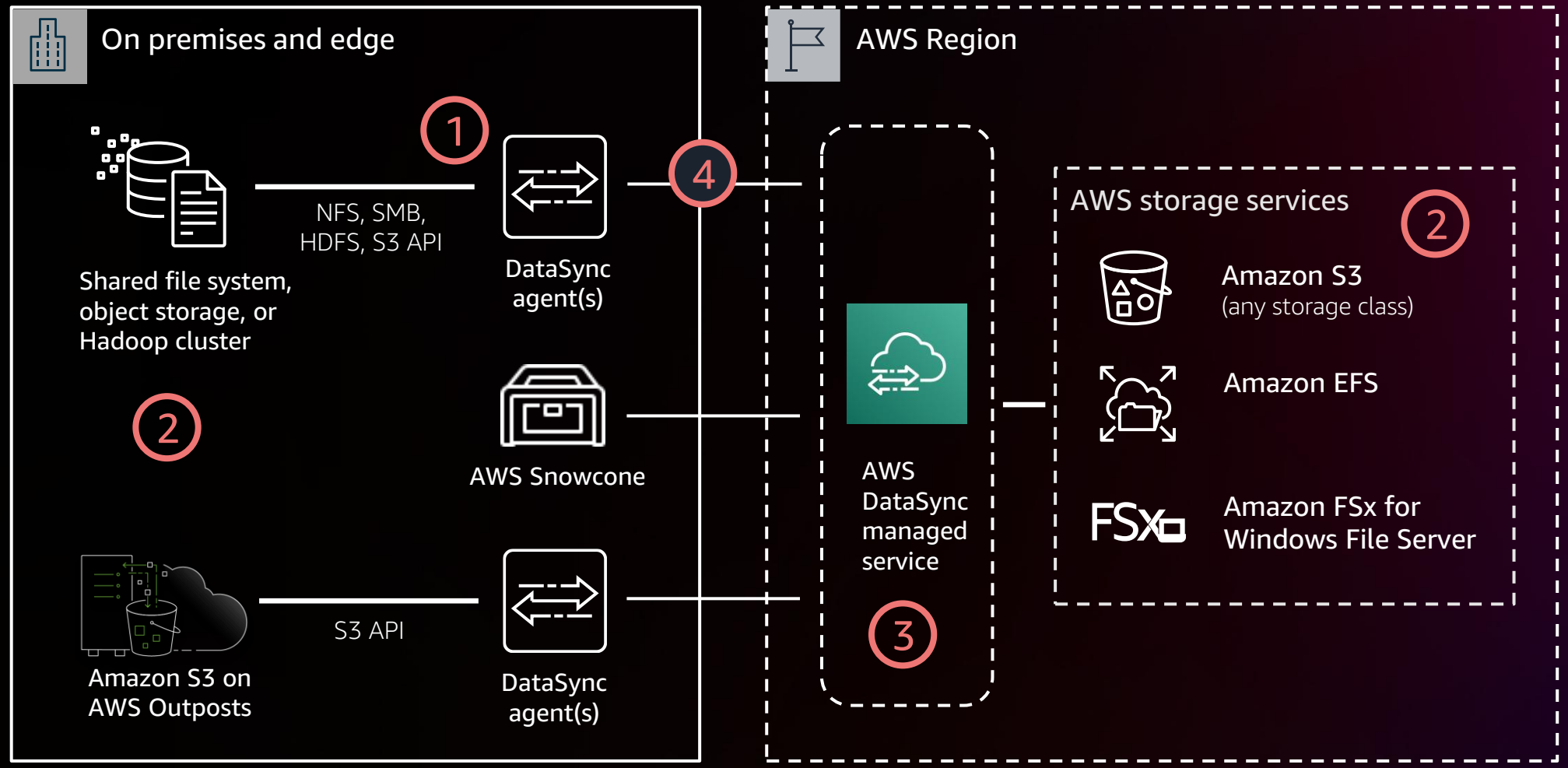


https://docs.aws.amazon.com/dms/latest/userguide/CHAP_Source.html
https://docs.aws.amazon.com/dms/latest/userguide/CHAP_Target.html

https://docs.aws.amazon.com/lake-formation/latest/dg/workflows-about.html

File shares

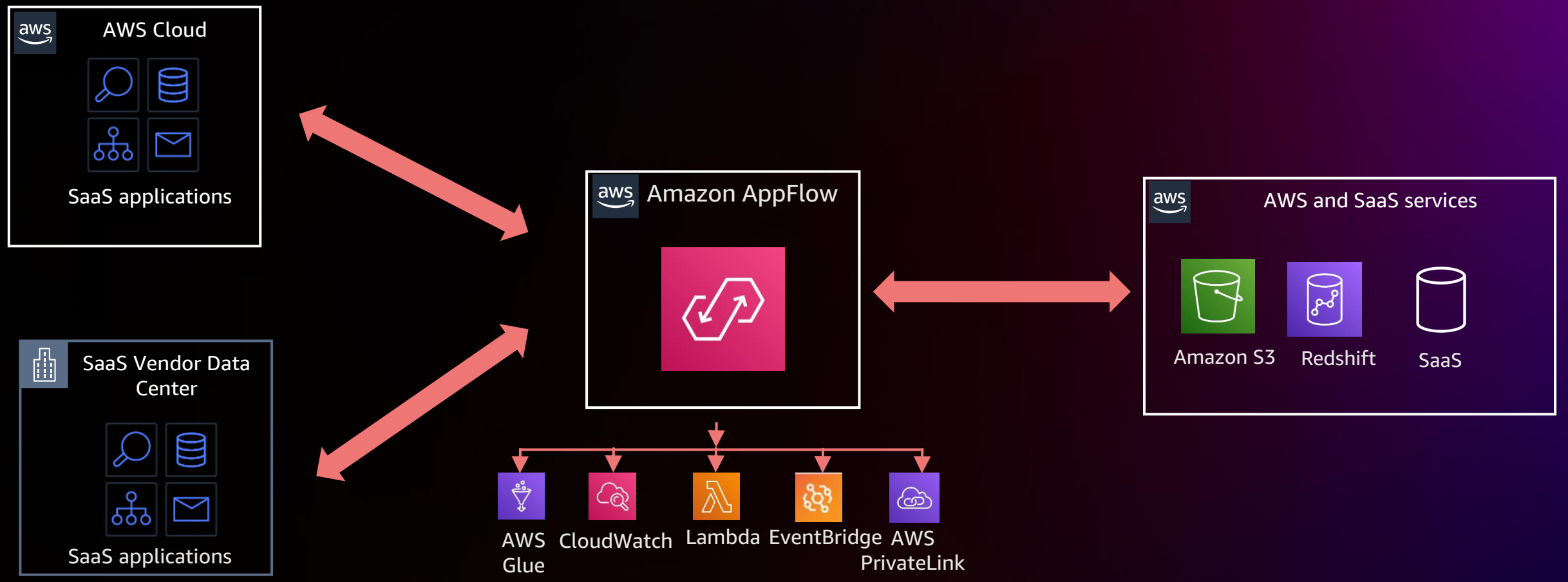
AWS DataSync makes it simple and fast to move large amounts of files from Network File System (NFS) shares, Server Message Block (SMB) shares, Hadoop Distributed File Systems (HDFS) into Amazon S3 data lake



1. Agents are deployed to connect to on-premises storage
2. Locations control how AWS DataSync connects to storage
3. AWS DataSync managed service connects to AWS storage and coordinates the transfer
4. For transfers between on premises and AWS, internet, AWS Direct Connect, and AWS Virtual Private Network (VPN) are supported

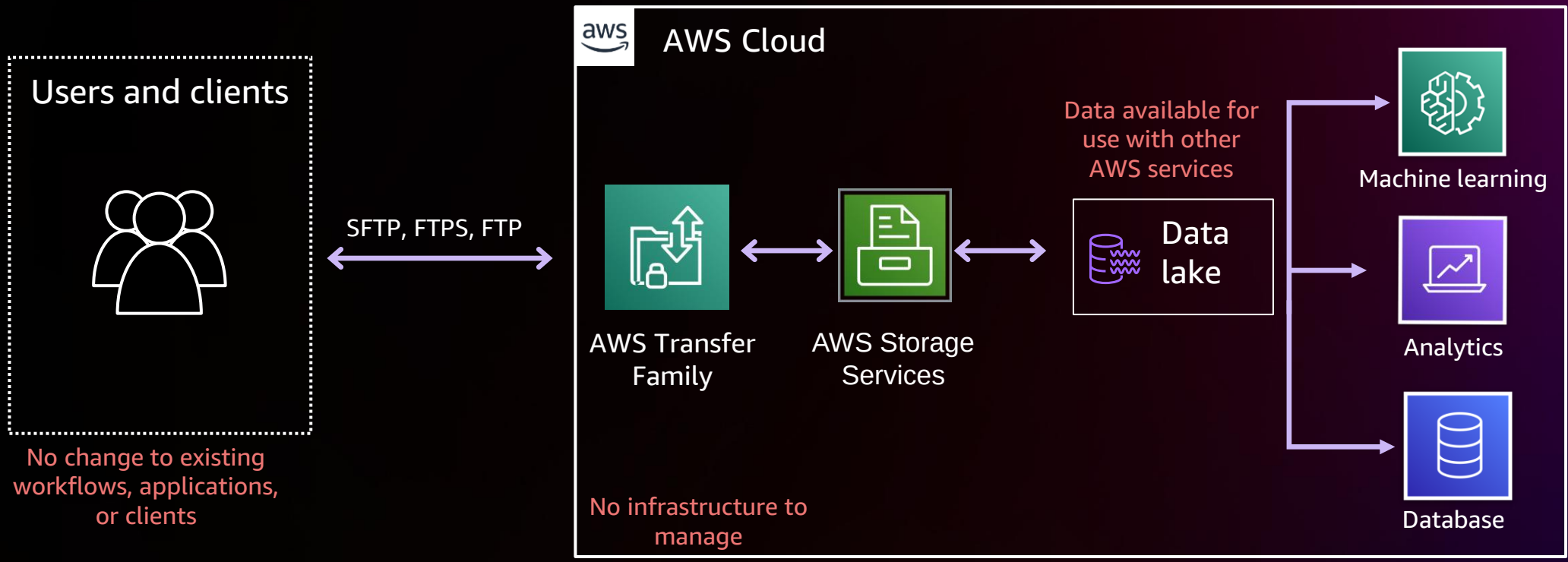
SaaS applications data

Amazon AppFlow makes it easy to ingest SaaS applications data into storage layer



Partner data feeds

AWS Transfer Family is a serverless service that provides secure FTP endpoints and integrates with Amazon S3 and it stores partner data feeds as S3 objects in the landing zone of the data lake



Third-party data sources

AWS Data Exchange has hundreds of commercial data products across various industries such as financial services, healthcare, retail, media & entertainment

CHANGE HEALTHCARE
EQUIFAX
arabesque
FACTSET
FOURSQUARE
REUTERS
experian
dun & bradstreet
rearc
IMDb
CSRHUB
Tradition

Ingest files

AWS Data Exchange for Amazon S3

Automatic exports of new or updated data as files in Amazon S3

Query tables

AWS Data Exchange for Amazon Redshift

Access to live, production-ready tables in Amazon Redshift, without ETL

Call APIs

AWS Data Exchange for APIs

Integrate third-party data into any software via API requests, without ETL

Custom data sources ingestion using AWS Glue connectors

Built-in connectors

Out of box connectors to support high performance ingestion

Custom connectors

Flexible method to build your own connectors

Marketplace connectors

Subscription-based low-cost connectors

Elasticsearch

JIRA

Amazon S3

Amazon Redshift

Microsoft Dynamics 365

SharePoint

Amazon Athena

Amazon DynamoDB

Microsoft SQL Server

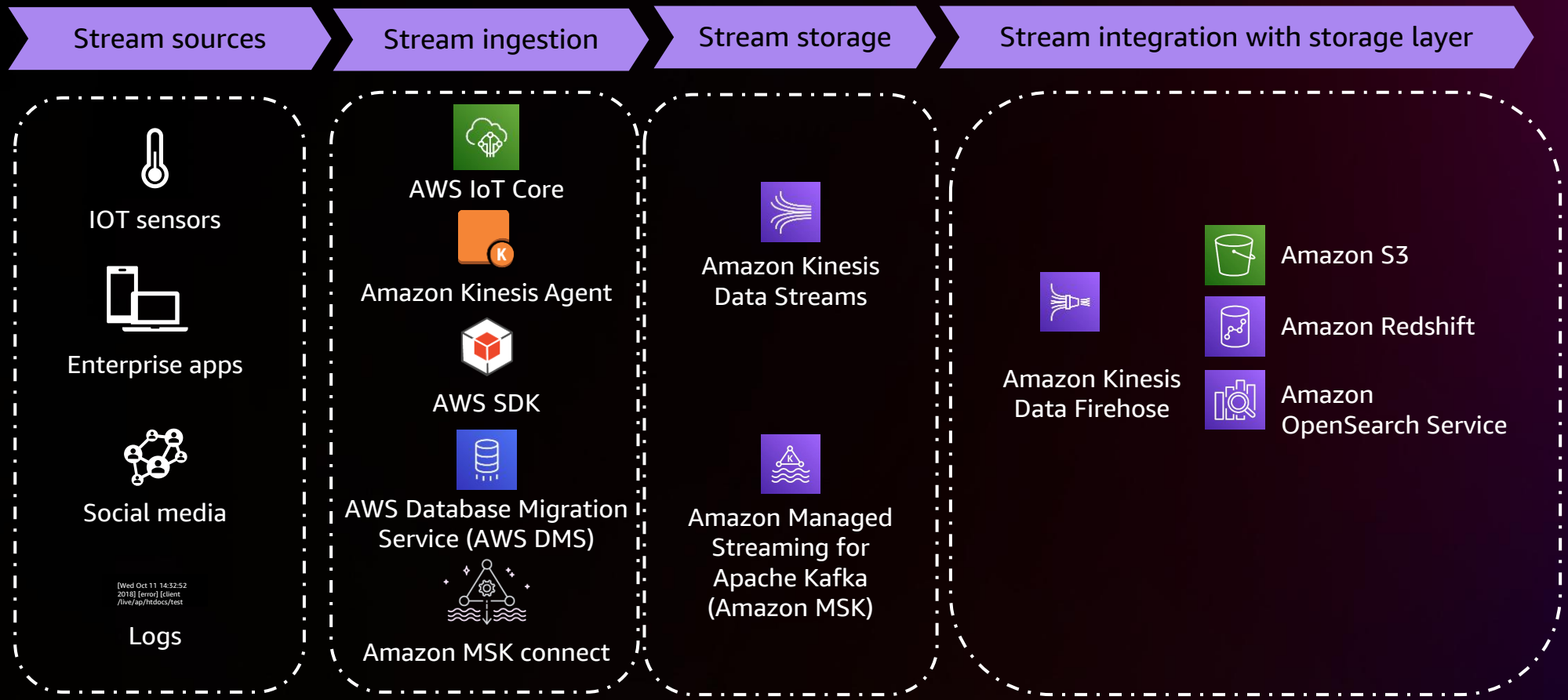
Amazon Athena

Google Ads

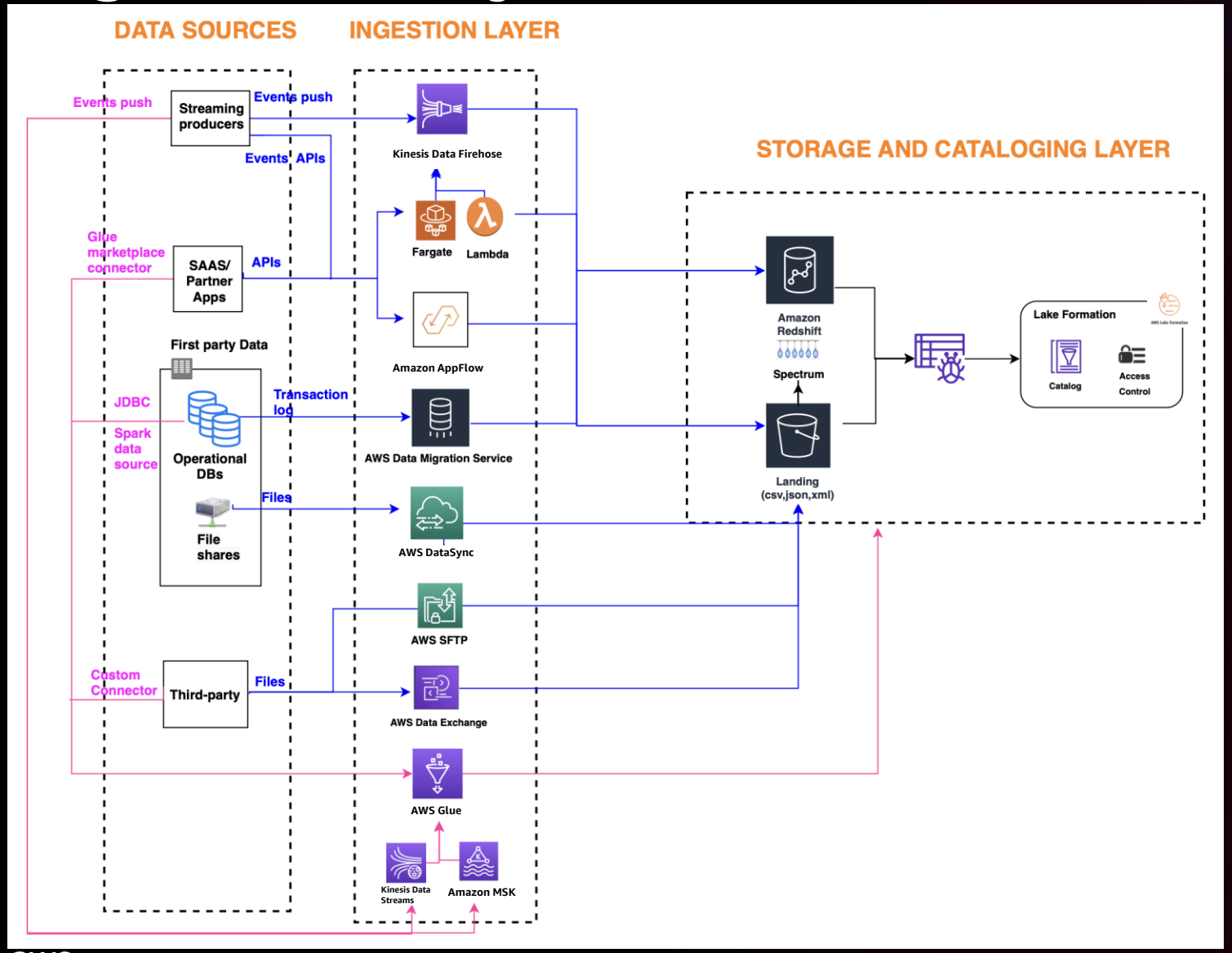
100+ connectors

Streaming data sources

AWS provides Kinesis Data Streams, Kinesis Data Firehose and Managed Streaming for Apache Kafka (Amazon MSK) services to ingest streaming data into the storage layer



Ingestion layer architecture



Design considerations

- Support diverse data types (structured, semi-structured, unstructured)
- Support diverse source connection methods (JDBC, APIs, FTP, Streaming event capture and buffering, Log delivery)
- Support variety of ingestion latencies
- Support for Snapshop and Deltas
- Support diverse targets (Data lake, Data Warehouse, Operational DB, Streaming buffer), data formats, compression, partitioning
- Support Snapshot and delta ingestion
- Cost Efficient, Secure, No/Low Code

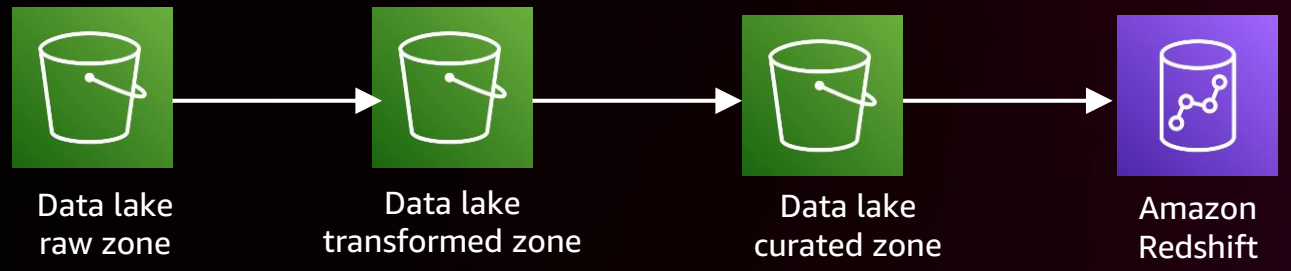
Building modern data architecture

ENVISION A MODERAN DATA ARCHITECTURE AS A STACK OF SIX LAYERS

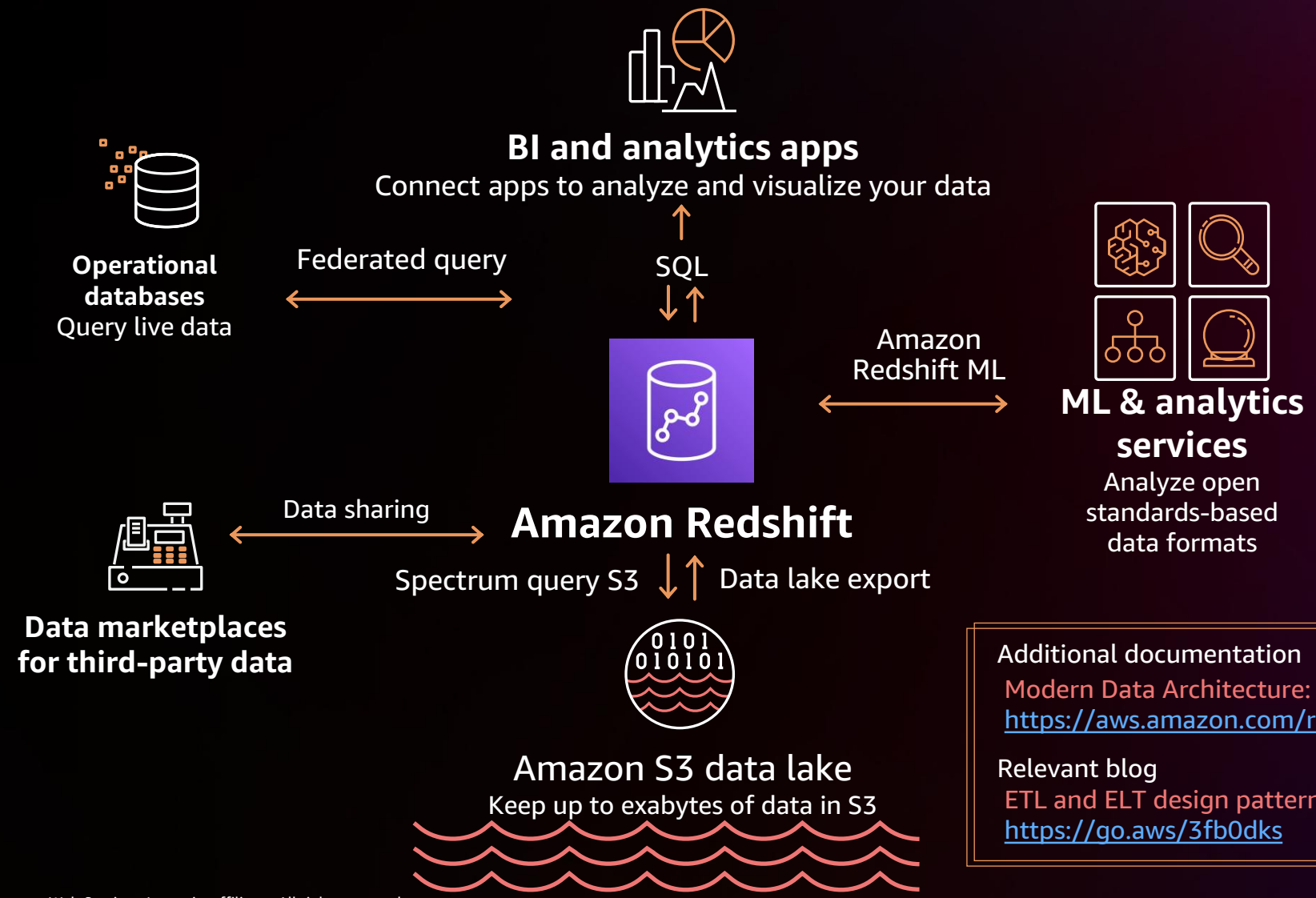
1. Data ingestion	2. Data storage	3. Data cataloging	4. Data processing	5. Data consumption	6. Security and governance
Bring the data into your data platform	Store your structured and unstructured data	Store your metadata	Create data processing pipelines	Enable your user personas for purpose-built analytics and machine learning	Protect your data across the layers and data access management

Data storage layer

The storage layer consists of Amazon S3 and Amazon Redshift, an integrated storage layer for the modern data architectures on AWS. You can put datasets into three different areas in S3 data lake: raw zone, cleaned or transformed zone, and curated zone



Modern data architecture storage layer integrates Amazon S3 data lake and Amazon Redshift data warehouse



Additional documentation

Modern Data Architecture:
https://aws.amazon.com/redshift/lake-house-architecture/

Relevant blog

ETL and ELT design patterns for Modern Data Architecture:
https://go.aws/3fb0dks

Building modern data architecture

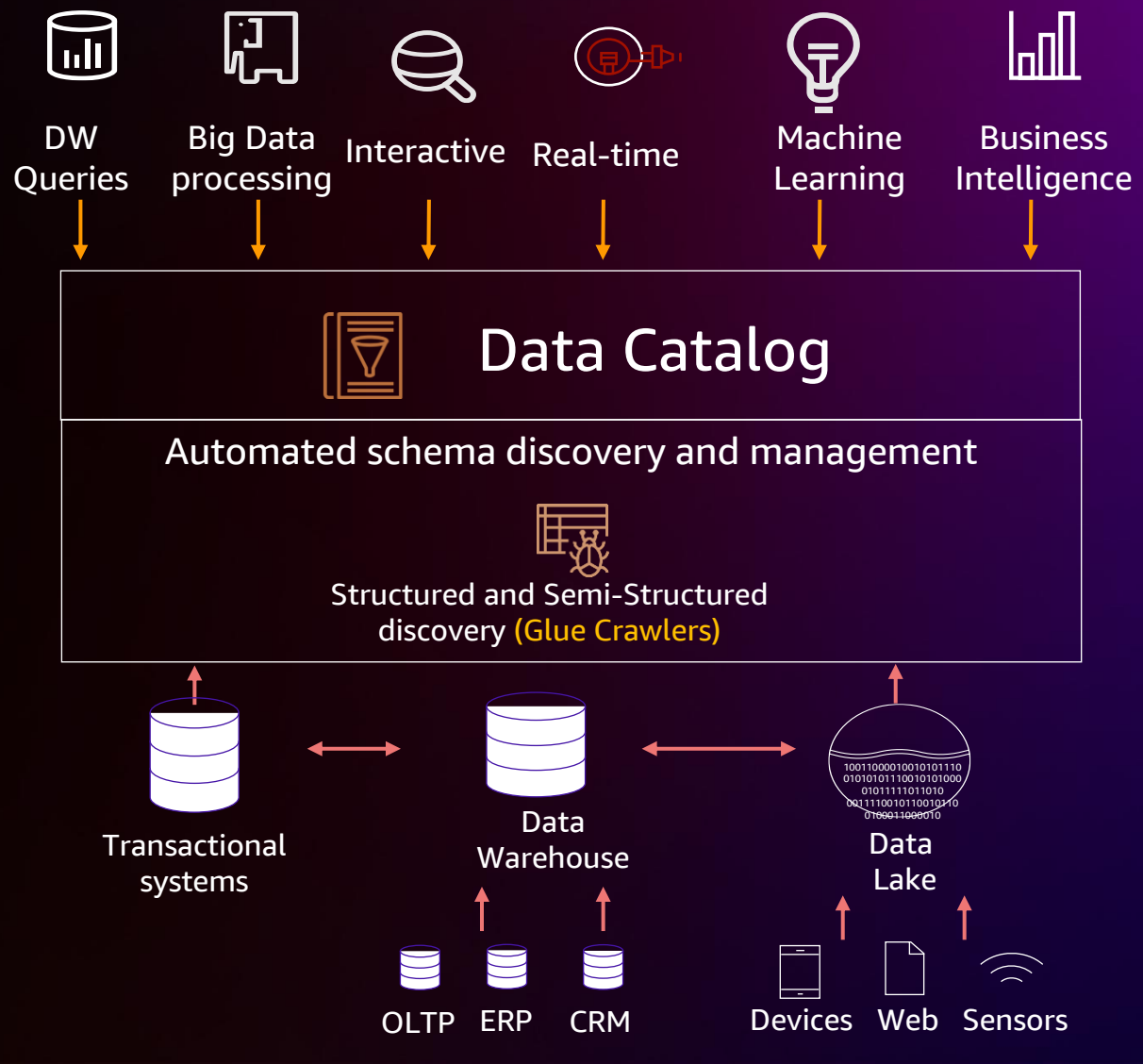
ENVISION A MODERAN DATA ARCHITECTURE AS A STACK OF SIX LAYERS

1. Data ingestion	2. Data storage	3. Data cataloging	4. Data processing	5. Data consumption	6. Security and governance
Bring the data into your data platform	Store your structured and unstructured data	Store your metadata	Create data processing pipelines	Enable your user personas for purpose-built analytics and machine learning	Protect your data across the layers and data access management

Data catalog layer

AWS Glue Data Catalog provides the central catalog to store metadata for all datasets hosted in the storage layer

- No movement of data = Low Costs/Admin
- All metadata centrally available for search and query = Productivity
- Unify structured, semi-structured data = Speed to Insight
- Automate data discovery = Productivity



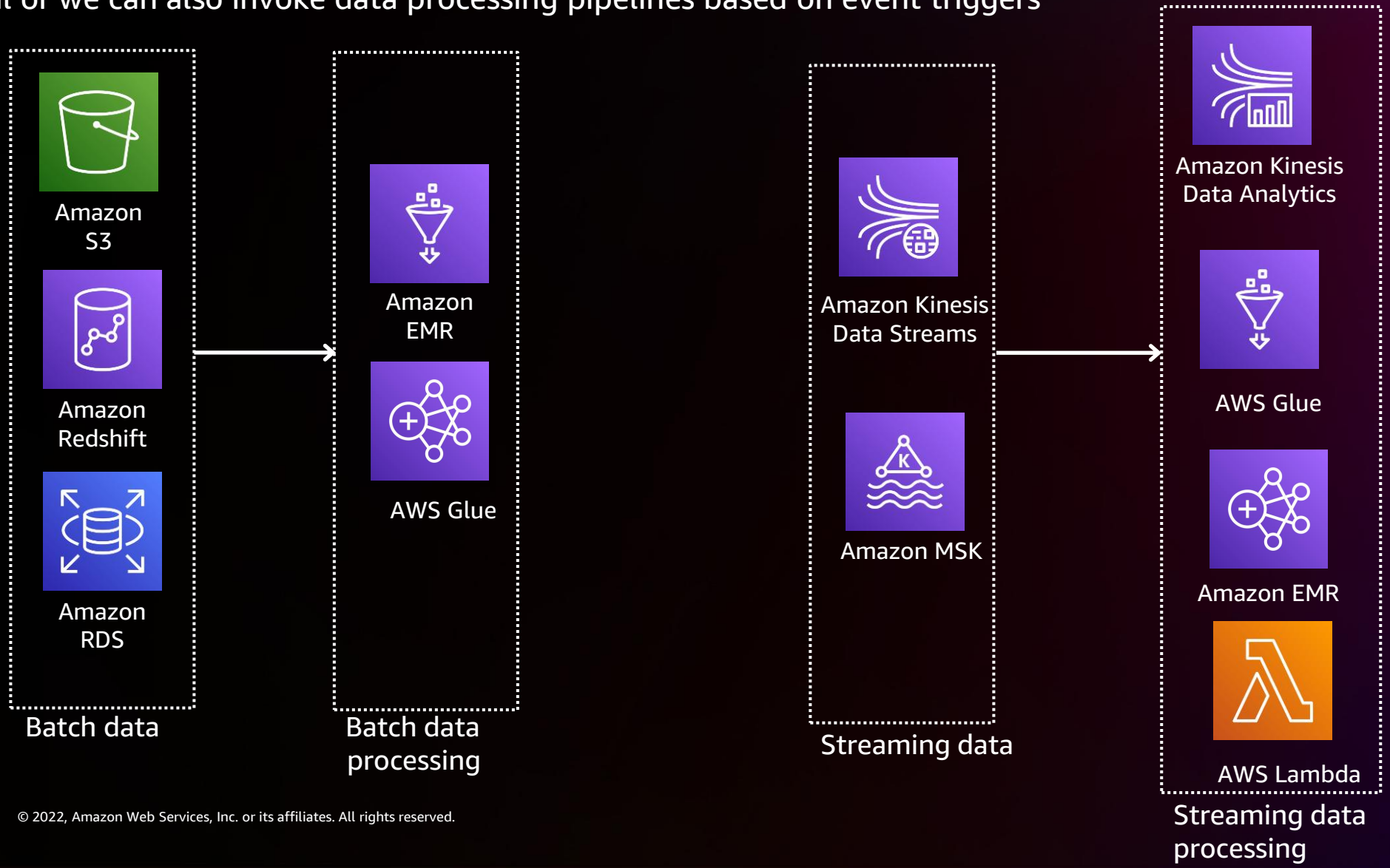
Building modern data architecture

ENVISION A MODERAN DATA ARCHITECTURE AS A STACK OF SIX LAYERS

1. Data ingestion	2. Data storage	3. Data cataloging	4. Data processing	5. Data consumption	6. Security and governance
Bring the data into your data platform	Store your structured and unstructured data	Store your metadata	Create data processing pipelines	Enable your user personas for purpose-built analytics and machine learning	Protect your data across the layers and data access management

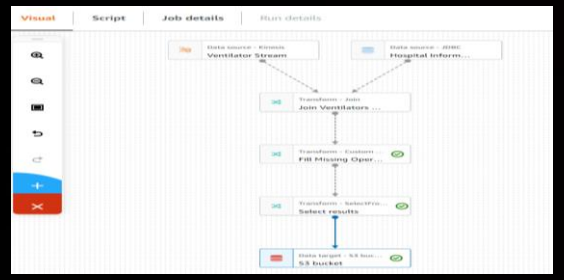
Data processing layer

Data processing pipelines can be multistep data processing pipelines or scheduled data processing pipelines on a regular interval or we can also invoke data processing pipelines based on event triggers

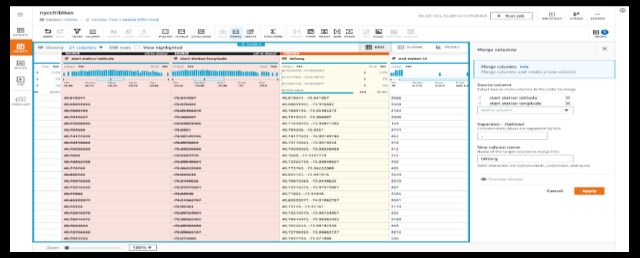


Data processing with AWS Glue

Clients



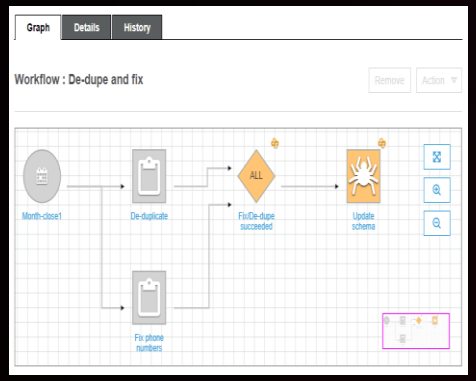
AWS Glue Studio
Visual ETL



AWS Glue DataBrew
Visual data preparation

Interactive Notebooks

Orchestration



Workflows

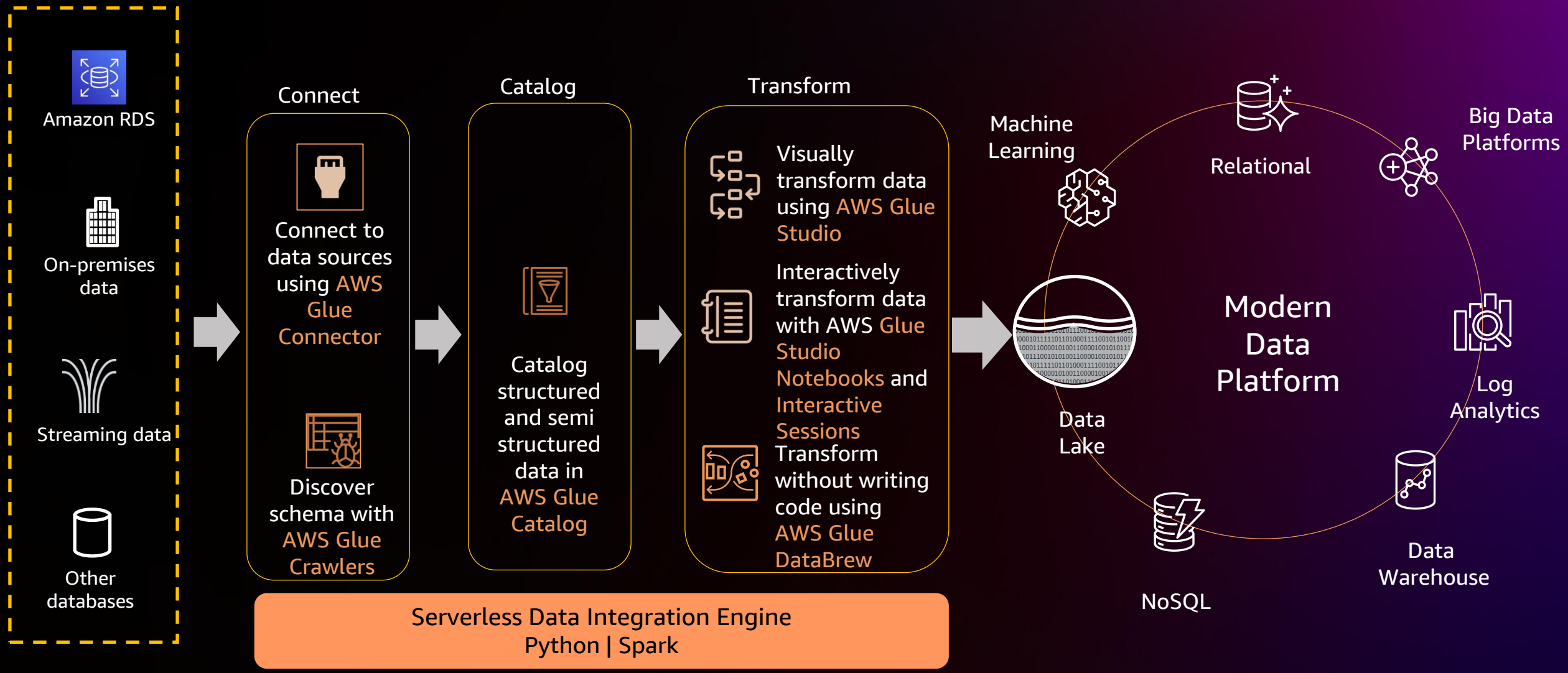
AWS Step Functions

Run time

AWS Glue

- Serverless data processing
- Dynamic scaling
- Simple per-second billing
- Apache Spark or Pure Python
- Connections to dozens of sources and targets

Seamless data integration with AWS Glue



Data processing with Amazon EMR

Clients

EMR Studio

SageMaker Studio

- Manage processing logic + EMR clusters

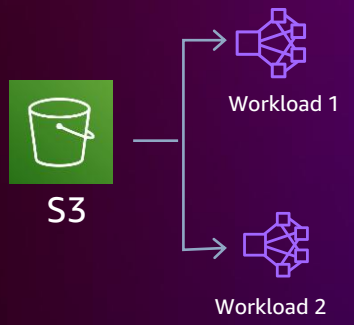
Orchestration

Apache Airflow

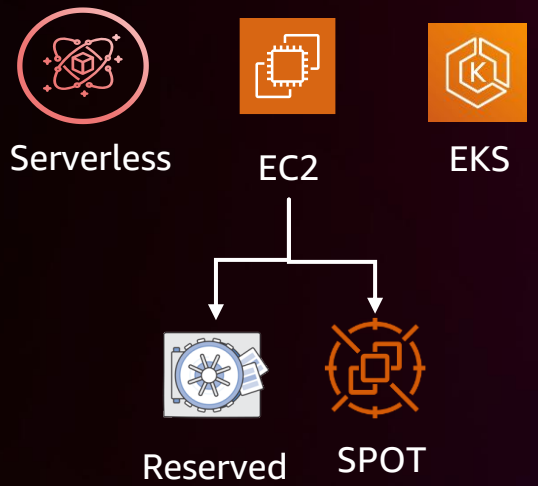
Amazon Managed Workflows for Apache Airflow (Amazon MWAA)

AWS Step Functions

Run time



Compute-storage decoupling



Compute options

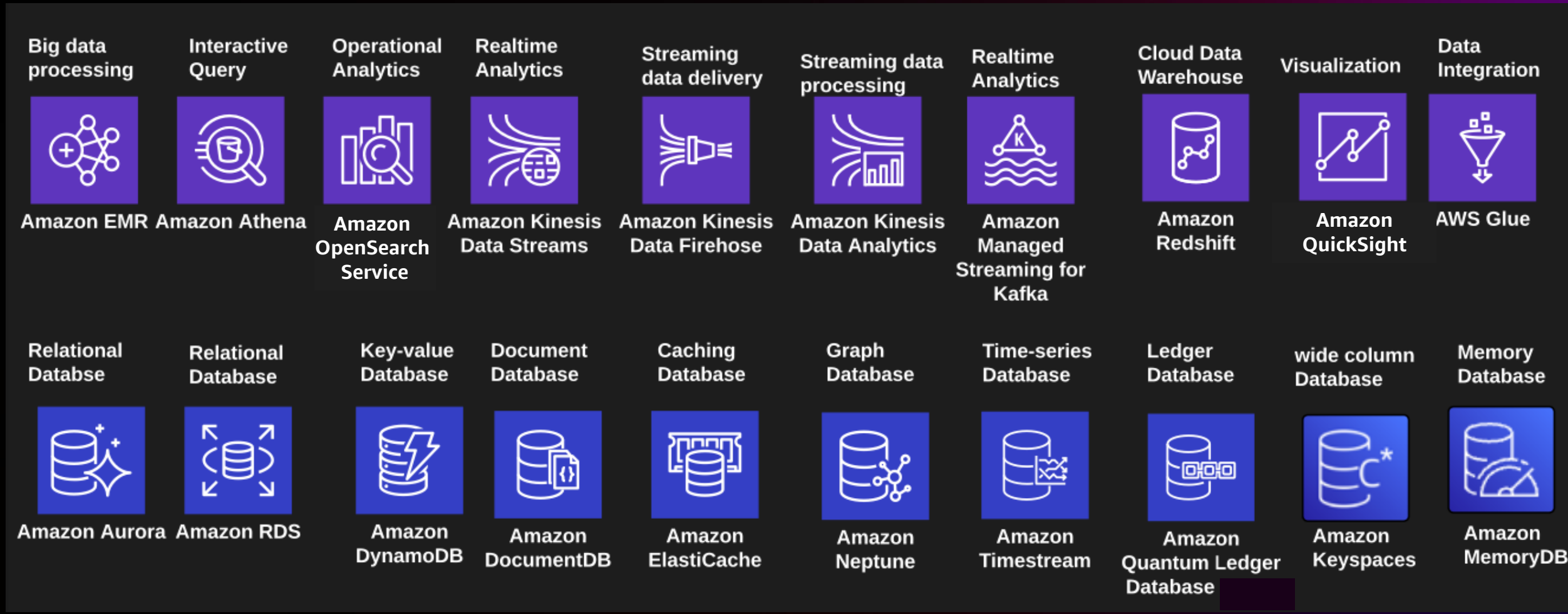
Building modern data architecture

ENVISION A MODERAN DATA ARCHITECTURE AS A STACK OF SIX LAYERS

1. Data ingestion	2. Data storage	3. Data cataloging	4. Data processing	5. Data consumption	6. Security and governance
Bring the data into your data platform	Store your structured and unstructured data	Store your metadata	Create data processing pipelines	Enable your user personas for purpose-built analytics and machine learning	Protect your data across the layers and data access management

Data consumption layer – data services

The modern data architecture on AWS powers deeper and faster purpose-build data services for a wide variety of use cases



Data consumption layer – Machine learning

Amazon SageMaker is a complete, end-to-end service for machine learning

PREPARE

SageMaker Ground Truth
Label training data for machine learning

SageMaker Data Wrangler
Aggregate and prepare data for machine learning

SageMaker Processing
Built-in Python, BYO R/Spark

SageMaker Feature Store
Store, update, retrieve, and share features

SageMaker Clarify
Detect bias and understand model predictions

BUILD

SageMaker Studio Notebooks
Jupyter notebooks with elastic compute and sharing

Built-in and Bring your-own Algorithms
Dozens of optimized algorithms or bring your own

Local Mode
Test and prototype on your local machine

SageMaker Autopilot
Automatically create machine learning models with full visibility

SageMaker JumpStart NEW
Pre-built solutions for common use cases

TRAIN & TUNE

Managed Training
Distributed infrastructure management

SageMaker Experiments
Capture, organize, and compare every step

Automatic Model Tuning
Hyperparameter optimization

Distributed Training Libraries NEW
Training for large datasets and models

SageMaker Debugger NEW
Debug and profile training runs

Managed Spot Training
Reduce training cost by 90%

DEPLOY & MANAGE

Managed Deployment
Fully managed, ultra low latency, high throughput

Kubernetes & Kubeflow Integration
Simplify Kubernetes-based machine learning

Multi-Model Endpoints
Reduce cost by hosting multiple models per instance

SageMaker Model Monitor
Maintain accuracy of deployed models

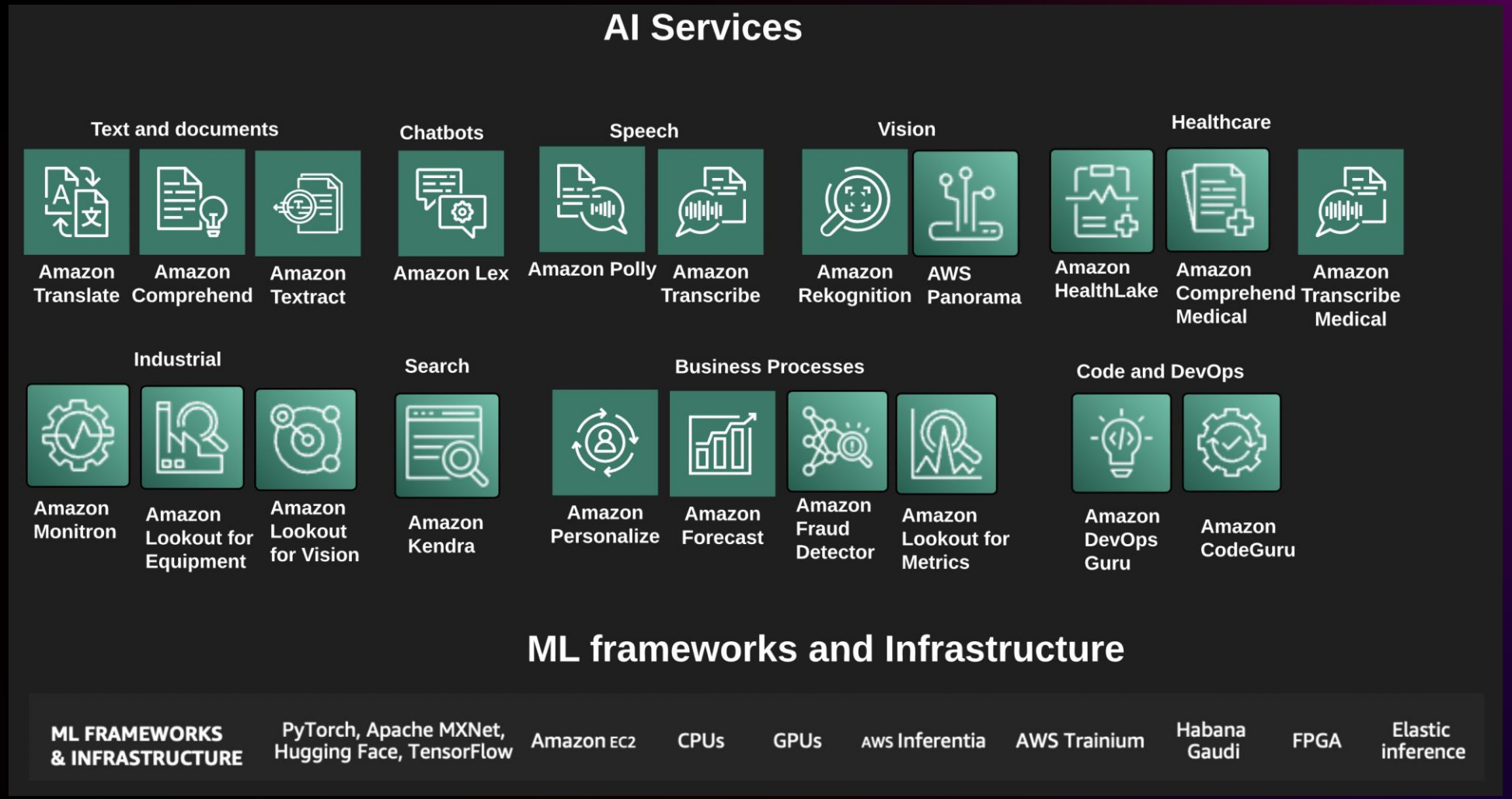
SageMaker Edge Manager NEW
Manage and monitor models on edge devices

SageMaker Pipelines NEW
Workflow orchestration and automation

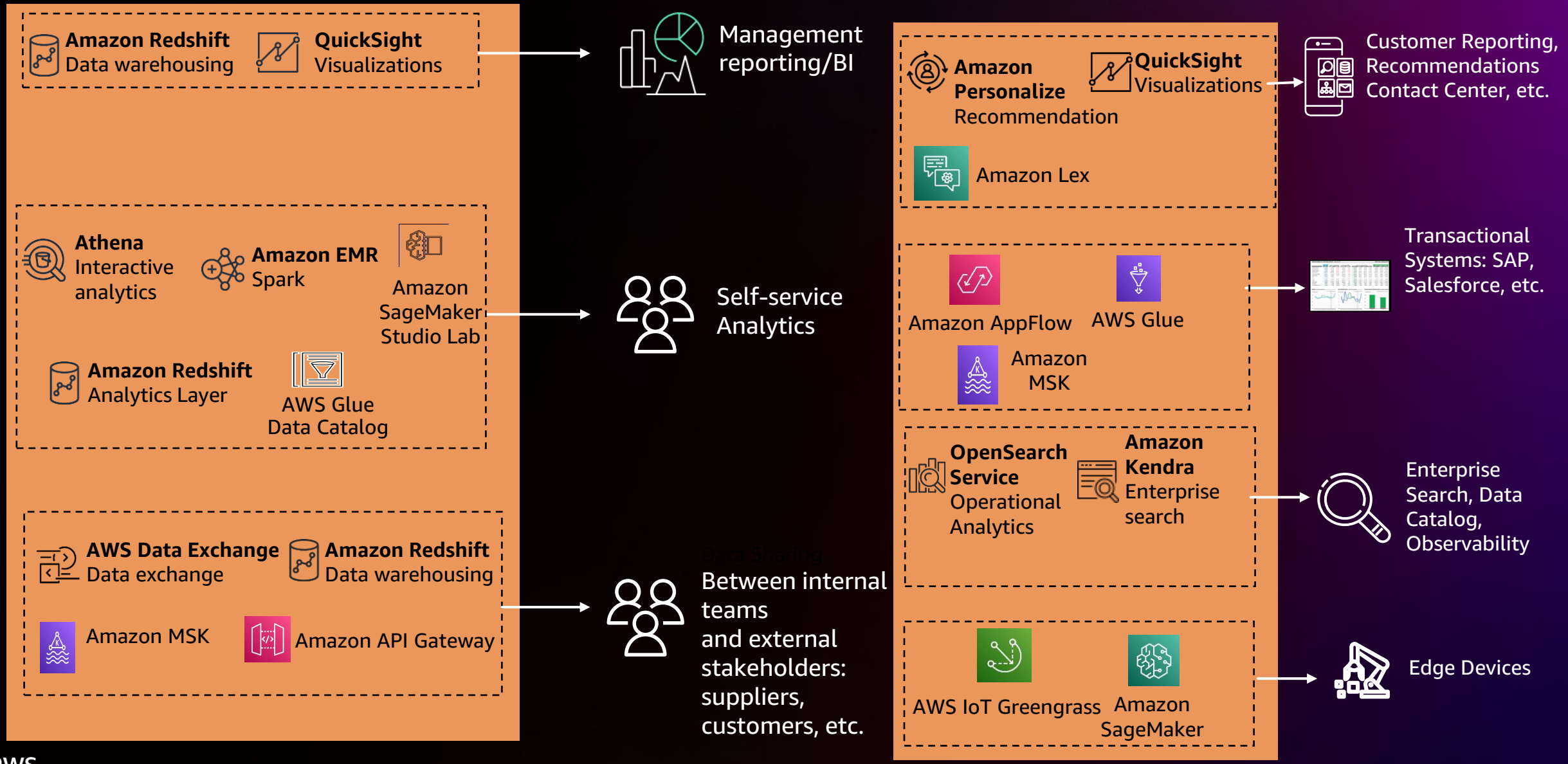
SageMaker Studio
Integrated development environment ((IDE) for ML

Artificial intelligence (AI) & ML Frameworks

You can build intelligent apps by using purpose-build AI services with no machine learning experience
Expert ML practitioners can choose framework of their choice by using AWS ML frameworks and infrastructure



Data consumption patterns



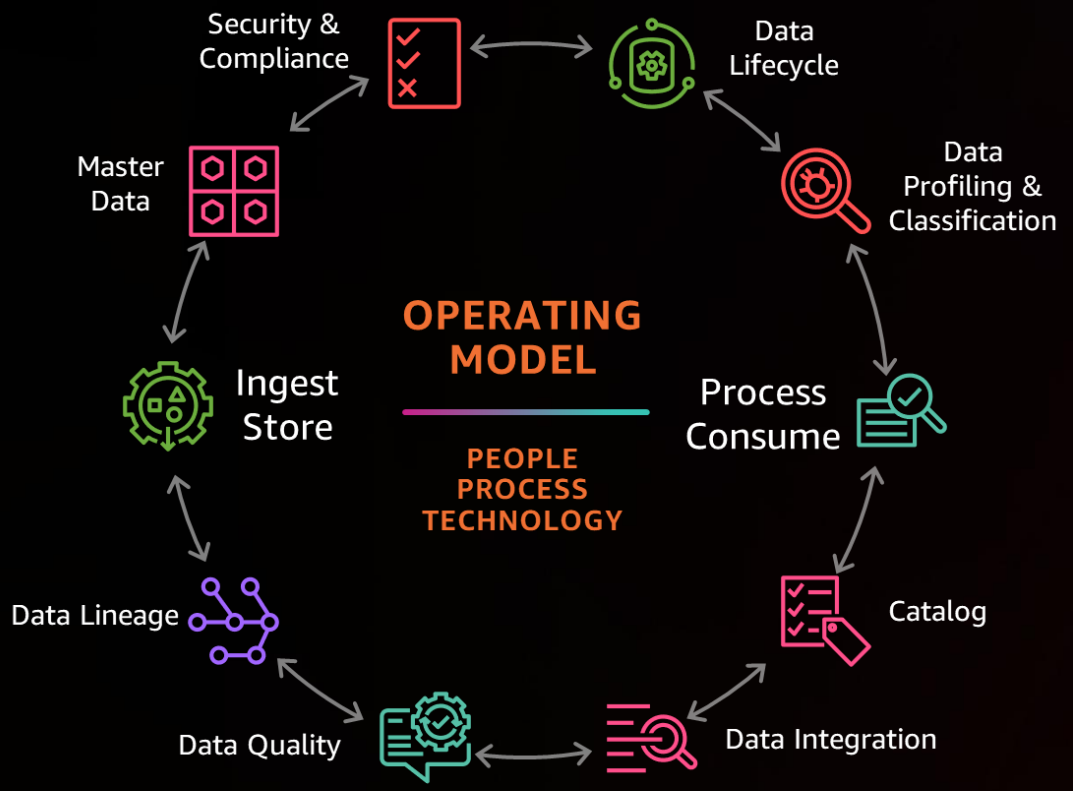
Building modern data architecture

ENVISION A MODERAN DATA ARCHITECTURE AS A STACK OF SIX LAYERS

1. Data ingestion	2. Data storage	3. Data cataloging	4. Data processing	5. Data consumption	**6. Security and governance**
Bring the data into your data platform	Store your structured and unstructured data	Store your metadata	Create data processing pipelines	Enable your user personas for purpose-built analytics and machine learning	Protect your data across the layers and data access management

Security and governance layer

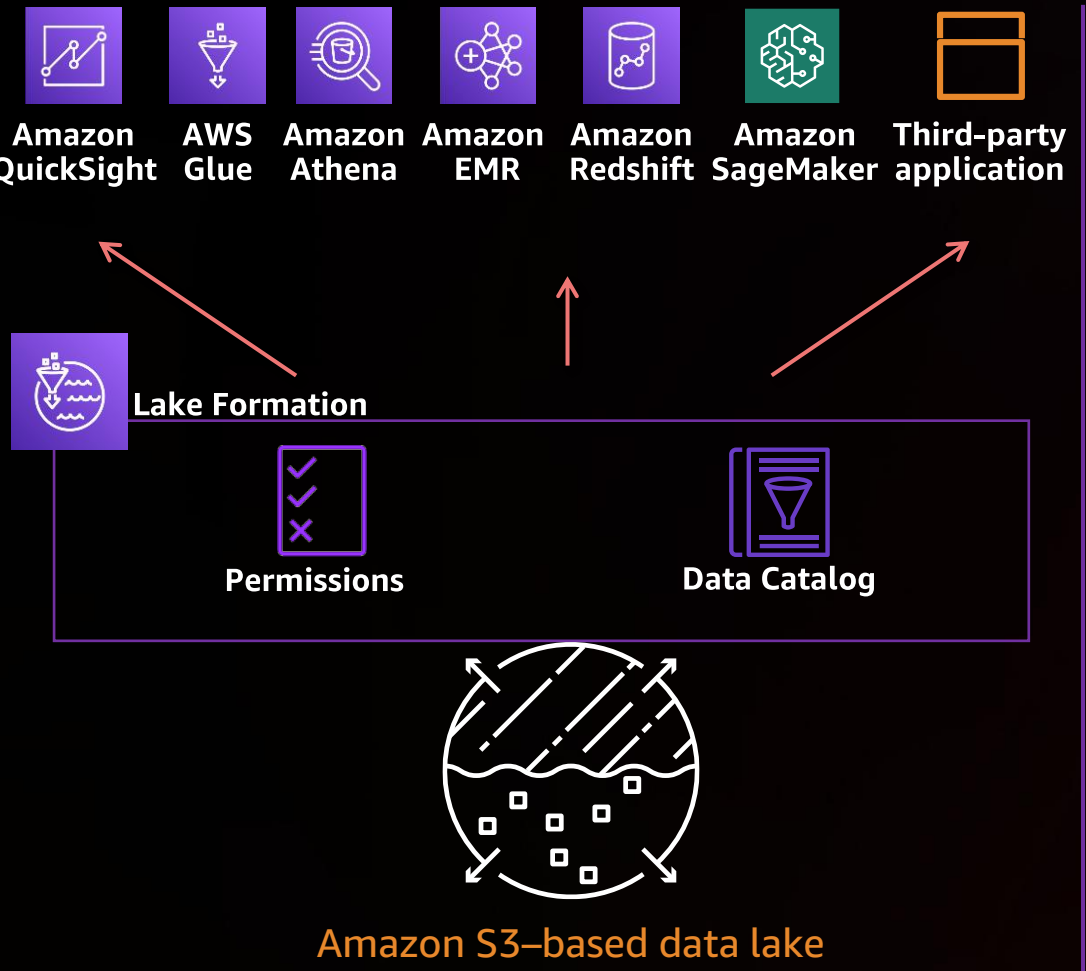
Data governance is the combination of people, processes, and technology that organizations use to ensure the quality and security of their data throughout its lifecycle



AWS Services

Lake Formation
AWS Glue Data Catalog
Macie
CloudWatch
CloudTrail
IAM
AWS Backup
Others

Lake Formation permissions model



DB-style, fine-grained permissions on resources

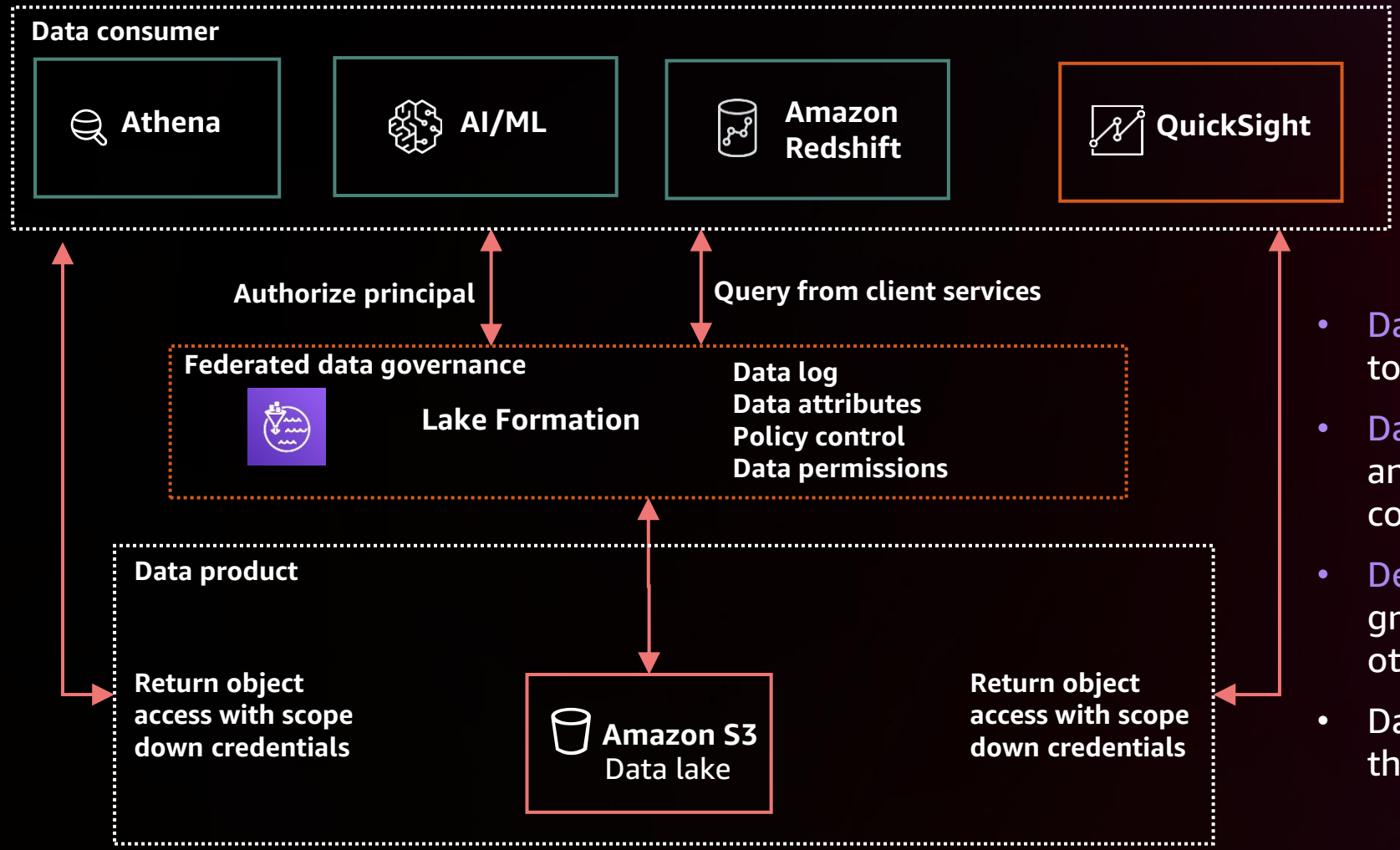
Scale permissions management: Lake Formation tag-based access control (LF-TBAC)

Unified Amazon S3 permissions

Integrated with services and tools

Intuitive permissions and access auditing

Data governance with data mesh architecture



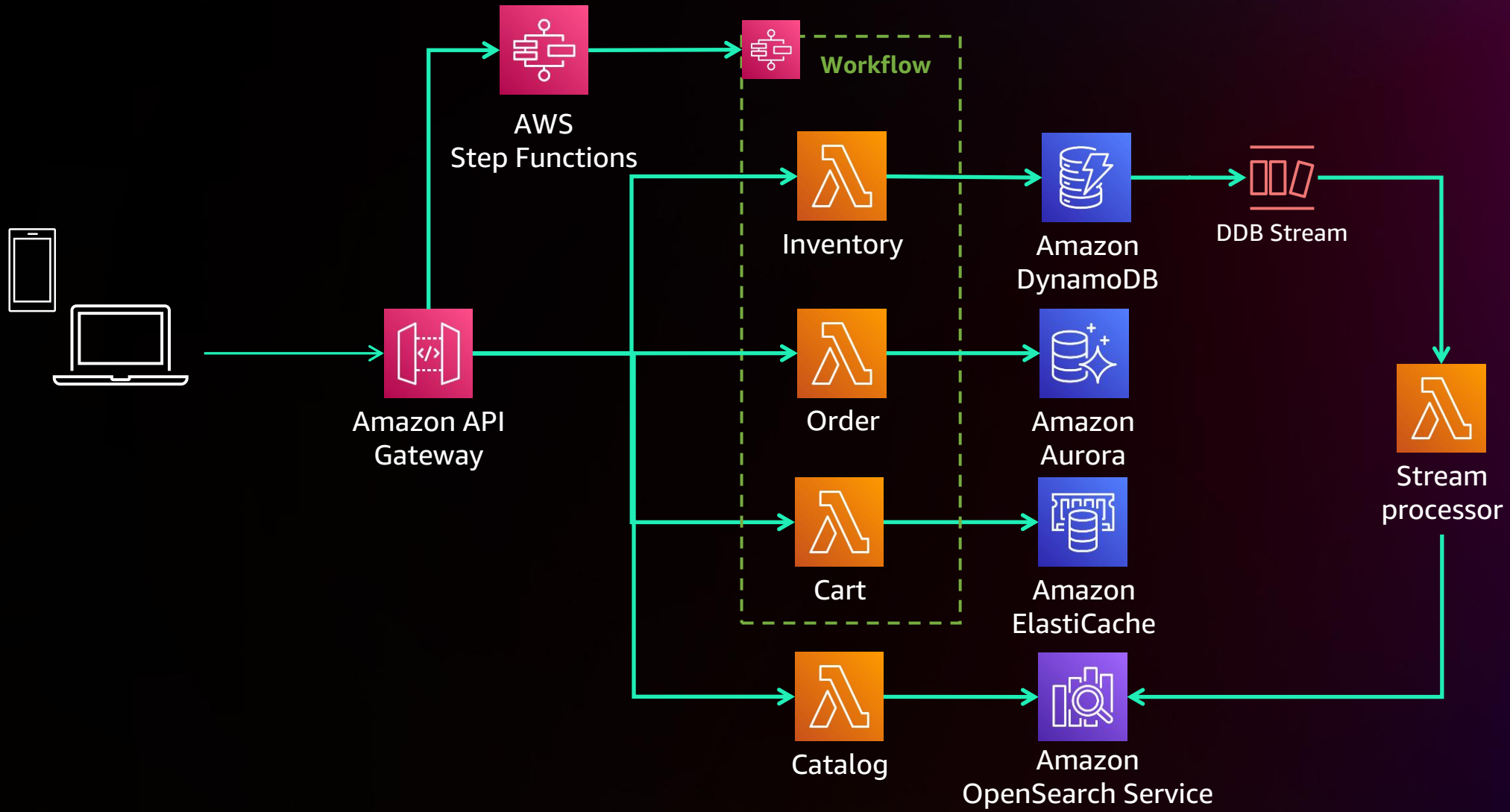
Data consumer

- Data consumer finds and requests access to a data product
- Data scientist receives approved request and initiates a resource share with consumer account
- Delegate permission grants at fine-grained level to data products shared to other consumer personas like analyst
- Data access can be validated and audited through federated data governance

Reference architectures for common scenarios

Modern applications with AWS

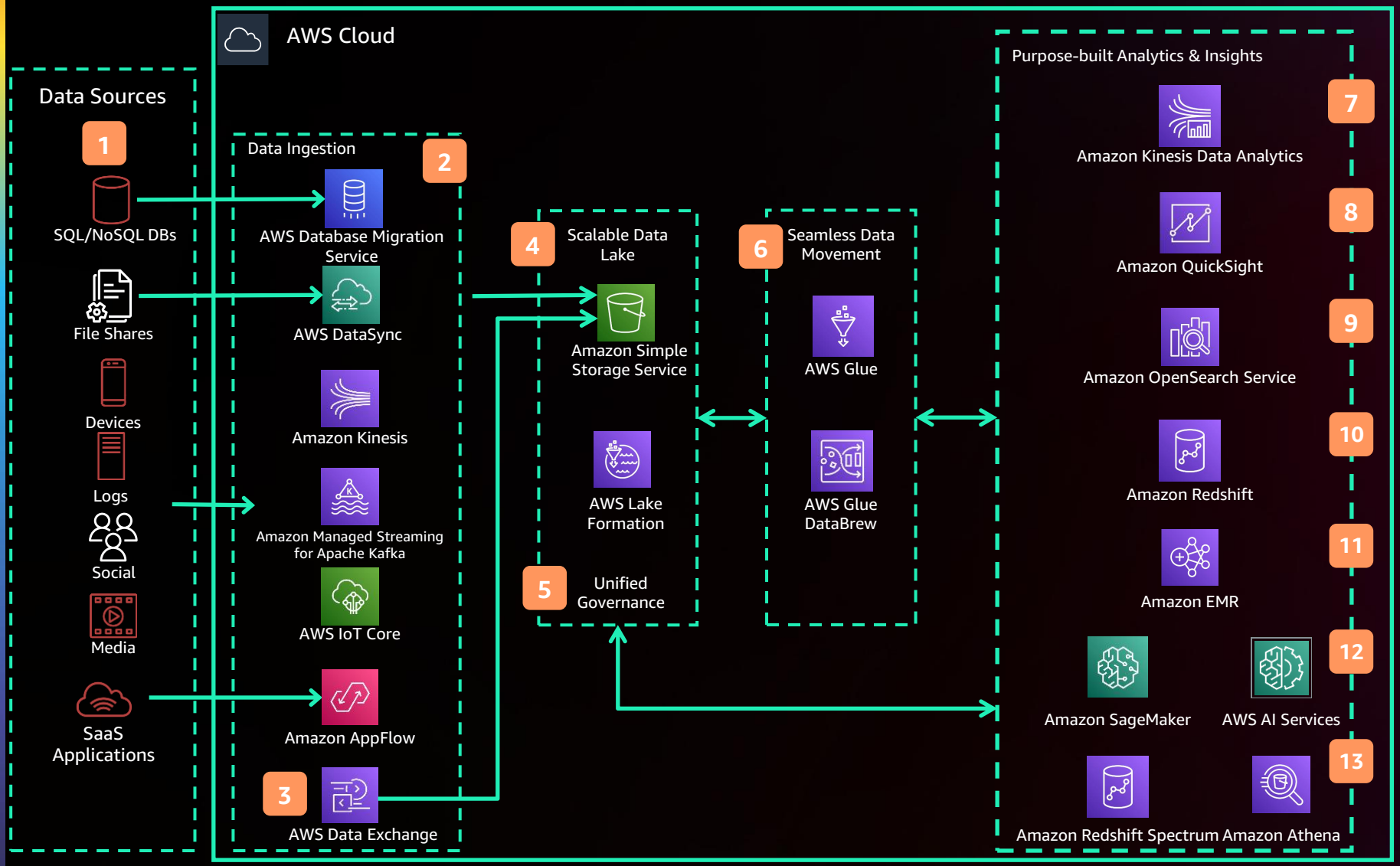
DECOUPLE YOUR DATA WITH PURPOSE-BUILT DATABASES



https://go.aws/3u59jTS

Modern data analytics reference architecture on AWS

BUILD DATA ANALYTICS PIPELINES USING MODERN DATA ANALYTICS APPROACH TO DERIVE INSIGHTS FROM THE DATA



1. Data is collected from multiple data sources across the enterprise, SaaS applications, edge devices, logs, streaming media, and social network
2. Based on the type of the data source, **AWS Database Migration Service, AWS DataSync, Amazon Kinesis, Amazon Managed Streaming for Apache Kafka, AWS IoT Core**, and **Amazon AppFlow** are used to ingest the data into a Data Lake in AWS
3. **AWS Data Exchange** is used for integrating third-party data into the Data Lake
4. **AWS Lake Formation** is used to build the scalable data lake, and **Amazon S3** is used as the data lake storage
5. **AWS Lake Formation** is also used to enable unified governance to centrally manage the security, access control, and audit trails
6. **AWS Glue, AWS Glue DataBrew,** and **AWS Glue Elastic Views** are used to transform, enrich, move, and replicate data across multiple data stores and the data lake
7. **Amazon Kinesis Data Analytics** is used to transform and analyze streaming data in real time
8. **Amazon QuickSight** provides machine learning-powered business intelligence
9. **Amazon OpenSearch Service** can be used for operational analytics
10. **Amazon Redshift** is used as a Cloud Data Warehouse
11. **Amazon EMR** provides the cloud big data platform for processing vast amounts of data using open source tools
12. **Amazon SageMaker** and **AWS AI services** can be used to build, train, and deploy machine learning models, and add intelligence to your applications
13. **Amazon Redshift Spectrum** and **Amazon Athena** enable interactive querying, analyzing, and processing capabilities

AWS Reference Architecture

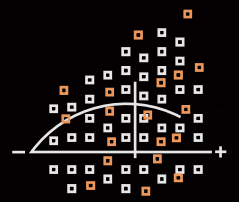
http://bit.ly/3VoIVQE

Modern data streaming architecture

INGEST, PROCESS, AND ANALYZE HIGH VOLUMES OF HIGH-VELOCITY DATA FROM VARIOUS SOURCES IN REAL TIME

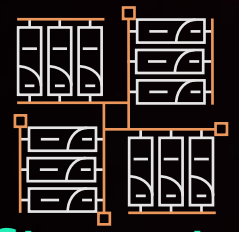
Source

- IOT sensors
- Enterprise apps
- Social media
- Logs



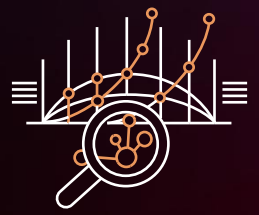
Stream ingestion

- AWS IoT Core
- Amazon Kinesis Agent
- AWS SDK
- AWS Database Migration Service (AWS DMS)
- Amazon MSK connect



Stream storage

- Amazon Kinesis Data Streams
- Amazon Managed Streaming for Apache Kafka (Amazon MSK)



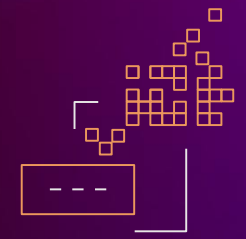
Stream processing

Stream processing

- Amazon Kinesis Data Analytics
- AWS Lambda
- Amazon EMR
- AWS Glue

Stream integration

- Amazon Kinesis Data Firehose
- Amazon S3
- Amazon Redshift
- Amazon OpenSearch Service



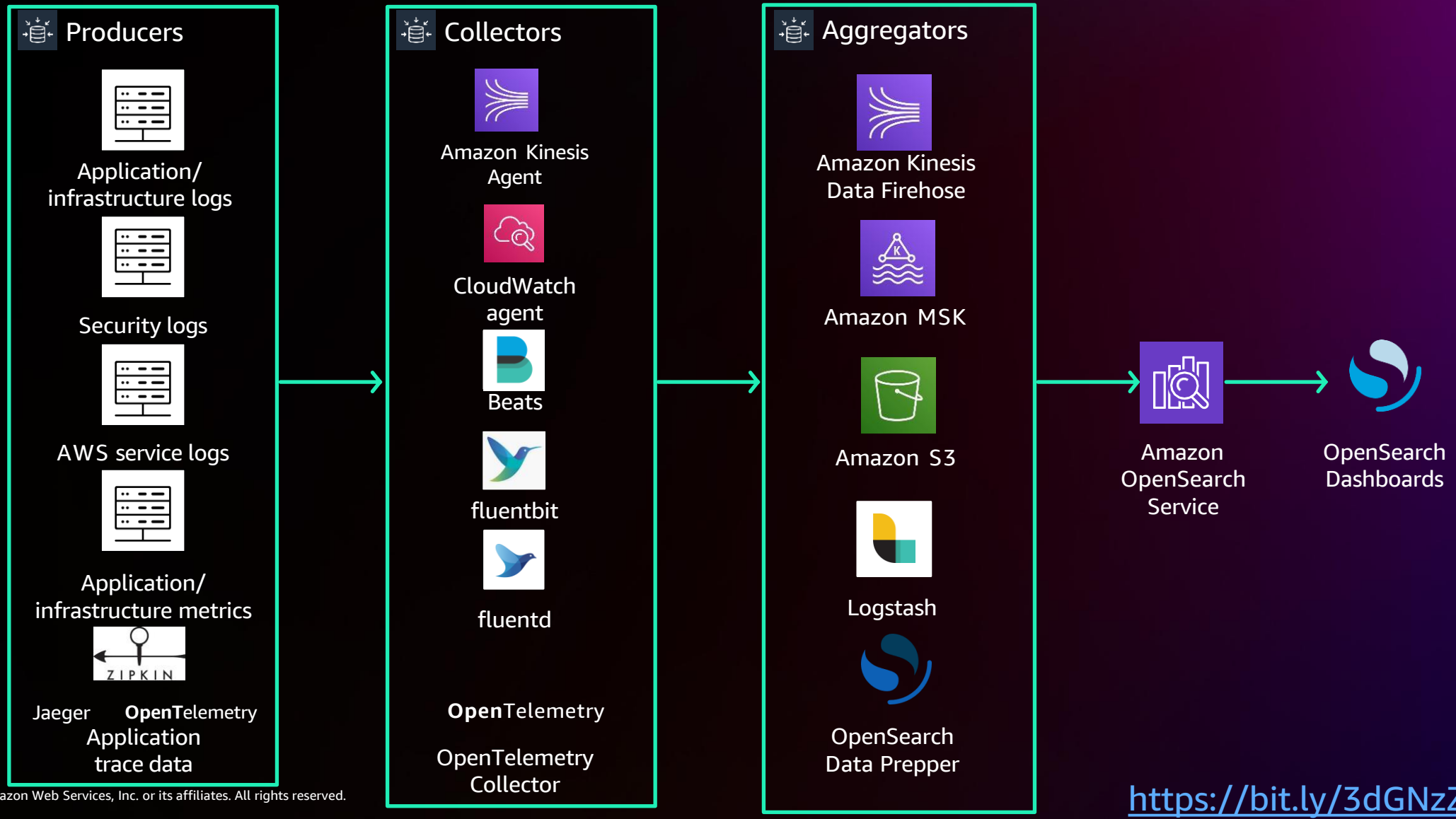
Destination

- Automatic decision
- Interactive dashboard
- Alerting
- Real-time ML inference

https://go.aws/3bt0HAm

Operational analytics reference architecture

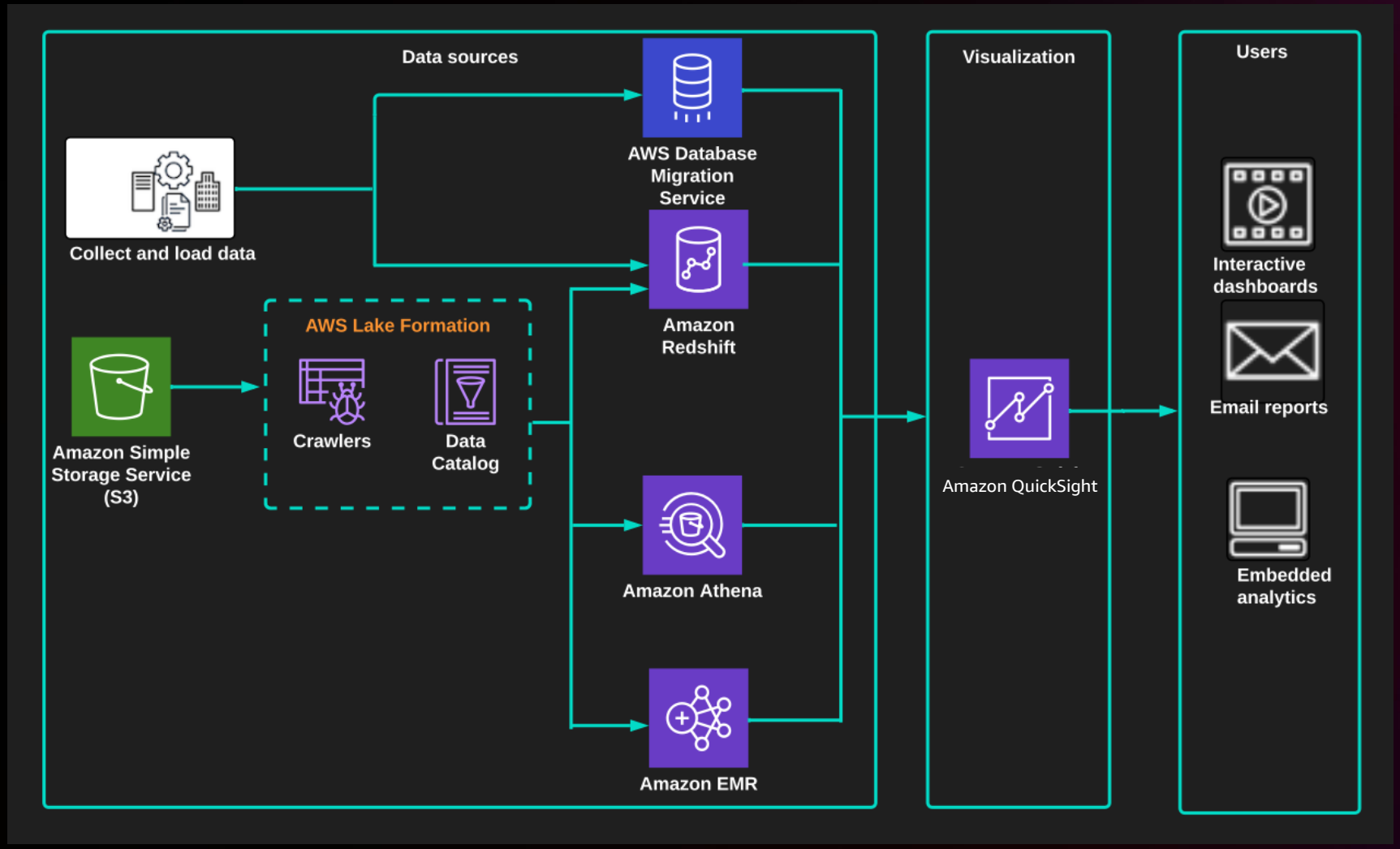
TECHNIQUES TO IMPROVE DAY-TO-DAY BUSINESS PERFORMANCE IN BUSINESS PROCESSES



https://bit.ly/3dGNzZH

Business intelligence reference architecture

INTERACTIVE DATA VISUALIZATION TO ACCELERATES DATA-DRIVEN INSIGHTS

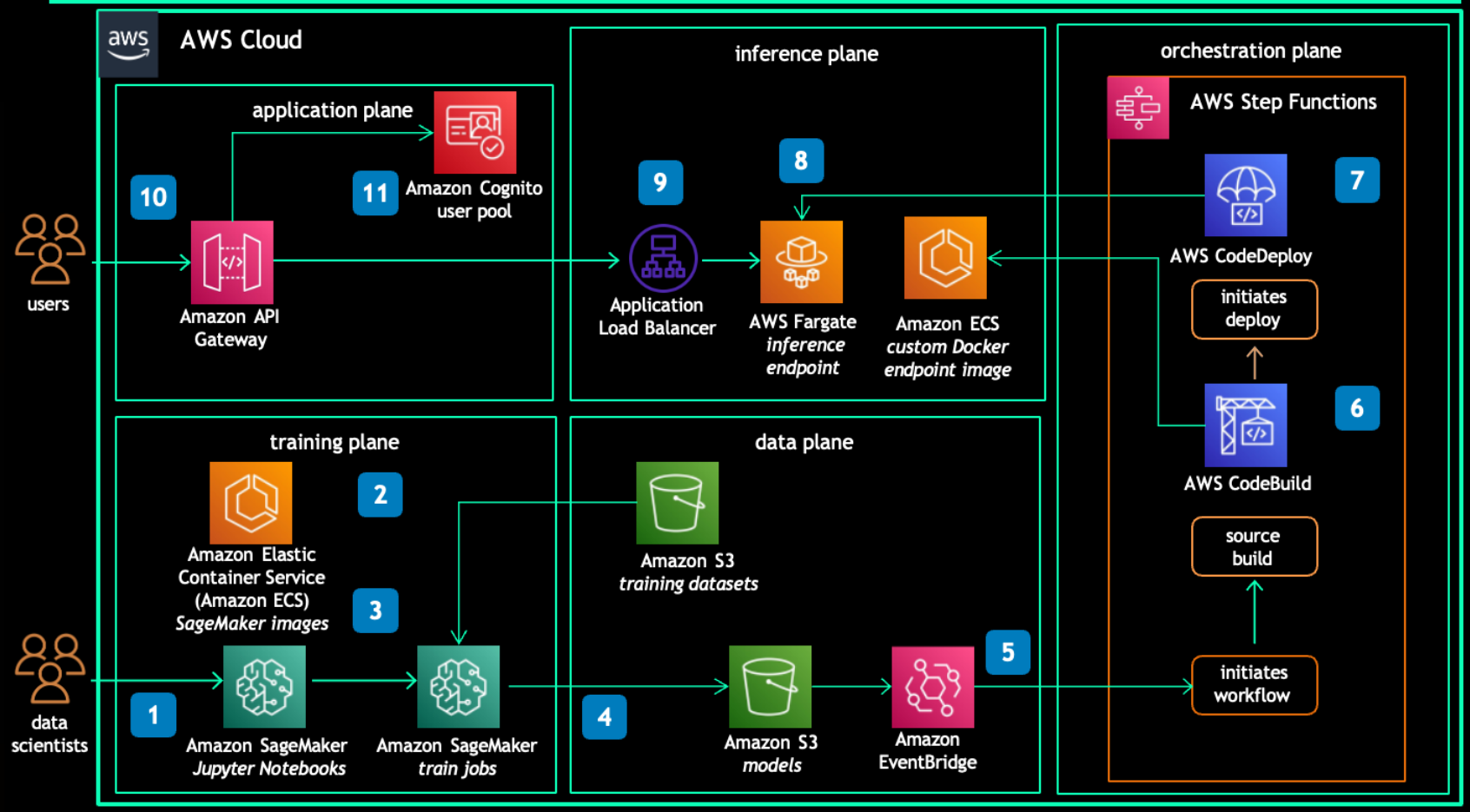


https://go.aws/3OHdi2f

Build end-to-end machine learning using Amazon SageMaker

Enhance Existing ML Lifecycles with Amazon SageMaker Training and AWS Fargate

Enhance your existing machine learning (ML) workflow by integrating with SageMaker model training features while preserving the rest of your custom serverless endpoints with Fargate. This reference architecture illustrates how to integrate SageMaker with other compute services when you do not use SageMaker for your full ML lifecycle. Fargate lets you maintain a serverless approach while enabling a higher memory model and concurrency limits with no changes to your code.



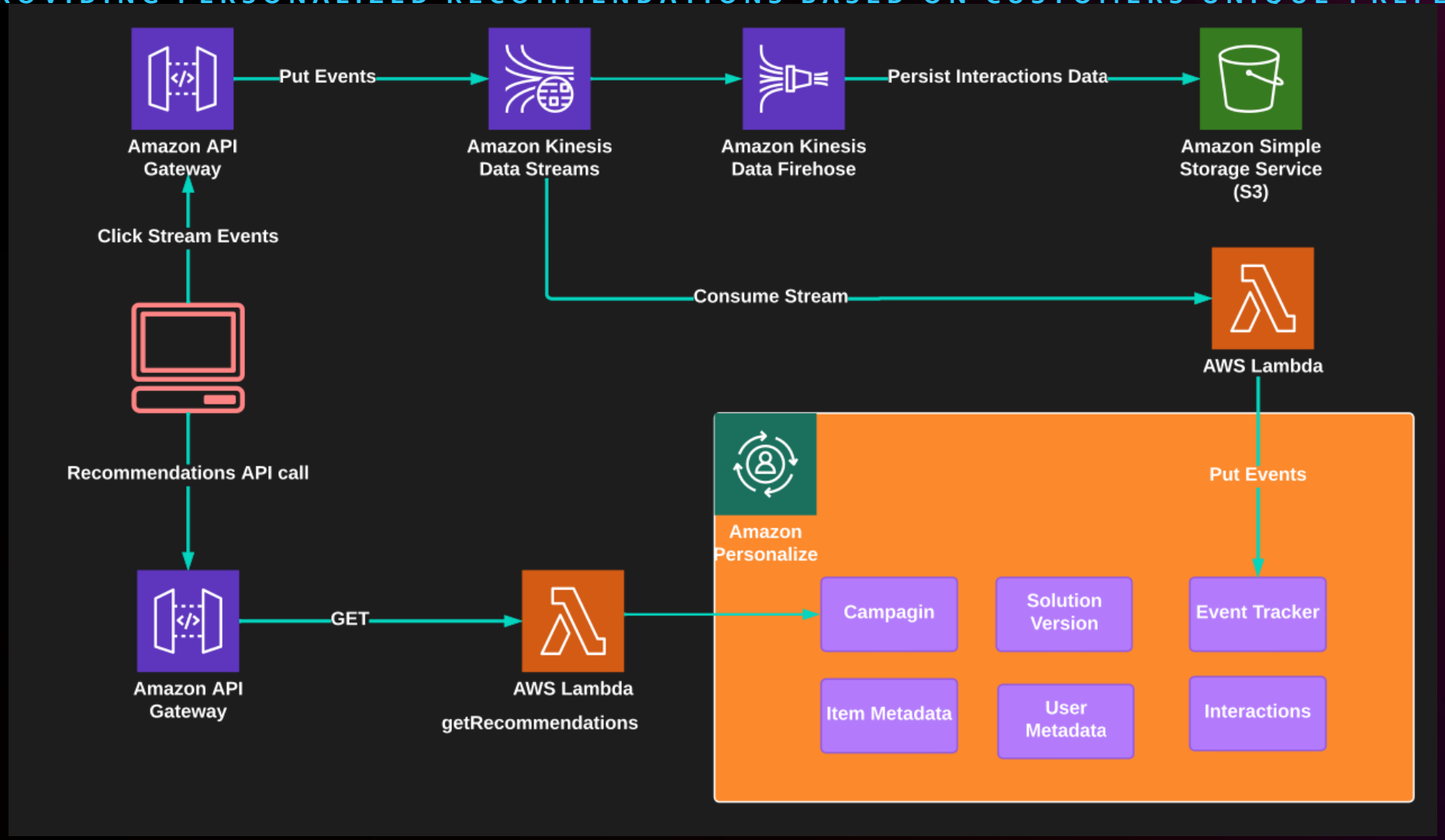
AWS Reference Architecture

1. instances for data scientists to prepare their data and launch **SageMaker** training jobs.
2. By selecting the desired algorithm, **SageMaker** image, and configuring the appropriate parameters, you can run multiple **SageMaker** training jobs and benefit from many other **SageMaker** features, such as automated hyperparameter tuning.
3. If you need additional algorithms or have pre-existing code, you can build your own **SageMaker** Docker images.
4. As a result of the training jobs, a trained model artifact is generated and stored in a dedicated **Amazon Simple Storage Service (Amazon S3)** bucket. When the data scientist is satisfied with the level of performance, the model is uploaded to a dedicated path in the bucket.
5. Using **Amazon EventBridge**, you can easily connect to the rest of the ML lifecycle and initiate the remaining lifecycle tasks outside of **SageMaker**.
6. The **AWS Step Functions** workflow orchestrates all the steps of the remaining ML pipeline using **AWS Lambda** functions. The **AWS CodeBuild** job generates custom Docker images with the correct algorithm, framework, and model, as well as a web service wrapper for the inference endpoint to be created in.
7. The new container image is tested and deployed into **AWS Fargate** via **AWS CodeDeploy** using blue/green deployment.
8. The **SageMaker** trained model can now run in your pre-existing compute environment; in this case, Docker containers that were deployed in **AWS Fargate** with **ECS** service automatic scaling in place.
9. An **Application Load Balancer** fronts **Fargate** to balance the incoming load with automatic horizontal scaling based on model endpoint latency.
10. The model is consumed by the end user browser or mobile application directly via **API Gateway**.
11. Authentication and authorization of the **API Gateway** endpoint is managed through an **Amazon Cognito** user pool integrated with **API**

http://bit.ly/3gBTSzw

Building real-time recommendations

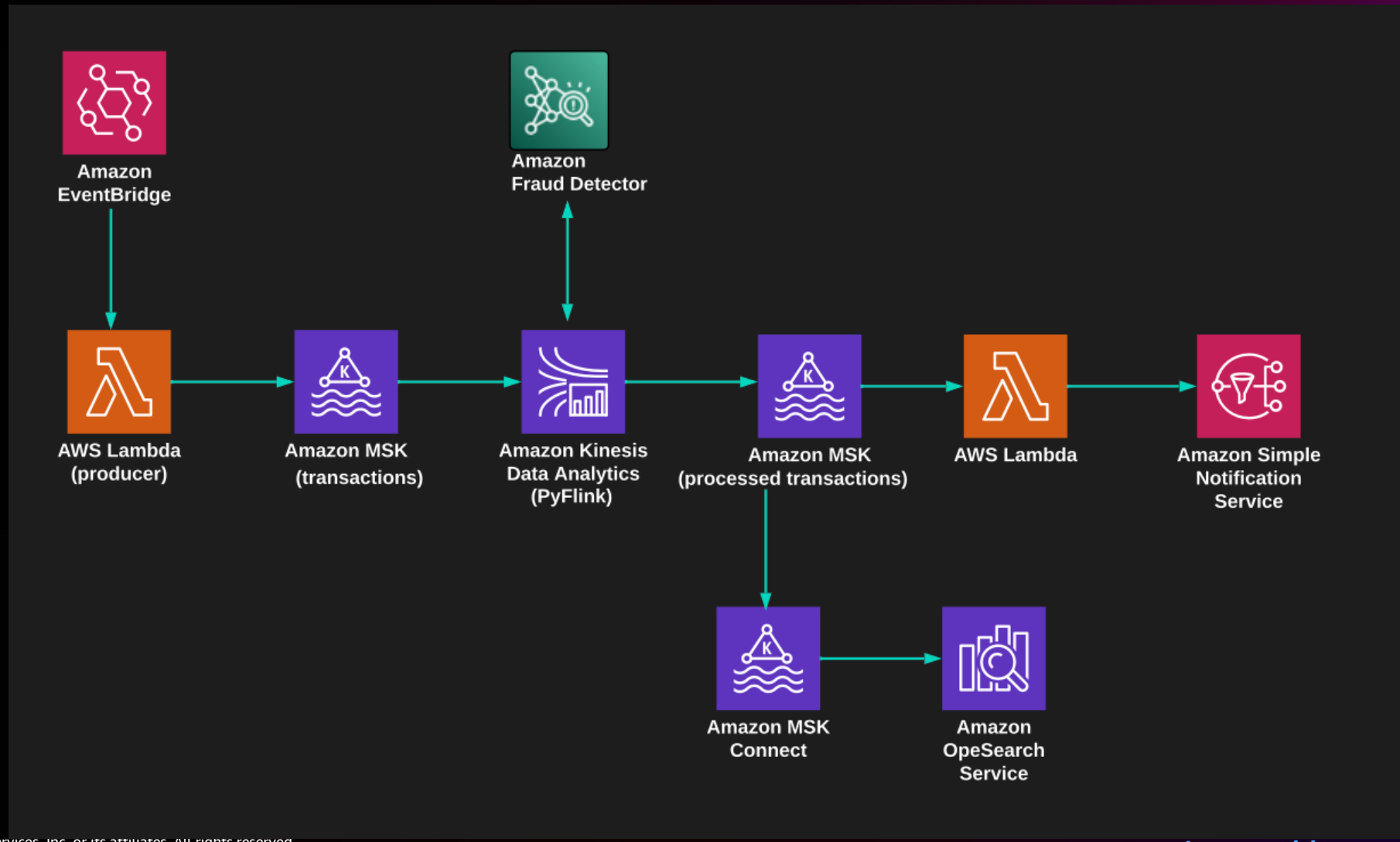
PROVIDING PERSONALIZED RECOMMENDATIONS BASED ON CUSTOMERS UNIQUE PREFERENCES AND BEHAVIOR



https://go.aws/3u3Gg38

Building a fraud detection system on AWS

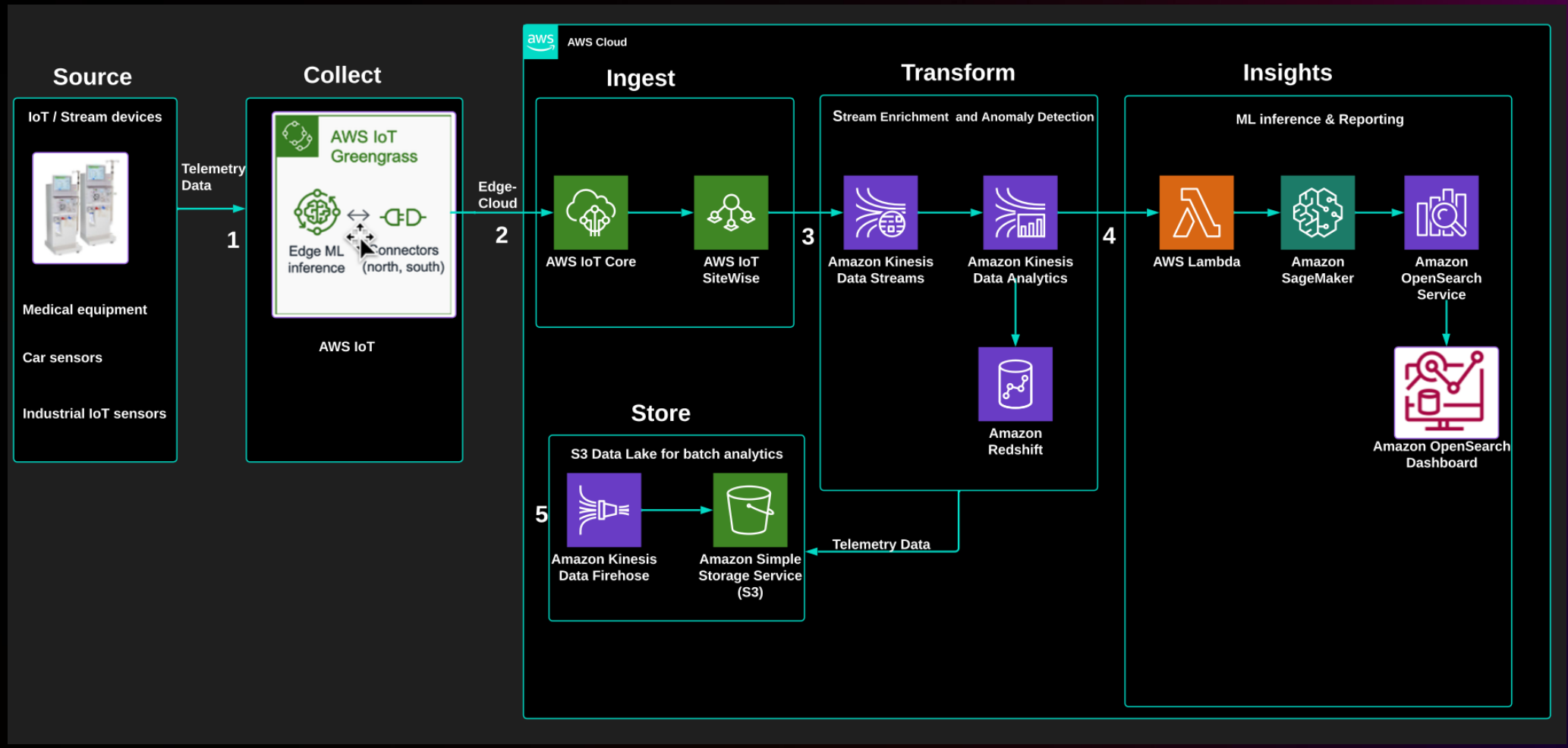
IDENTIFYING FRAUDULENT TRANSACTIONS, FLAGGING THEM TO BE BLOCKED, AND SENDING AN ALERT NOTIFICATION



https://go.aws/3wdwv3A

Building event-driven architectures with IoT sensor data

IOT BRINGS SENSORS, CLOUD COMPUTING, ANALYTICS, AND PEOPLE TOGETHER TO IMPROVE PRODUCTIVITY AND EFFICIENCY

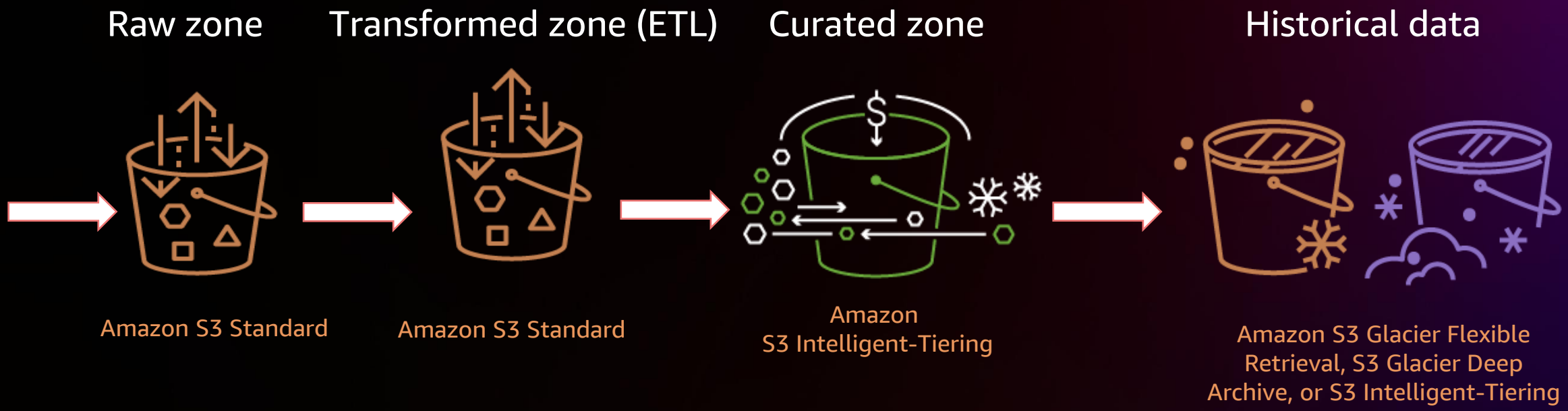


https://go.aws/3gw193T

Best practices & key takeaways

Best practices

1. Data discovery should be your first step in building modern data architectures
2. You must define business value and then identify your data sources and user personas to achieve desired business outcome
3. Migrate and modernize your operational databases into cloud with purpose-build databases to reduce operational overhead and to improve your applications performance
4. Data tiering – organize the data in raw zone, transformed zone, and curated zone for your data lake storage. Keep untouched original raw data, ideally with no transformations applied. It will be treated as golden source in case of reprocessing needs.
5. Use Amazon S3 storage classes – use Amazon S3 Standard for both raw zone and transformed zone as they are kind of temporary storage areas for data lake. We recommend using S3 Intelligent Tiering for curated zone because it automatically moves data sets between the access tiers that are not accessed for months at a time



Best practices

6. Use IAM control across for Amazon S3 and all other AWS data services. Create IAM roles for different user groups. If possible integrate it with Federated IP like Active Directory.
7. Encrypt your data at rest and in-motion
8. You can use IAM coarse-grained access controls and AWS Lake Formation for fine access controls for your data lake access management
9. Store the data in the optimum format

File Format	Properties	Use Cases
Orc	Columnar, schema stored in footer	Read heavy analytics workloads, e.g., Hive Tables
Parquet	Columnar, schema stored in footer	Read heavy analytics workloads, e.g., Spark processing
Avro	Row-major, schema and data separate	Write heavy workloads, e.g., Apache Kafka
CSV	Human readable , fixed schema	Small volumes, consumer is an analyst
JSON	Human readable , flexible schema	Small volumes, consumer is an application

10. Adjust file size. We recommend that you batch small objects into large objects to reduce the total number of requests, and therefore it will decrease your requests costs.

Best practices

11. Automate data pipelines by AWS Step functions or AWS Glue workflows or Amazon Managed Workflows for Apache Airflow. Monitor data pipelines with CloudWatch dashboards and setup alarm for for unnormal behavior, i.e., “no data ingested in last 6 hours, etc.

12. Choose your data processing services based on your specific use case and business requirements

AWS Lambda

Simple programming interface and scaling

- Serverless functions
- Six languages
- Event-based, stateless processing
- Continuous and simple scaling mechanism

Kinesis Data Analytics

Easy and powerful stream processing

- Serverless applications
- Apache Flink
- Stateful processing with automatic backups
- Stream operators make building app easy

Amazon EMR

Flexibility and choice for your needs

- Choose your instances
- Use your favorite open-source framework
- Fine-grained control over cluster, debugging tools, and more
- Deep open-source tool integrations with AWS

AWS Glue

Serverless data integration

- Choose if you are already using AWS Glue or Apache Spark
- You need to process data in batch, streaming, and event modes
- You want to build your streaming jobs visually
- Near real-time use cases – your SLA is 1 second or more

Best practices

13. Choose your streaming services based on your specific use case and business requirements

Amazon Kinesis Data Streams

- Streams and shards
- Throughput provisioning model
- Seamless scaling
- Typically lower cost
- AWS application programming interface experience
- Deep AWS integrations

Amazon MSK

- Topics and partitions
- Cluster provisioning model
- Scaling isn't seamless to clients
- Raw performance
- Open-source compatibility
- Strong third-party tooling

14. Choose right Adhoc query engine based on your use case and business requirements

Amazon Athena

- Ad-hoc querying
- Serverless – no management of clusters
- Run queries using the standard SQL

Amazon EMR

- Data processing and ETL
- Build your own clusters or use EMR Serverless
- Run custom applications and code using big data processing frameworks such as Spark, Hadoop, Presto, Hbase, etc.

Amazon Redshift

- Data warehouse for historical analysis and reporting
- Build your own clusters or use Redshift Serverless
- Run queries against highly structured data with many joins

Best practices

15. When to use data mesh architecture

Pros

- Distributed control
- Solutions aligned with business needs
- Data ownership and accountability
- Encourage product thinking

Cons

- Distributed knowledge and control
- Product thinking is not for everyone
- Building data as a product is not simple
- Not always good for small organizations

16. Before framing your business problem, first consider whether ML is an appropriate solution to your problem

17. Use the right tool for the right job by leveraging AWS purpose-built data services. Refer to respective AWS purpose-built data services best practices

- How to plan your database migration & modernization https://aws.amazon.com/blogs/architecture/selecting-the-right-database-and-database-migration-plan-for-your-workloads/
- Database migration best practices https://d1.awsstatic.com/whitepapers/Migration/migrating-applications-to-aws.pdf
- AWS Glue https://aws.amazon.com/blogs/big-data/best-practices-to-scale-apache-spark-jobs-and-partition-data-with-aws-glue/
- Amazon Athena https://aws.amazon.com/blogs/big-data/top-10-performance-tuning-tips-for-amazon-athena/
- Amazon EMR https://aws.amazon.com/blogs/big-data/best-practices-to-optimize-data-access-performance-from-amazon-emr-and-aws-glue-to-amazon-s3/

Best practices

- **Amazon Redshift** https://aws.amazon.com/blogs/big-data/best-practices-to-optimize-your-amazon-redshift-and-microstrategy-deployment/
- **Amazon OpenSearch Service** https://aws.amazon.com/blogs/big-data/best-practices-for-configuring-your-amazon-opensearch-service-domain/
- **Amazon Kinesis Data Streams** https://aws.amazon.com/blogs/big-data/best-practices-for-consuming-amazon-kinesis-data-streams-using-aws-lambda/
- **Amazon Kinesis Data Firehose** https://docs.aws.amazon.com/firehose/latest/dev/security-best-practices.html
- **Amazon Kinesis Data Analytics** https://docs.aws.amazon.com/kinesisanalytics/latest/dev/best-practices.html
- **Amazon MSK** https://aws.amazon.com/blogs/big-data/best-practices-from-delhivery-on-migrating-from-apache-kafka-to-amazon-msk/
- **Amazon QuickSight** https://aws.amazon.com/blogs/big-data/tips-and-tricks-for-high-performant-dashboards-in-amazon-quicksight/
- **Amazon SageMaker** https://docs.aws.amazon.com/sagemaker/latest/dg/best-practices.html
- **Amazon Personalize** https://aws.amazon.com/blogs/machine-learning/optimize-personalized-recommendations-for-a-business-metric-of-your-choice-with-amazon-personalize/
- **Amazon Forecast** https://aws.amazon.com/blogs/machine-learning/tailor-and-prepare-your-data-for-amazon-forecast/
- **Amazon Fraud Detector** https://aws.amazon.com/blogs/machine-learning/catching-fraud-faster-by-building-a-proof-of-concept-in-amazon-fraud-detector/

18. AWS Well-Architected Framework Lens for data analytics and machine learning for a collection of proven best practices

- **Data Analytics Lens**, we describe a collection of customer-proven best practices for designing well-architected analytics workloads https://docs.aws.amazon.com/wellarchitected/latest/analytics-lens/analytics-lens.html
- **Machine Learning Lens**, we focus on how to design, deploy, and architect your machine learning workloads in the AWS Cloud https://docs.aws.amazon.com/wellarchitected/latest/machine-learning-lens/machine-learning-lens.html

Best practices

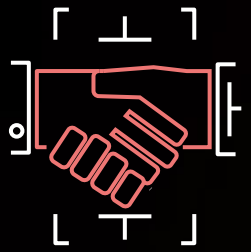
19. Empowers all personas

Personas	Responsibility	Areas of Interest	AWS data services
Chief Data Officer (CDO)	Build a culture of using data to solve problems and accelerate innovation	Data quality, data governance, data/AI strategy, evangelize the value of data to business	Amazon QuickSight, AWS Lake Formation, Amazon OpenSearch Service
Data Architect	Driven to architect technical solutions to meet business needs	Data processing pipelines and data integration, data governance, and data catalogs	AWS Glue, Amazon EMR, Amazon Redshift, Amazon Athena, Amazon OpenSearch Service
Data Engineer	Deliver usable, accurate dataset to organization in a secure and performant manner	Variety of tools to build data pipeline, ease of use, configuration, and maintenance	AWS Glue, Amazon EMR, Amazon Kinesis, Amazon Redshift, Amazon Athena, Amazon OpenSearch Service
Data Security Officer	Data security, privacy, and data governance	Keeping information secure. Comply with data privacy regulations	AWS Lake Formation, AWS Identity and Access Management (IAM)
Data Scientist	Construct the means for extracting business focused insights from data	Tools that simplify data manipulation, and tools that help build ML pipeline	Amazon SageMaker, Amazon Athena, AWS Glue DataBrew
Data Analyst	React to market conditions in real time, need to have the ability to find data, and perform analytics quickly and easily	Querying data and performing analysis to create new business insights, producing reports and visualizations that explain the business insights	Amazon Athena, Amazon QuickSight, AWS Glue Studio, Amazon Redshift

20. Build automation across your data and ML pipelines

Accelerate your modern data strategy on AWS

GETTING STARTED: NEXT STEPS



MODERNIZE

DB Freedom Program

Migration Assistance Program

ProServe/Partner Accelerators

UNIFY

MAP for Analytics

Data-Driven Everything (D2E)

Data Labs

Immersion Days

INNOVATE

AWS Machine Learning Embark Program

ML Solutions Lab

ProServe/Partner Accelerators

Learn more

1. Derive Insights from Modern Data
https://go.aws/3xVU3dn

2. Build Modern Data Streaming Architectures on AWS
https://go.aws/3bt0HAm

3. Architectural Patterns to Build End-to-End Data Driven Applications on AWS
https://docs.aws.amazon.com/whitepapers/latest/build-e2e-data-driven-applications/build-e2e-data-driven-applications.html

4. Big Data Analytics Options on AWS
https://docs.aws.amazon.com/whitepapers/latest/big-data-analytics-options/welcome.html

5. Build a Secure Enterprise Machine Learning Platform on AWS
https://docs.aws.amazon.com/whitepapers/latest/build-secure-enterprise-ml-platform/build-secure-enterprise-ml-platform.html?mld_prac6

6. Migrate oracle databases to AWS
https://docs.aws.amazon.com/whitepapers/latest/strategies-migrating-oracle-db-to-aws/data-migration-methods.html

7. Migrate SQL Server databases to AWS
https://docs.aws.amazon.com/prescriptive-guidance/latest/migration-sql-server/migration-sql-server.pdf

8. Automated Data Analytics on AWS
https://aws.amazon.com/solutions/implementations/automated-data-analytics-on-aws/

9. Amazon SageMaker Examples
https://github.com/aws/amazon-sagemaker-examples

Thank you!

Raghavarao Sodabathina
https://www.linkedin.com/in/raghavaraos

Please complete the session survey in the **mobile app**