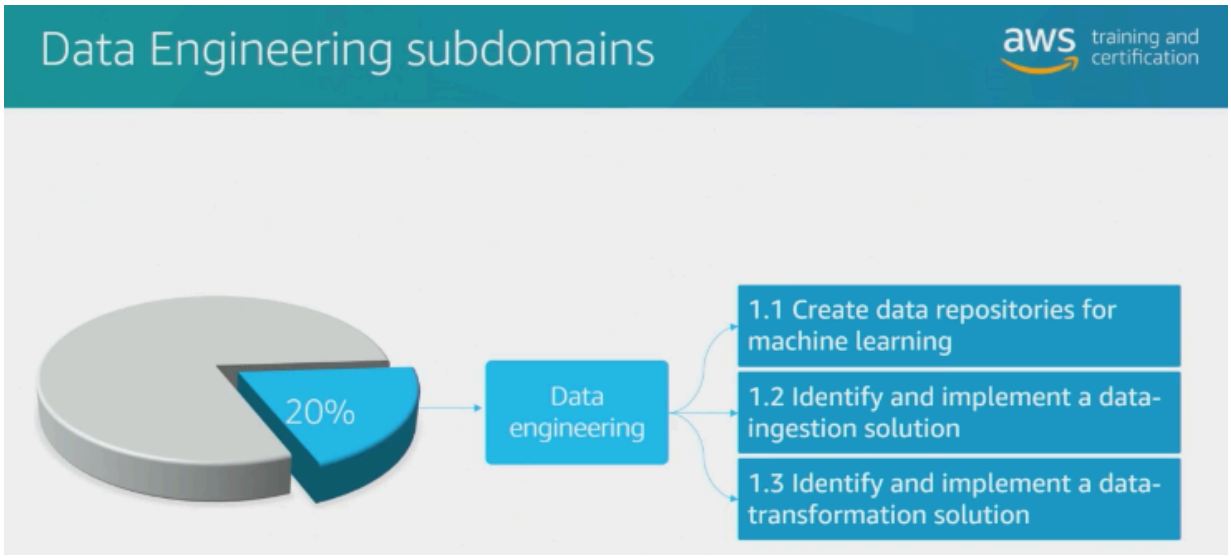
AWS Exam Readiness: Domain 1: Data Engineering

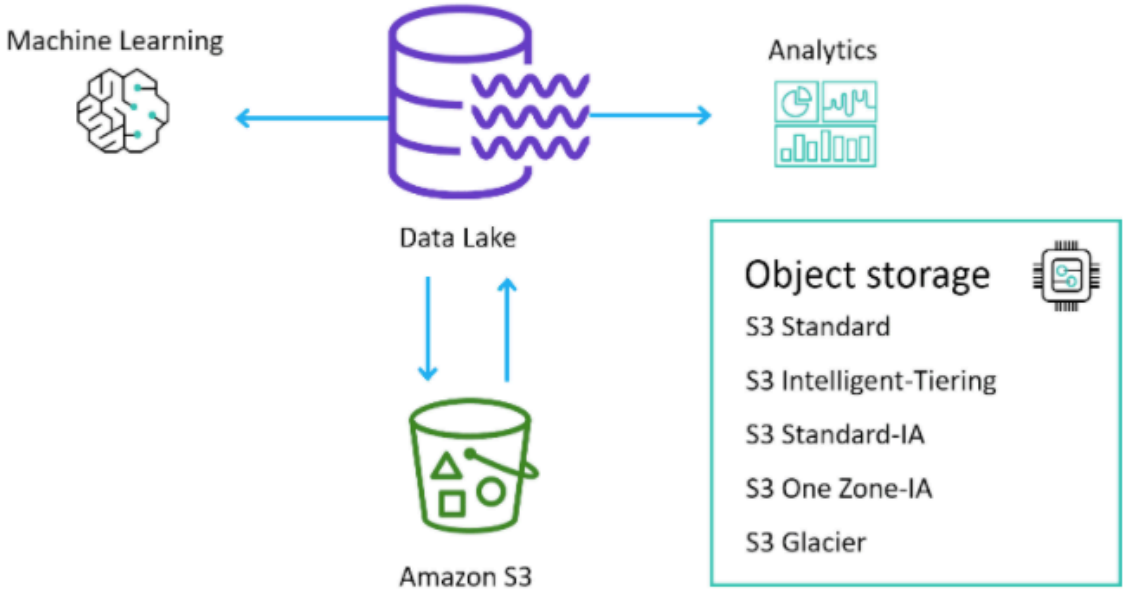
Module 3 of 9

Domain 1: Data Engineering



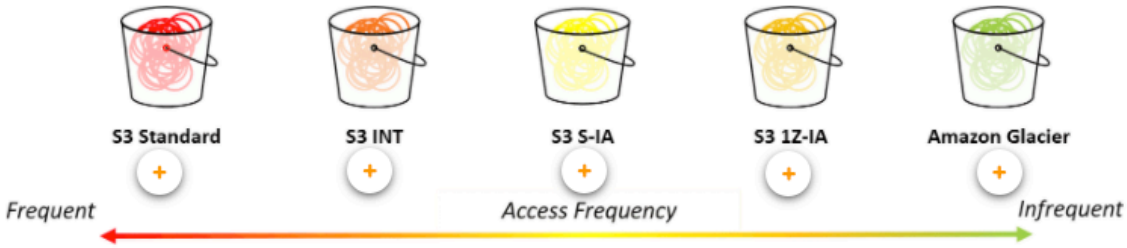
AWS Lake Formation and Amazon S3

AWS Lake Formation is your data lake solution, and Amazon S3 is the preferred storage option for data science processing on AWS



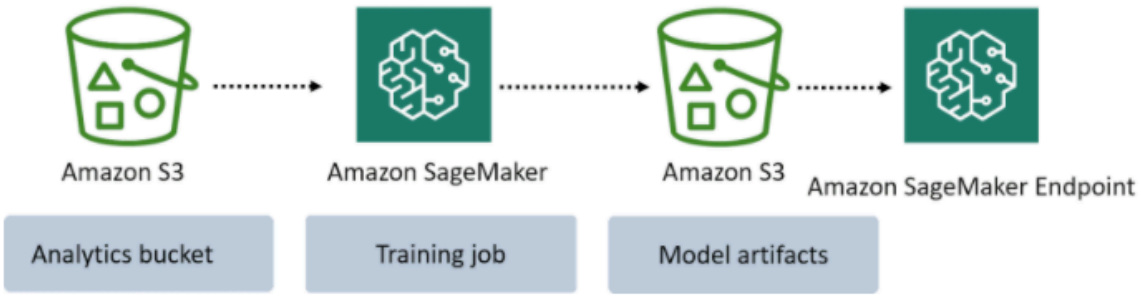
More on Amazon S3

Use Amazon S3 storage classes to reduce the cost of data storage. See the differences between those classes by clicking in the image below.



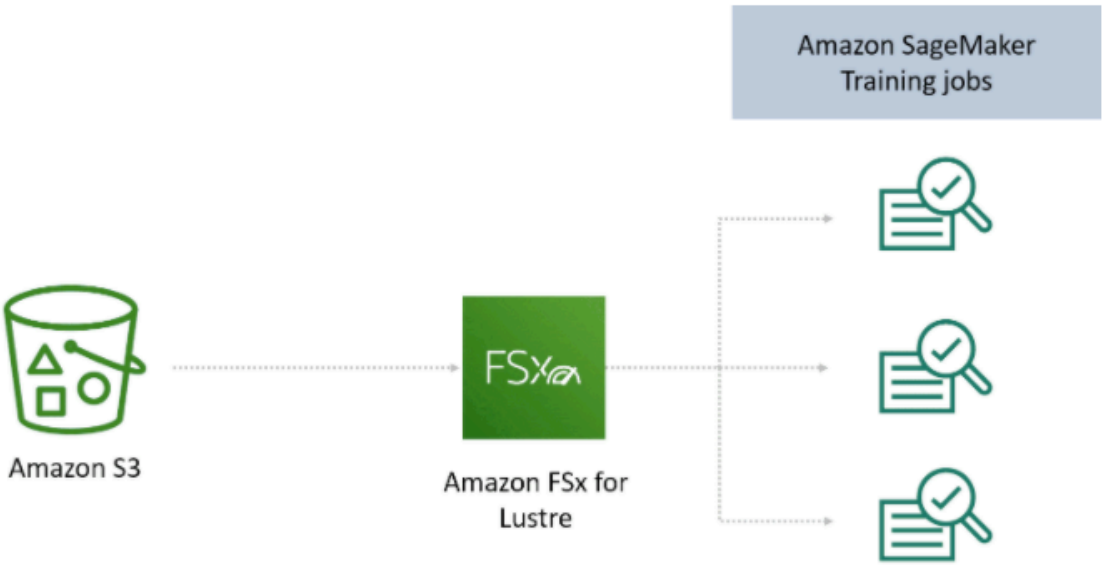
Amazon S3 with Amazon SageMaker

You can use Amazon S3 while you're training your ML models with Amazon SageMaker. Amazon S3 is integrated with Amazon SageMaker to store your training data and model training output.



Amazon FSx for Lustre

When your training data is already in Amazon S3 and you plan to run training jobs several times using different algorithms and parameters, consider using Amazon FSx for Lustre, a file system service. FSx for Lustre speeds up your training jobs by serving your Amazon S3 data to Amazon SageMaker at high speeds. The first time you run a training job, FSx for Lustre automatically copies data from Amazon S3 and makes it available to Amazon SageMaker. You can use the same Amazon FSx file system for subsequent iterations of training jobs, preventing repeated downloads of common Amazon S3 objects.



https://aws.amazon.com/fsx/

Amazon FSx for Windows File Server: feature-rich file storage for business applications

Amazon FSx for Windows File Server provides fully managed file storage that is accessible over the industry-standard Server Message Block (SMB) protocol. Built on Windows Server, Amazon FSx delivers a wide range of administrative features such as data deduplication, end-user file restore, and Microsoft Active Directory (AD) integration. It offers single-AZ and multi-AZ deployment options, fully managed backups, and encryption of data at rest and in transit. With the HDD storage option, Amazon FSx for Windows File Server offers the lowest-cost file storage in the cloud for Windows applications and workloads. You can access Amazon FSx from Windows, Linux, or MacOS compute instances and devices running on AWS or on premises. You can optimize cost and performance with multiple storage options, and you can grow your file system storage and scale your performance at any time.

Amazon FSx for Lustre: fast and scalable shared storage to power your compute workloads

Amazon FSx for Lustre is a fully managed service that provides cost-effective high-performance storage for compute workloads. Many workloads such as machine learning, high-performance computing (HPC), video rendering, and financial simulations depend on compute instances accessing the same set of data through high-performance shared storage. Powered by Lustre, the world's most popular high-performance file system, FSx for Lustre offers shared storage with low latencies, up to hundreds of gigabytes per second of throughput, and millions of IOPS. FSx for Lustre offers multiple deployment types, storage types, and throughput performance levels to optimize cost and performance for your workload requirements. FSx for Lustre file systems can also be linked to Amazon S3 buckets, allowing you to access and process data concurrently from both a high-performance file system and from the S3 API.

Amazon S3 with Amazon EFS

Alternatively, if your training data is already in Amazon Elastic File System (Amazon EFS), we recommend using that as your training data source. Amazon EFS has the benefit of directly launching your training jobs from the service without the need for data movement, resulting in faster training start times. This is often the case in environments where data scientists have home directories in Amazon EFS and are quickly iterating on their models by bringing in new data, sharing data with colleagues, and experimenting with including different fields or labels in their dataset. For example, a data scientist can use a Jupyter notebook to do initial cleansing on a training set, launch a training job from Amazon SageMaker, then use their notebook to drop a column and re-launch the training job, comparing the resulting models to see which works better.

When choosing a file system, take into consideration the training load time

The table below shows an example of some different file systems and the relative rate that they can transfer images to a compute cluster. This performance is only a single measurement and is only a suggestion for which file systems you could use for a given workload. The specifics of a given workload might change these results.

File System	Relative Speed*
Amazon S3	<1.00
Amazon EFS	1
Amazon EBS	1.29
Amazon FSx	>1.6

**Comparison of the relative (to Amazon EFS) images per second that each file system can load*

Topics related to this subdomain

Here are some topics you may want to study for more in-depth information related to this subdomain:

- [ ] AWS Lake Formation
- [ ] Amazon S3 (as storage for a data lake)
- [ ] Amazon FSx for Lustre
- [ ] Amazon EFS
- [ ] Amazon EBS volumes
- [ ] Amazon S3 lifecycle configuration
- [ ] Amazon S3 data storage options

https://aws.amazon.com/lake-formation/?whats-new-cards.sort-by=item.additionalFields.postDateTime&whats-new-cards.sort-order=desc

Creating a data lake with Lake Formation is as simple as defining data sources and what data access and security policies you want to apply. Lake Formation then helps you collect and catalog data from databases and object storage, move the data into your new Amazon S3 data lake, clean and classify your data using machine learning algorithms, and secure access to your sensitive data. Your users can access a centralized data catalog which describes available data sets and their appropriate usage. Your users then leverage these data sets with their choice of analytics and machine learning services, like Amazon Redshift, Amazon Athena, and (in beta) Amazon EMR for Apache Spark. Lake Formation builds on the capabilities available in AWS Glue.

Build data lakes quickly

With **Lake Formation**, you can move, store, catalog, and clean your data faster. You simply point Lake Formation at your data sources, and **Lake Formation crawls** those sources and **moves the data into your new Amazon S3 data lake**.

- Lake Formation organizes data in S3 around **frequently used query terms and into right-sized chunks** to increase efficiency.
- Lake Formation also changes data into formats like **Apache Parquet and ORC** for faster analytics.
- In addition, Lake Formation has **built-in machine learning to deduplicate and find matching records** (two entries that refer to the same thing) to increase data quality.

https://www.youtube.com/watch?v=hJ8Hdt7y0qw

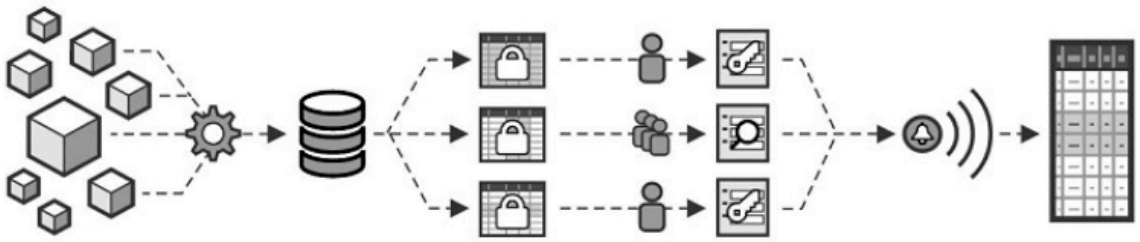
AWS Lake Formation

Build a secure data lake in days

Identify, ingest, clean, and transform data

Enforce security policies across multiple services

Gain and manage new insights



Ingest & Organize

Automatically ingest, clean, encrypt, and register existing S3 bucket content, including log data from CloudTrail, CloudFront, and Amazon ELB.

Secure & Control

Define access control that provides the right data to the right users, groups, and roles. Flexible database, table, and column permissions enable granular security.

Collaborate & Use

Search and discover using catalog metadata. All access is checked against policy, so your data is protected even if tools change or new data arrives.

Monitor & Audit

Be alerted of access requests and policy exceptions. Review activity history with detailed change logs and data lineage.

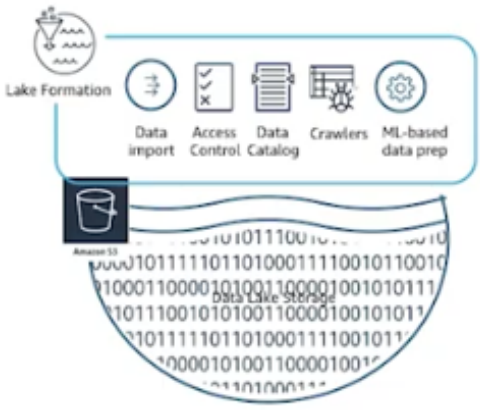
Register existing data or import new

Amazon S3 forms the storage layer for AWS Lake Formation

Register existing Amazon S3 buckets that contain your data

Ask AWS Lake Formation to create required Amazon S3 buckets and import data into them

Data is stored in your account. You have direct access to it. No lock-in.



With blueprints

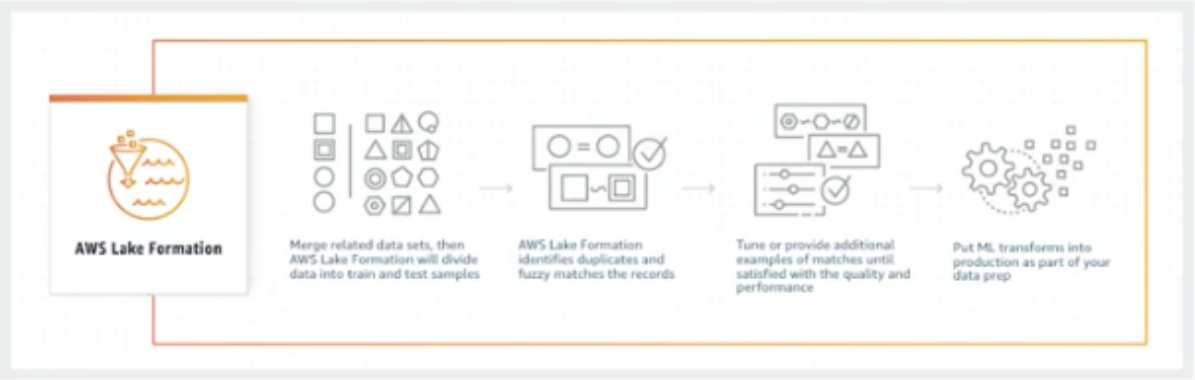
You

1. Point us to the source
2. Tell us the location to load to in your data lake
3. Specify how often you want to load the data

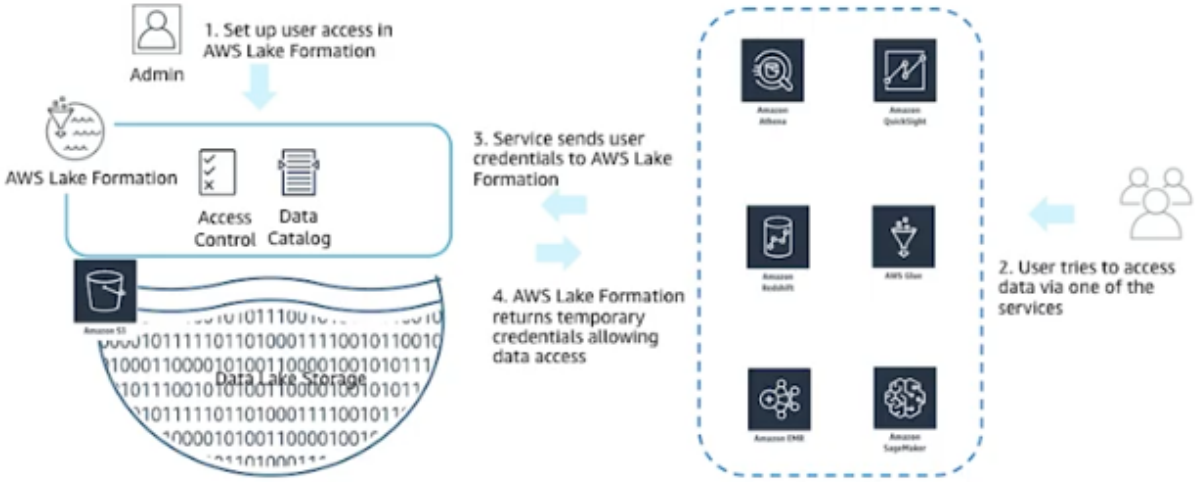
Blueprints

1. Discover the source table(s) schema
2. Automatically convert to the target data format
3. Automatically partition the data based on the partitioning schema
4. Keep track of data that was already processed
5. You can customize any of the above

Easily de-duplicate your data with ML transforms



Secure once, access in multiple ways



Security permissions in AWS Lake Formation

Control data access with simple grant and revoke permissions

Specify permissions on tables and columns rather than on buckets and objects

Easily view policies granted to a particular user

Audit all data access at one place

Grant table and column-level permissions

Amazon EFS: https://aws.amazon.com/efs/

Amazon EFS offers two storage classes: the Standard storage class, and the Infrequent Access storage class (EFS IA). EFS IA provides price/performance that's cost-optimized for files not accessed every day. By simply enabling EFS Lifecycle Management on your file system, files not accessed according to the lifecycle policy you choose will be automatically and transparently moved into EFS IA. The EFS IA storage class costs only $0.025/GB-month*.

Amazon EFS is designed to provide massively parallel shared access to thousands of Amazon EC2 instances, enabling your applications to achieve high levels of aggregate throughput and IOPS with consistent low latencies.

Amazon EFS is a regional service storing data within and across multiple Availability Zones (AZs) for high availability and durability. Amazon EC2 instances can access your file system across AZs, regions, and VPCs, while on-premises servers can access using AWS Direct Connect or AWS VPN.

Amazon EBS: https://aws.amazon.com/ebs/?ebs-whats-new.sort-by=item.additionalFields.postDateTime&ebs-whats-new.sort-order=desc

Amazon Elastic Block Store (EBS) is an easy to use, high performance block storage service designed for use with Amazon Elastic Compute Cloud (EC2) for both throughput and transaction intensive workloads at any scale. A broad range of workloads, such as relational and non-relational databases, enterprise applications, containerized applications, big data analytics engines, file systems, and media workflows are widely deployed on Amazon EBS.

You can choose from five different volume types to balance optimal price and performance. You can achieve single digit-millisecond latency for high performance database workloads such as SAP HANA or gigabyte per second throughput for large, sequential workloads such as Hadoop. You can change volume types, tune performance, or increase volume size without disrupting your critical applications, so you have cost-effective storage when you need it.

Designed for mission-critical systems, EBS volumes are replicated within an Availability Zone (AZ) and can easily scale to petabytes of data. Also, you can use EBS Snapshots with automated lifecycle policies to back up your volumes in Amazon S3, while ensuring geographic protection of your data and business continuity.

Amazon EFS vs EBS vs S3

Amazon EFS is a file storage service for use with Amazon EC2. Amazon EFS provides a file system interface, file system access semantics (such as strong consistency and file locking), and concurrently-accessible storage for up to thousands of Amazon EC2 instances.
Amazon EBS is a block level storage service for use with Amazon EC2. Amazon EBS can deliver performance for workloads that require the lowest-latency access to data from a single EC2 instance.
Amazon S3 is an object storage service. Amazon S3 makes data available through an Internet API that can be accessed anywhere.

Amazon **EBS** delivers high-availability **block-level storage** volumes for Amazon Elastic Compute Cloud (EC2) instances. It stores data on a file system which is **retained after the EC2 instance is shut down**.
Amazon **EFS** offers **scalable file storage**, also optimized for EC2. It can be used as a common data source for any application or workload that r**uns on numerous**

instances. Using an EFS file system, you may configure instances to mount the file system.
The main differences between EBS and EFS is that **EBS is only accessible from a single EC2 instance in your particular AWS region, while EFS allows you to mount the file system across multiple regions and instances.**

Finally, Amazon **S3** is an object store good at storing vast numbers of backups or user files. Unlike EBS or EFS, S3 is not limited to EC2. Files stored within an S3 bucket can be accessed programmatically or directly from services such as AWS CloudFront. This is why many websites use it to hold their content and media files, which may be served efficiently from AWS CloudFront.

Amazon S3 lifecycle configuration
An *S3 Lifecycle configuration* is a set of rules that define actions that Amazon S3 applies to a group of objects. There are two types of actions:

- **Transition actions**—Define when objects transition to another storage class. For example, you might choose to transition objects to the S3 Standard-IA storage class 30 days after you created them, or archive objects to the S3 Glacier storage class one year after creating them.
  There are costs associated with the lifecycle transition requests. For pricing information, see Amazon S3 pricing.

- **Expiration actions**—Define when objects expire. Amazon S3 deletes expired objects on your behalf.
  The lifecycle expiration costs depend on when you choose to expire objects. For more information, see Understanding object expiration.

Domain 1.2: Identify and implement a data ingestion solution

To use this data for ML, you need to ingest it into a service like Amazon S3

One of the core benefits of a data lake solution is the ability to quickly and easily ingest multiple types of data. In some cases, your data will reside outside your Amazon S3 data lake solution, in databases, on-premises storage platforms, data warehouses, and other locations. To use this data for ML, you may need to ingest it into a storage service like Amazon S3.

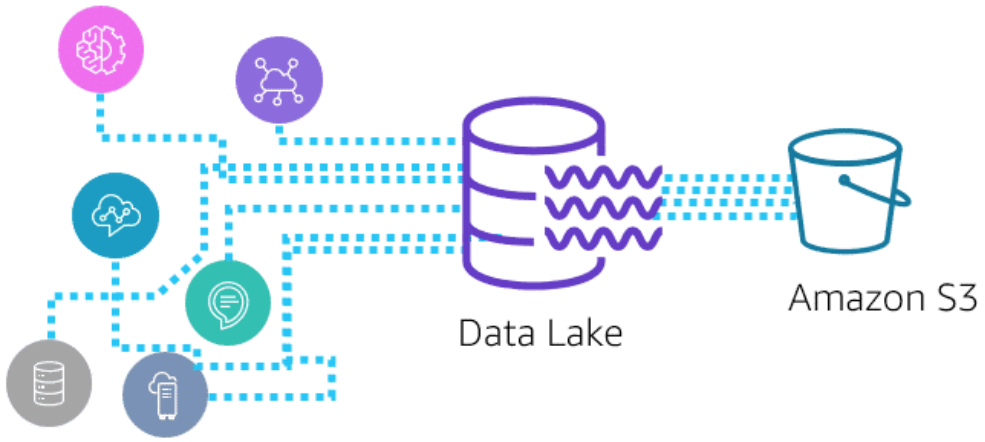
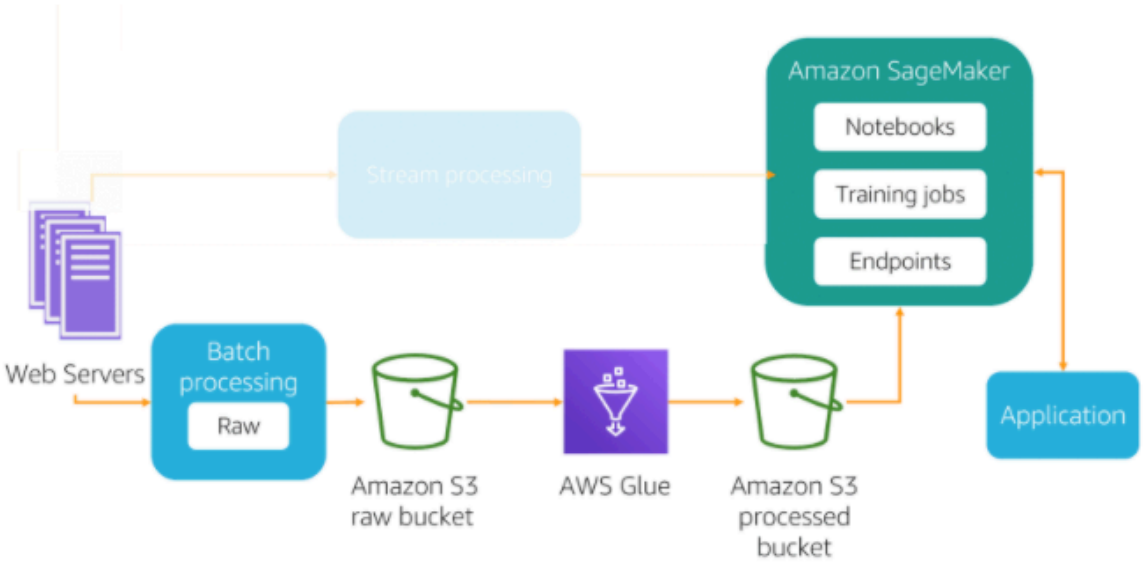


Image recapping how data coming from many different sources is ingested into Amazon S3

Several services can help with batch processing into the AWS Cloud

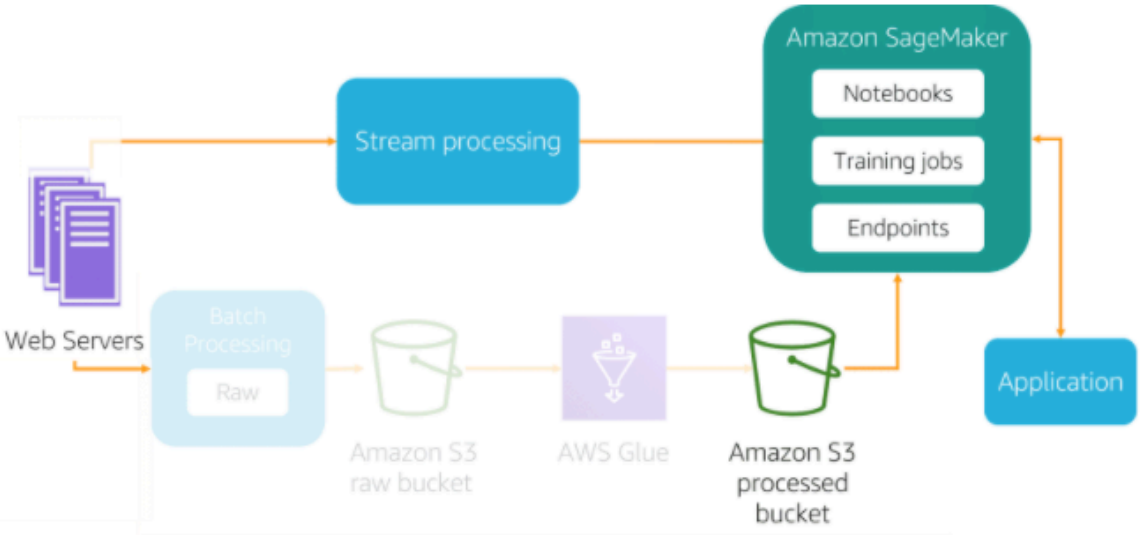
For batch ingestions to the AWS Cloud, you can use services like AWS Glue, an ETL (extract, transform, and load) service that you can use to categorize your data, clean it, enrich it, and move it between various data stores. AWS Database Migration Service (AWS DMS) is another service to help with batch ingestions. This service reads from historical data from source systems, such as relational database management systems, data warehouses, and NoSQL databases, at any desired interval. You can also automate various ETL tasks that involve complex workflows by using AWS Step Functions.



Stream processing

Stream processing manipulates and loads data as it's recognized

Stream processing, which includes real-time processing, involves no grouping at all. Data is sourced, manipulated, and loaded as soon as it is created or recognized by the data ingestion layer. This kind of ingestion is less cost-effective, since it requires systems to constantly monitor sources and accept new information. But you might want to use it for real-time predictions using an Amazon SageMaker endpoint that you want to show your customers on your website or some real-time analytics that require continually refreshed data, like real-time dashboards.



Amazon Kinesis is a platform for streaming data on AWS

AWS recommends that you capture and ingest this fast-moving data using Amazon Kinesis, a platform for streaming data on AWS. Amazon Kinesis gives you the opportunity to build custom streaming data applications for specialized needs, and it offers several services focused on making it easier to load and analyze your streaming data.

Amazon Kinesis Data Streams

With Amazon Kinesis Data Streams, you can use the Kinesis Producer Library (KPL), an intermediary between your producer application code and the Kinesis Data Streams API data, to write to a Kinesis data stream. With the Kinesis Client Library (KCL), you can build your own application to preprocess the streaming data as it arrives and emit the data for generating incremental views and downstream analysis.

Amazon Kinesis Data Firehose

As data is ingested in real time, you can use Amazon Kinesis Data Firehose to easily batch and compress the data to generate incremental views. Kinesis Data Firehose also allows you to execute custom transformation logic using AWS Lambda before delivering the incremental view to Amazon S3.

Amazon Kinesis Data Analytics

Amazon Kinesis Data Analytics provides the easiest way to process and transform the data that is streaming through Kinesis Data Streams or Kinesis Data Firehose using SQL. This lets you gain actionable insights in near-real time from the incremental stream before storing it in Amazon S3.

Kinesis Data Streams: https://aws.amazon.com/kinesis/data-streams/
enable real-time analytics use cases such as real-time dashboards, real-time anomaly detection, dynamic pricing,

Benefits

Real-time performance

Make your streaming data available to multiple real-time analytics applications, to Amazon S3, or to AWS Lambda within 70 milliseconds of the data being collected.

Durable

Reduce the probability of data loss. Synchronous replication of your streaming data across three Availability Zones in an AWS Region, and the storage of that data for up to seven days, provide multiple layers of protection from data loss.

Secure

Meet your regulatory and compliance needs by encrypting sensitive data within KDS, and privately accessing your data via your Amazon Virtual Private Cloud (VPC). Data can be secured at-rest by using server-side encryption and AWS KMS master keys.

Easy to use

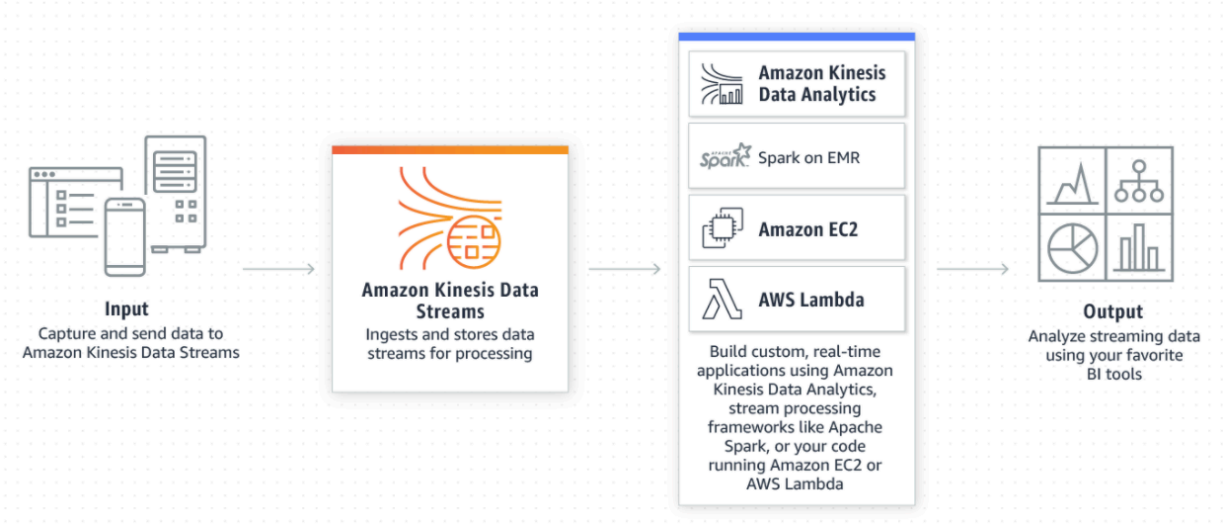
Build your streaming applications quickly using the AWS SDK, the Kinesis Client Library (KCL), connectors, and agents. Easily process data with built-in integrations to AWS Lambda, Amazon Kinesis Data Analytics, and Amazon Kinesis Data Firehose.

Elastic

Dynamically scale your applications. Kinesis data streams scale from megabytes to terabytes per hour, and scale from thousands to millions of PUT records per second. You can dynamically adjust the throughput of your stream at any time based on the volume of your input data.

Low cost

Kinesis Data Streams has no upfront cost, and you only pay for the resources you use. For as little as $0.015 per hour, you can have a Kinesis data stream with 1MB/second ingest and 2MB/second egress capacity.



Kinesis Data analytics: https://aws.amazon.com/kinesis/data-analytics/

Amazon Kinesis Data Analytics is the easiest way to transform and analyze streaming data in real time with Apache Flink. Apache Flink is an open source framework and engine for processing data streams. Amazon Kinesis Data Analytics reduces the complexity of building, managing, and integrating Apache Flink applications with other AWS services.

Amazon Kinesis Data Analytics takes care of everything required to run streaming applications continuously, and scales automatically to match the volume and throughput of your incoming data. With Amazon Kinesis Data Analytics, there are no servers to manage, no minimum fee or setup cost, and you only pay for the resources your streaming applications consume.

Benefits

Powerful real-time processing

Amazon Kinesis Data Analytics provides built-in functions to filter, aggregate, and transform streaming data for advanced analytics. It processes streaming data with sub-second latencies, enabling you to analyze and respond to incoming data and events in real time.

No servers to manage

Amazon Kinesis Data Analytics is serverless; there are no servers to manage. It runs your streaming applications without requiring you to provision or manage any infrastructure. Amazon Kinesis Data Analytics automatically scales the infrastructure up and down as required to process incoming data.

Pay only for what you use

With Amazon Kinesis Data Analytics, you only pay for the processing resources that your streaming applications use. There are no minimum fees or upfront commitments.

Build sophisticated streaming applications with Apache Flink

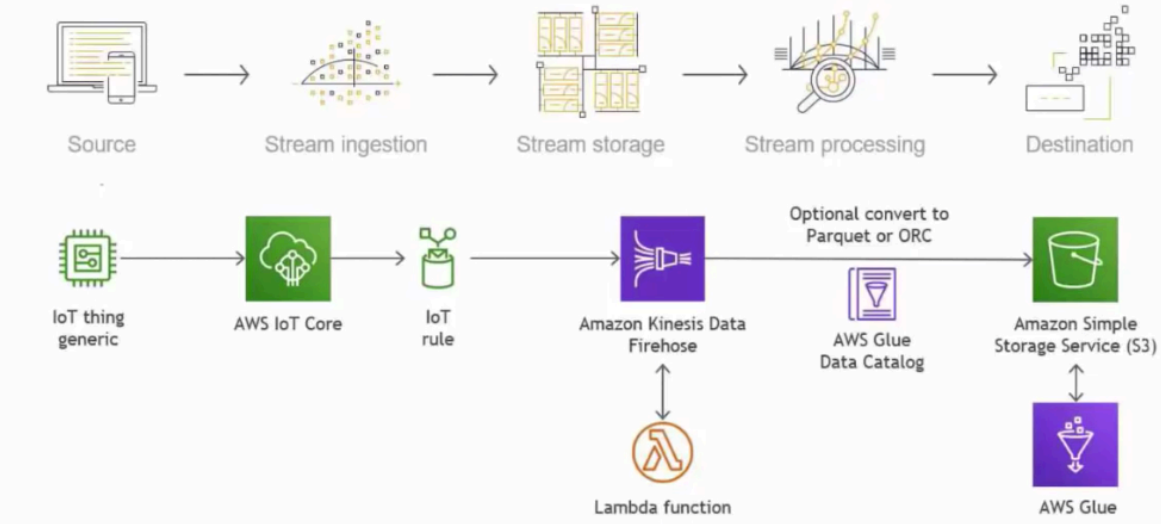
Amazon Kinesis Data Analytics includes open source libraries and runtimes based on Apache Flink that enable you to build an application in hours instead of months using your favorite IDE. The extensible libraries include specialized APIs for different use cases, including stateful stream processing, streaming ETL, and real-time analytics. You can use the libraries to integrate with AWS services like Amazon Managed Streaming for Apache Kafka (Amazon MSK), Amazon Kinesis Data Streams, Amazon Kinesis Data Firehose, Amazon Elasticsearch Service, Amazon S3, Amazon DynamoDB, and more.

Use standard SQL for interactive queries

Amazon Kinesis Data Analytics provides templates and an interactive editor that enable you to build SQL queries that perform joins, aggregations over time windows, filters, and more. You simply select the template appropriate for your analytics task, and then edit the provided code using the SQL editor to customize it for your specific use case. Without writing a single line of code, you can send your SQL results to other AWS services like AWS Lambda, Amazon Kinesis Data Streams, and Amazon Kinesis Data Firehose.

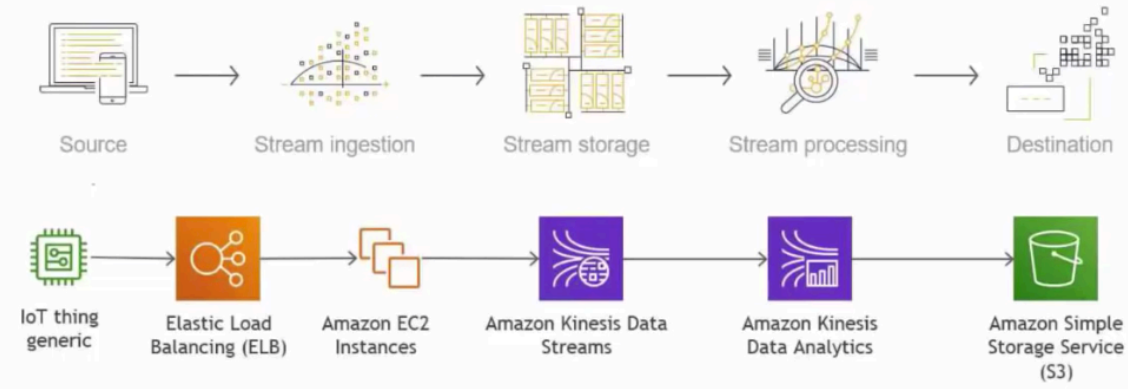
https://www.youtube.com/watch?v=BkinvmBRFHY

Simple ETL to your data lake



JOHN DEERE aws

Sophisticated streaming ETL to your data lake



Topics related to this subdomain

Here are some topics you may want to study for more in-depth information related to this subdomain:

- [ ] Amazon Kinesis Data Streams
- [ ] Amazon Kinesis Data Firehose
- [ ] Amazon Kinesis Data Analytics
- [ ] Amazon Kinesis Video Streams
- [ ] AWS Glue
- [ ] Apache Kafka

Domain 1.3: Identify and implement a data transformation solution

Raw ingested data is not ML ready

The raw data ingested into a service like Amazon S3 is usually not ML ready as is. The data needs to be transformed and cleaned, which includes deduplication, incomplete data management, and attribute standardization. Data transformation can also involve changing the data structures, if necessary, usually into an OLAP model to facilitate easy querying of data.

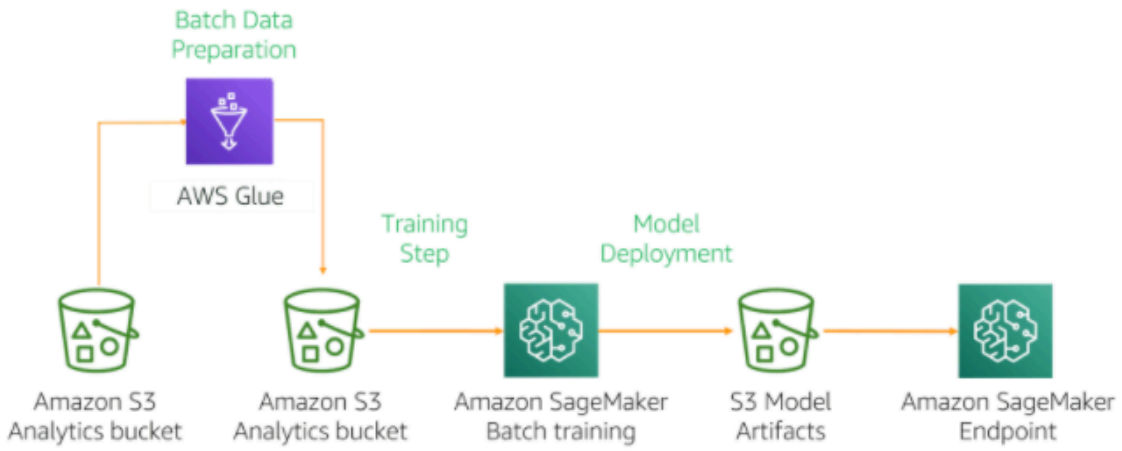
Doing this in the context of ML, while using key services that help you with data transformation, is the focus of this subdomain.

Using Apache Spark on Amazon EMR provides a managed framework

Using Apache Spark on Amazon EMR provides a managed framework that can process massive quantities of data. Amazon EMR supports many instance types that have proportionally high CPU with increased network performance, which is well suited for HPC (high-performance computing) applications.

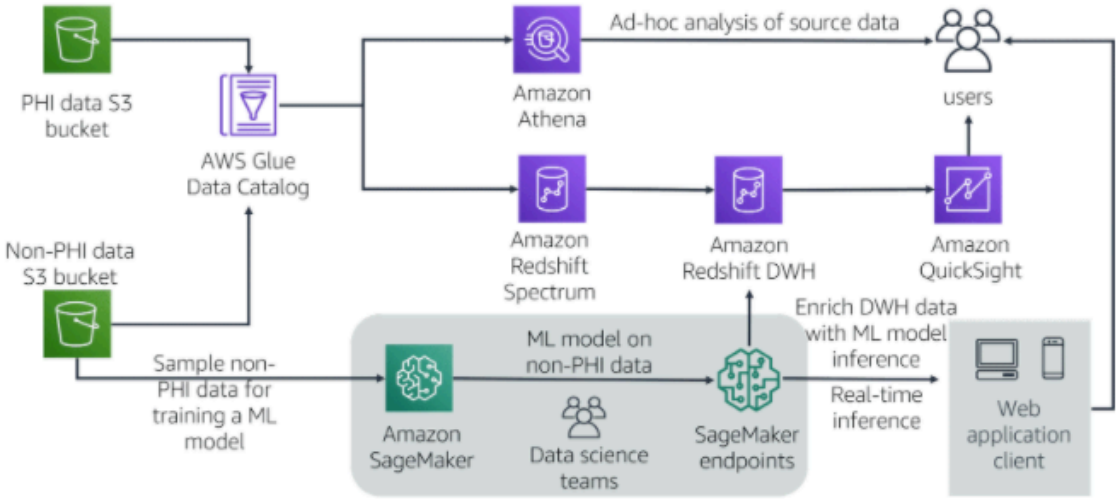
A key step in data transformation for ML is partitioning your dataset

Datasets required for ML applications are often pulled from database warehouses, streaming IoT input, or centralized data lakes. In many use cases, you can use Amazon S3 as a target endpoint for their training datasets. ETL processing services (Amazon Athena, AWS Glue, Amazon Redshift Spectrum) are functionally complementary and can be built to preprocess datasets stored in or targeted to Amazon S3. In addition to transforming data with services like Athena and Amazon Redshift Spectrum, you can use services like AWS Glue to provide metadata discovery and management features. The choice of ETL processing tool is also largely dictated by the type of data you have. For example, tabular data processing with Athena lets you manipulate your data files in Amazon S3 using SQL. If your datasets or computations are not optimally compatible with SQL, you can use AWS Glue to seamlessly run Spark jobs (Scala and Python support) on data stored in your Amazon S3 buckets.



You can store a single source of data in Amazon S3 and perform ad hoc analysis

This reference architecture shows how AWS services for big data and ML can help build a scalable analytical layer for healthcare data. Customers can store a single source of data in Amazon S3 and perform ad hoc analysis with Athena, integrate with a data warehouse on Amazon Redshift, build a visual dashboard for metrics using Amazon QuickSight, and build an ML model to predict readmissions using Amazon SageMaker. By not moving the data around and connecting to it using different services, customers avoid building redundant copies of the same data.



Topics related to this subdomain

Here are some topics you may want to study for more in-depth information related to this subdomain:

- [ ] Apache Spark on Amazon EMR
- [ ] Apache Spark and Amazon SageMaker
- [ ] AWS Glue

AWS Glue and scala ETL: https://docs.aws.amazon.com/glue/latest/dg/aws-glue-programming-scala.html

Apache Spark and Pagemaker: https://docs.aws.amazon.com/sagemaker/latest/dg/apache-spark-example1.html
Amazon SageMaker provides an Apache Spark library (in both Python and Scala) that you can use to integrate your Apache Spark applications with SageMaker. For example, you might use Apache Spark for data preprocessing and SageMaker for model training and hosting

Walk-through of sample questions

In this section, you'll have a chance to answer and walk through the solution to two different sample questions. The questions reflect some of the design and technical content you may see on the exam. The videos below will give you the answers to each question by walking you through the test-taking strategies presented to you earlier in this course.

Question 1

Answer the question below before watching the corresponding solution video.

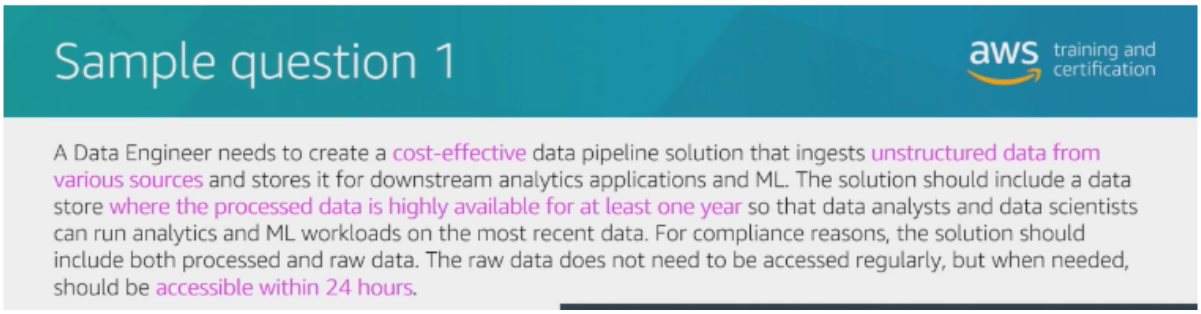
A data engineer needs to create a cost-effective data pipeline solution that ingests unstructured data from various sources and stores it for downstream analytics applications and ML. The solution should include a data store where the processed data is highly available for at least one year, so that data analysts and data scientists can run analytics and ML workloads on the most recent data. For compliance reasons, the solution should include both processed and raw data. The raw data does not need to be accessed regularly, but when needed, should be accessible within 24 hours.

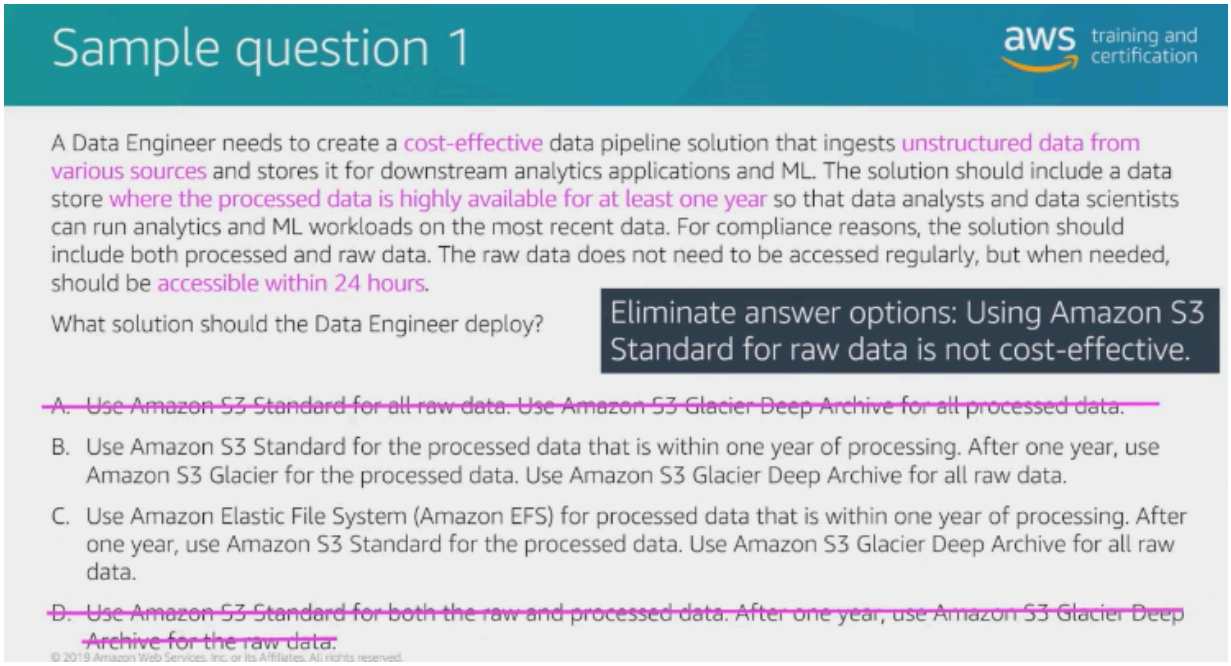
What solution should the data engineer deploy?

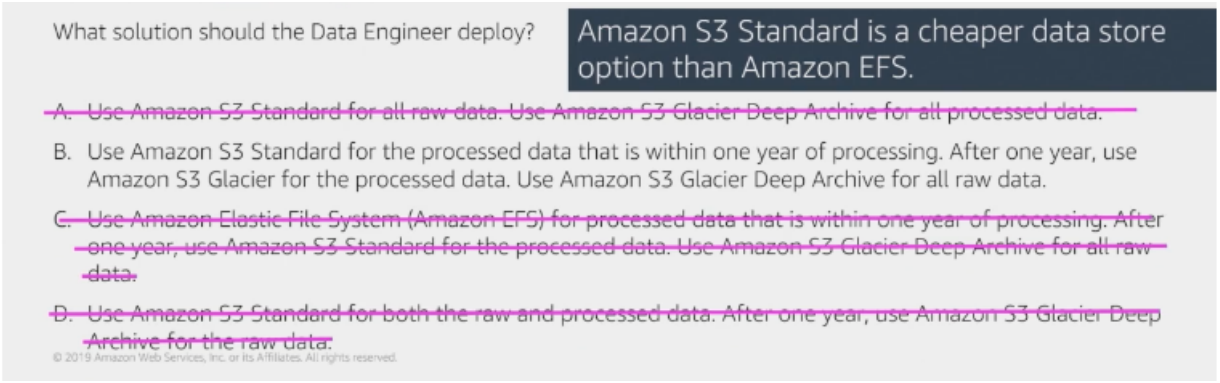
- Use Amazon S3 Standard for all raw data. Use Amazon S3 Glacier Deep Archive for all processed data.
- Use Amazon S3 Standard for the processed data that is within one year of processing. After one year, use Amazon S3 Glacier for the processed data. Use Amazon S3 Glacier Deep Archive for all raw data.
- (•) Use Amazon Elastic File System (Amazon EFS) for processed data that is within one year of processing. After one year, use Amazon S3 Standard for the processed data. Use Amazon S3 Glacier Deep Archive for all raw data.
- Use Amazon S3 Standard for both the raw and processed data. After one year, use Amazon S3 Glacier Deep Archive for the raw data.

=> Incorrect

Solution to question 1







S3 is cheaper than EFS

Question 2

Answer the question below before watching the corresponding solution video.

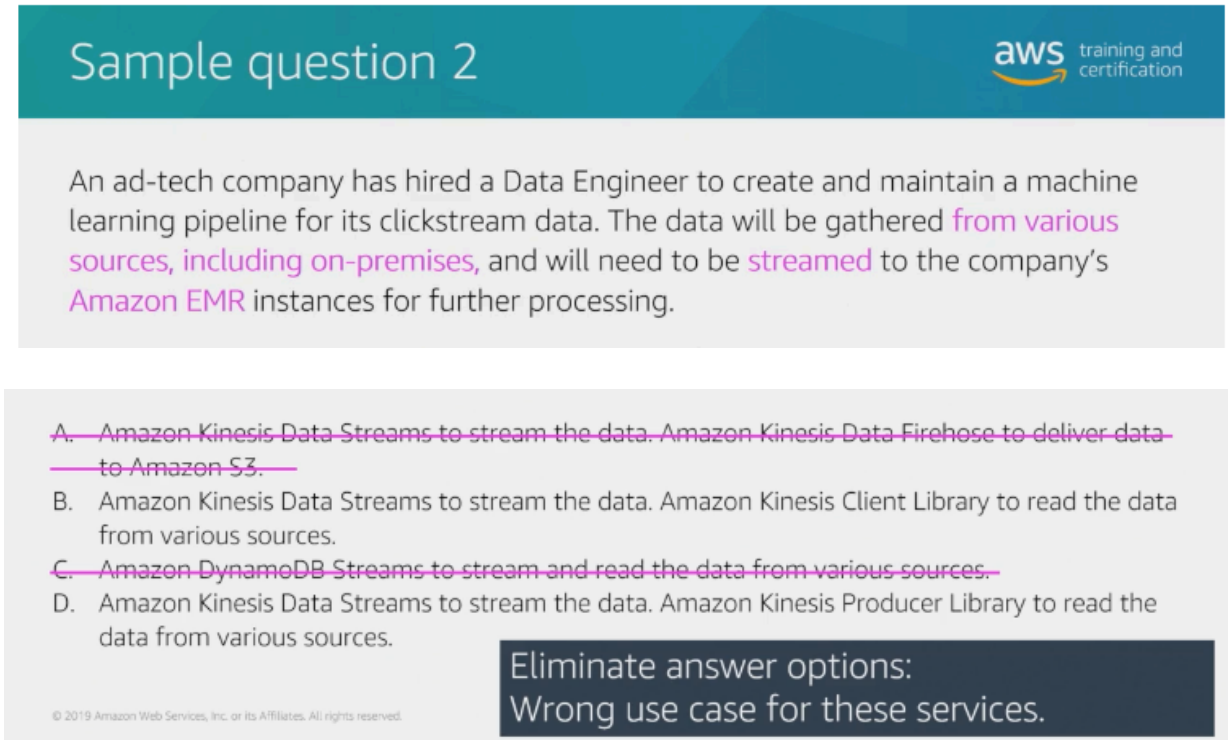
An ad-tech company has hired a data engineer to create and maintain a machine learning pipeline for its clickstream data. The data will be gathered from various sources, including on premises, and will need to be streamed to the company's Amazon EMR instances for further processing.

What service or combination of services can the company use to meet these requirements?

- ○ Amazon Kinesis Data Streams to stream the data. Amazon Kinesis Data Firehose to deliver data to Amazon S3.
- ○ Amazon Kinesis Data Streams to stream the data. Amazon Kinesis Client Library to read the data from various sources.
- ○ Amazon DynamoDB Streams to stream and read the data from various sources.
- ● Amazon Kinesis Data Streams to stream the data. Amazon Kinesis Producer Library to read the data from various sources.

=> Incorrect

Solution to question 2



KPL is used to produce NOT read the data

Domain quiz

Take this short quiz to test your understanding of some of the topics related to this domain and experience the types of questions that will be on the exam. We recommend you use the test-taking strategies presented earlier in this course when completing the questions. Click the link below to begin.

https://amazonmr.au1.qualtrics.com/jfe/form/SV_dg3duwjYBh3zZnn

aws training and certification

A healthcare company using the AWS Cloud has access to a variety of data types, including raw and preprocessed data. The company wants to start using this data for its ML pipeline, but also wants to make sure the data is highly available and located in a centralized repository.

What approach should the company take to achieve the desired outcome?

- (●) Create a data lake using Amazon S3 as the data storage layer
- ( ) Store unstructured data in Amazon DynamoDB and structured data in Amazon RDS
- ( ) Use Amazon FSx to host the data for training
- ( ) Use Amazon Elastic Block Store (Amazon EBS) volumes to store the data with data backup

A Data Scientist wants to implement a near-real-time anomaly detection solution for routine machine maintenance. The data is currently streamed from connected devices by AWS IoT to an Amazon S3 bucket and then sent downstream for further processing in a real-time dashboard.

What service can the Data Scientist use to achieve the desired outcome with minimal change to the pipeline?

- (●) Amazon Kinesis Data Analytics
- ( ) Amazon SageMaker
- ( ) Amazon EMR with Spark
- ( ) Amazon CloudWatch

A transportation company currently uses Amazon EMR with Apache Spark for some of its data transformation workloads. It transforms columns of geographical data (like latitudes and longitudes) and adds columns to segment the data into different clusters per city to attain additional features for the k-nearest neighbors algorithm being used.

The company wants less operational overhead for their transformation pipeline. They want a new solution that does not make significant changes to the current pipeline and only requires minimal management.

What AWS services should the company use to build this new pipeline?

- ○ Use AWS Glue to transform files. Use Amazon EMR HDFS as the destination.
- ◉ Use AWS Glue to transform files. Use Amazon S3 as the destination.
- ○ Use Lambda to transform files. Use Amazon EMR HDFS as the destination.
- ○ Use Amazon EMR to transform files. Use Amazon S3 as the destination.

3/3 **100.0%**