# **Fundamentals of Analytics**

# **1. Analytics Concepts**

## **1.1. Analytics**

**Analytics:**The process of using specialized tools and techniques to find new value from raw data

**Data analysis:** The practice of interpreting data that leads to meaningful decisions.

### **Analytics benefits**

Analytics is vital to every organization. Analytics helps companies decide where and when to launch new products, when to offer discounts, and when to market in new areas.

### **Analytics types and techniques**

1. **Descriptive analytics** –Helps to answer what happened.

It uses data visualization techniques, such as the following:

* **Pie charts:** A diagram that shows data as slices in a circular-shaped graph.
* **Bar charts:** A diagram that shows data in rectangular bars horizontally or vertically.
* **Line graphs:** A diagram that uses lines to connect single data points generally plotted over a period of time.
* **Tables:** A diagram that shows data in rows and columns
* **Generated narratives:** You can ask questions about your data and receive answers with visualizations.

1. **Diagnostic analytics** –Helps to answer why it happened. It uses below techniques:

* **Drill-down:** Seeing an overview of the data, to a detailed view within the same dataset.
* **Data discovery:** A process for gathering, cataloguing, and classifying data from different databases for analytics.
* **Data mining:** Using analytics against a large dataset to discover meaningful insights.
* **Correlations:** A measure between two variables that shows how closely related they are without stating a cause-and-effect relationship

1. **Predictive analytics** – Helps to answer what might happen in the future. It uses below techniques:

* **Machine learning (ML):** A technique that teaches software how to learn from data, find patterns, and make decisions with minimal human intervention.
* **Forecasting:** Predicting future value by looking at unique trends
* **Pattern matching:** Finding pre-determined patterns in raw data.
* **Predictive modelling:** Predicting future events by analysing patterns with input data.

1. **Prescriptive analytics** –Prescriptive analytics recommends actions to the predicted outcome.

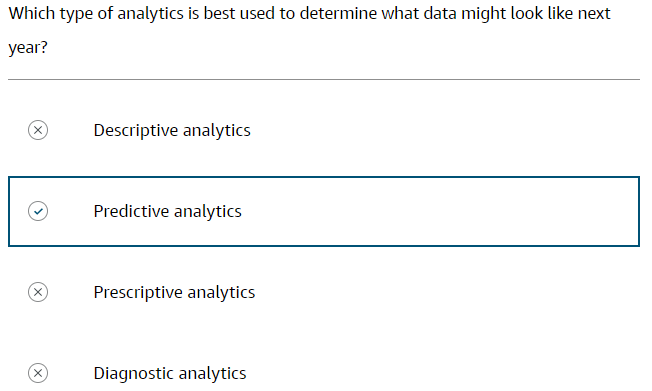
It uses techniques, such as the following:

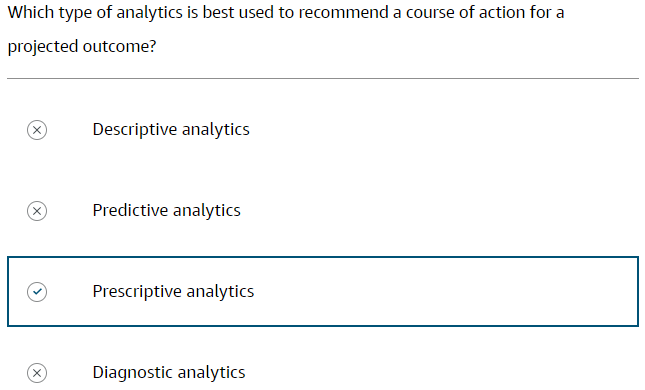
* **Graph analysis:** Analysing relationships between objects in a network or graph
* **Simulation:** Modelling the behaviour of a system by using the characteristics and relationships of system components
* **Complex event processing:** Deriving a conclusion through tracking and analyzing streaming data.
* **Neural networks:** An ML model that copies the function of neurons in the human brain to learn and solve complex problems.
* **Recommendation engines:** A system that uses ML recommends the most relevant items to a user, based on behaviour patterns.

### **Reference:**

* [What is Data Analytics?](https://aws.amazon.com/what-is/data-analytics/)

### **Knowledge Check:**





## **1.2. Machine Learning**

**Artificial intelligence (AI)**: A broad branch of computer science that is involved with building smart machines that can perform tasks requiring human intelligence.

**Machine learning (ML) model**: A computer program that is designed to find patterns from an unanalysed dataset.

**ML algorithm**: A computer program that helps computers understand hidden patterns in data, make predictions about the data, and recommend actions to take.

### **ML**

ML is a subset of AI. Computers use ML to learn from and make predictions based on data. ML models can forecast what might happen in the future (predictive analytics) and provide a course of action (prescriptive analytics). These models become more accurate through a process called training. During training, applications run data through rules and constraints several times. This refines the model’s ability to make accurate recommendations.

### **Analytics for ML**

ML algorithms can analyze huge volumes of data much faster than humans. Algorithms can also be built to identify trends, correlations, and anomalies in datasets. ML automates the process of extracting insights and patterns from data, saving time and effort.

As the volume of data continues to grow, ML helps organizations to process vast amounts quickly, all while discovering meaningful patterns and using insights for better decision-making. ML is essential for analytics to uncover valuable knowledge and drive innovation across organizations.

### **ML on AWS**

AWS AI and ML services make it convenient for developers to add intelligence to applications without needing ML expertise. You can use AWS pre-trained AI services to automate data extraction and analysis, personalize the customer experience, detect fraudulent online activity, and more.

There are three different levels of ML services in AWS:

1. **AWS AI services** –AI services provides developers ready-made AI intelligence to integrate into their applications and workflows. AI services use the same deep learning technology that powers Amazon.com and Amazon ML services, so you get quality and accuracy from continuously learning APIs. AI services on AWS don't require ML skills.
2. **ML services** –ML services makes it convenient for any developer to accelerate their ML innovation with purpose-built ML tools, optimized for ML applications.
3. **ML frameworks and infrastructure** –ML practitioners can design their own tools and workflows to build, train, tune, and deploy models.

### **Generative AI on AWS**

Generative AI is a type of ML model that creates new content and ideas from user prompts. Generative AI is not just outputting data, but generating content like conversations, stories, images, videos, and music.

* **What it does** – Generative AI learns from existing data and then uses that knowledge to create new, original content. It is done through a type of ML called deep learning, which uses algorithms to mimic the structure of the human brain. These algorithms, called artificial neural networks, and are able to learn complex patterns from data.

Generative AI is powered by large, pre-trained models called foundation models. Generative AI models are trained on massive amounts of data, which can be anything from text to images to music. The model learns the patterns and structures of the data. It then uses that data to make predictions about what the content should look like, similar to the data it was trained on.

* **Generative AI examples** –You can use generative AI models for the following:
  + Use a generative AI model trained on a corpus of text to generate new text, such as news articles, blog posts, or poems.
  + Use a generative AI model trained on a dataset of images to generate new images, such as paintings, photographs, or sketches.
  + Use a generative AI model trained on a dataset of music to generate new music, such as songs, symphonies, or concertos.

### **Amazon CodeWhisperer**

Amazon CodeWhisperer is a code generation service that analyzes your code and comments as you write code in your integrated development environment (IDE). It goes beyond code completion by leveraging natural language processing to write code by understanding the comments in the code.

##### **WHAT IT DOES**

By understanding English comments, Amazon CodeWhisperer generates complete functions and code blocks that align with your descriptions. Additionally, Amazon CodeWhisperer analyzes the surrounding code, ensuring the generated code matches your style, naming conventions, and seamlessly integrates into the existing context.

Amazon CodeWhisperer includes a security scanning feature that detects security vulnerabilities in both generated and developer-written code. It scans the code to identify potential vulnerabilities and provides suggestions for remediation. This includes scanning for hard-to-find vulnerabilities that may be overlooked. The security scan is compatible with popular IDEs such as Visual Studio Code (VS Code) and JetBrains. It supports over 15 programming languages.

Supported programming languages - [Language Support in Amazon CodeWhisperer](https://docs.aws.amazon.com/codewhisperer/latest/userguide/language-ide-support.html).

Amazon CodeWhisperer has the potential to increase development speed, security, and the quality of software.

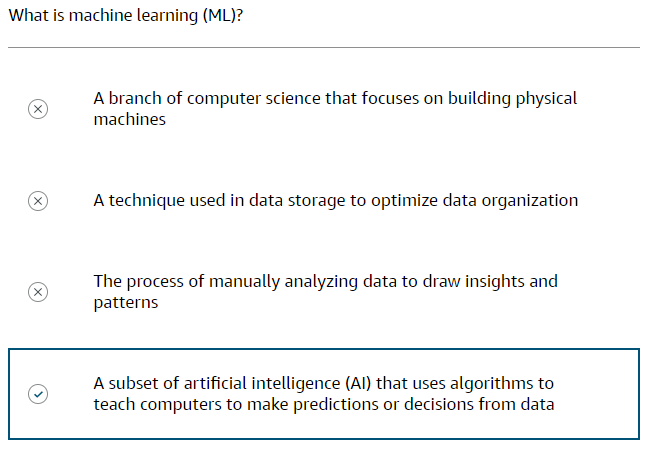
##### **FEATURES**

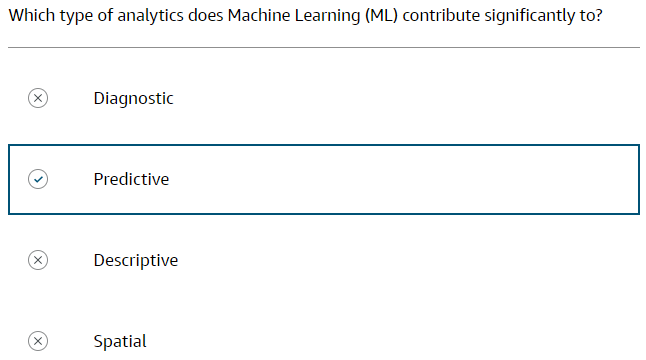
* Uses natural language processing of English comments in your code.
* Offers real-time code completion and code generation suggestions.
* Suggests whole lines of code, complete functions, and logical blocks of code.
* Reference tracker protects open-source intellectual property.
* Keeps codebase up to date with the latest security practices.

##### **BENEFITS**

* Accelerates application development for faster delivery of software solutions.
* Automates repetitive tasks to optimizes uses of developer time so they can focus on more critical aspects of the project.
* Mitigates security vulnerabilities.
* Enhances code quality and reliability to produce robust and efficient applications.
* Provides efficient responses to evolving software threats.

### **Knowledge check**



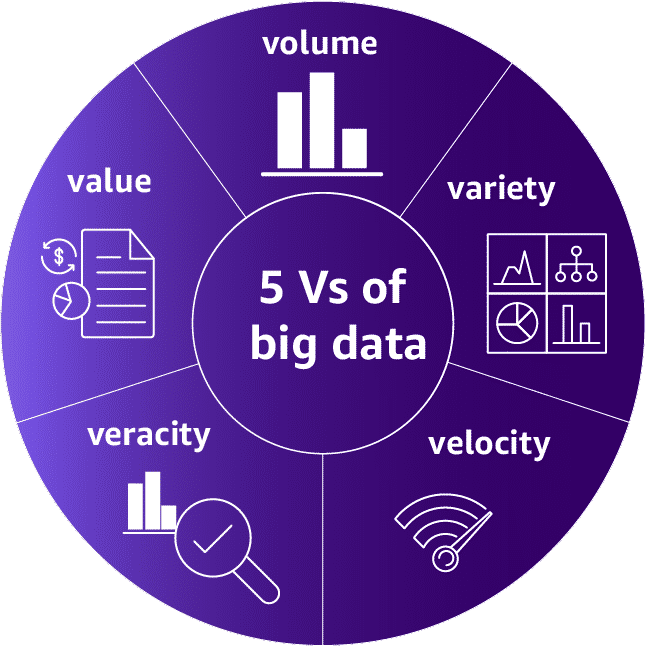


## **1.3. 5 Vs of Big Data**

**Big data:** Data that is stored rapidly from various sources, massive in size, and is complicated for organizations to secure, analyze, and gain valuable insight from.

### **Big data challenges**

Organizations encounter difficulties with the large amounts, rapid velocity, and diverse kinds of data being created and gathered. This burdens data storage systems and impacts the usefulness and insight that can be gained. Big data challenges cannot be solved with traditional database and processing solutions. You need a good understanding of the big data characteristics: volume, variety, velocity, veracity, and value. You can then begin to find the right AWS data and analytics solutions for the 5 V challenges of big data.



* 1. **Volume –** Volume means the amount of data that will be ingested by the solution-the total size of the data coming in.
  2. **Variety –** Data can come from many different sources. Variety means the number of different sources and the types of sources that the solution will use.
  3. **Velocity –** Velocity means the speed of data entering and flowing through to be processed. Many organizations now require near real-time ingestion and processing of data. The high velocity of data results in a shorter time to analyze than traditional data processing can provide.
  4. **Veracity –** Veracity is the degree to which data is accurate, precise, and trusted. It is contingent on the integrity and trustworthiness of the data.
  5. **Value –** Value is the ability to extract meaningful information from the data that has been stored and analysed.

Not all organizations experience challenges in every area. Some organizations struggle with ingesting large volumes of data rapidly. Others struggle with processing massive volumes of data to produce new predictive insights. Still, others have users that need to perform detailed data analysis on the fly over enormous data sets.

## **1.4. Volume**

### **Data volume**

The size of data stored by an organization can be from terabytes to petabytes to sometimes even exabytes. This makes it difficult for traditional storage systems to manage efficiently. Every organization's infrastructure has to support large amounts of scalable and durable data storage and must also be able to gather this data from many different sources. Every company with data needs vast amount of computing and storage capacity.

**Global data creation is projected to grow to more than 180 zettabytes by 2025.**

More information - [Volume of Data/Information Created, Captured, Copied and Consumed Worldwide from 2010 to 2020, with Forecasts from 2021 to 2025](https://www.statista.com/statistics/871513/worldwide-data-created/).

Here are some example data sources for ingestion and storage.

1. **Transactional data** – Transactional data includes the following:

* Customer information
* Online product purchases
* Service contracts

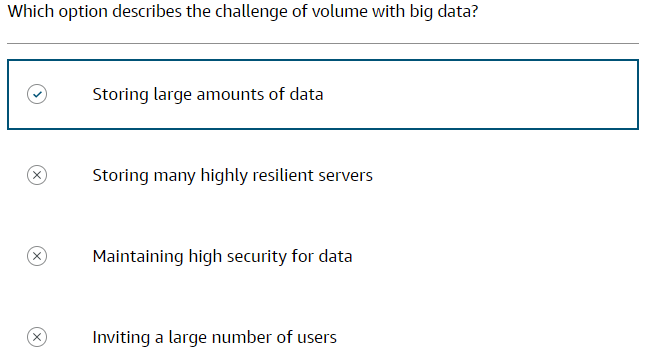
1. **Temporary data** – Temporary data includes the following:

* Moves you make in an online video game.
* Internet browser cache

1. **Objects** – Objects include the following:

* Images
* Email messages
* Text files
* Social media content
* Text messages
* Videos

### **Knowledge check**



## **1.5. Variety**

### **Data source types**

Analytics begins with data sources. A data source can be a folder on a file server, database, web page, and even a wearable device. Structured data, semi structured data, and unstructured data each have unique characteristics that form a recognizable pattern.

* 1. **Structured data**
* Customer Relationship Management (CRM) system
* Online forms
* Network logs
* Event reservation system
  1. **Semi-structured data**
* CSV
* JSON
* XML
  1. **Unstructured data**
* Clickstream data
* Emails
* Documents, PDFs
* Photos, Videos

The following is an example of semi-structured data: a chat conversation.

[

{

"user": "Jie",

"message": "Hey Pat, how's it going?",

"timestamp": "2023-12-18T12:00:00Z"

},

{

"user": "Pat",

"message": "Hi Jie, I'm good thanks! Just working on a project. How about you?",

"timestamp": "2023-12-18T12:01:30Z"

},

{

"user": "Jie",

"message": "Good to hear! I'm doing well, just catching up on some emails. Got any fun plans for the holidays coming up?",

"timestamp": "2023-12-18T12:03:00Z"

},

{

"user": "Pat",

"message": "Not too much planned yet! Going to see my family so that should be nice. What about you?",

"timestamp": "2023-12-18T12:05:15Z"

},

{

"user": "Jie",

"message": "Yeah, flying home next week to spend time with my parents and catch up with some old friends. Should be a nice break!",

"timestamp": "2023-12-18T12:07:45Z"

}

]

### **Data storage methods**

You can classify data source types based on the storage types that are used. Structured and semi-structured data are stored in database management systems (DBMS). Unstructured data is stored in data lakes or object storage solutions.

#### **Structured data stores**

Structured data is stored within a relational database management system (RDBMS). A database is a structured set of data stored in a computer. A RDBMS provides structure to the data, lifecycle maintenance, and management of interactions with other processes and systems.

##### **Structured data storage methods**

Relational databases store data in tables that are related to one another. The goal for relational databases is optimized storage. For querying the data, relational databases use the structured query language (SQL). There are many types of databases to choose from.

**Strengths -** Strengths of a relational database include the following:

* Proven technology widely adopted and understood.
* Less risk involved.
* Very good transactional latency
* Outstanding for relatively small data sets

**Weaknesses -** Weaknesses of a relational database include the following:

* Scalability concerns
* Need to increase hardware capacities as data grows.
* Fixed schema makes it difficult to make non-disruptive changes.
* Struggles with storing unstructured data.

#### **Semi-structured data stores**

Semi-structured data are often stored in non-relational database systems, sometimes called NoSQL databases. This term can be confusing. Non-relational or NoSQL does not mean the data stored cannot be queried using SQL. A better way to think of it is not only SQL.

##### **NoSQL database storage methods**

Non-relational databases are built to store semi-structured data for rapid collection and retrieval. NoSQL databases store data as a collection of documents or key-value pairs.  NoSQL databases are denormalized, meaning they can have duplicate data, and have the option of using internal document structure or no structure at all.

###### **KEY VALUE**

Key-value databases are a type of non-relational database that store unstructured data in the form of key-value pairs.

Logically, data is stored in a single table. Within the table, the values are associated with a specific key. The values are stored in the form of blob objects and do not require a predefined schema. The values can be of nearly any type.

**Strengths include the following:**

* Very flexible
* Able to handle a wide variety of data types.
* Keys are linked directly to their values with no need for indexing or complex join operations.
* Content of a key can be conveniently copied to other systems without reprogramming the data.

**Weaknesses include the following:**

* Difficult to query values because they are stored as a single blob.
* Updating or editing the content of a value is quite difficult.
* Not all objects are conveniently modelled as key-value pairs.

###### **DOCUMENT STORES**

Document stores are a type of non-relational database that store semi-structured and unstructured data in the form of files. These files range in form, but include JSON, BSON, and XML. The files can be navigated using numerous languages, including Python and Node.js.

Logically, files contain data stored as a series of elements. Each element is an instance of a person, place, thing, or event. For instance, the document store might hold a series of log files from a set of servers. These log files can each contain the specifics for that system without concern for what the log files in other systems contain.

**Strengths include the following:**

* Flexibility
* No need to plan for a specific type of data when creating one.
* Convenient to scale

**Weaknesses:**

* Cannot query across files.

NoSQL databases have multiple languages that can be used to query their data, and they focus on object querying. NoSQL databases place the emphasis on compute power. In terms of scalability, NoSQL databases are good at being distributed across multiple servers or instances. NoSQL databases are commonly used for OLTP web and mobile applications.

**Strengths -** Strengths of NoSQL databases include the following:

* Ability to update schemas on the fly.
* Faster development cycles
* Less downtime
* Scales really well for larger data sets

**Weaknesses -** Weaknesses of NoSQL databases include the following:

* Data is not instantaneously updated with every change.
* Does not perform well for applications requiring extremely low transactional latency.
* It is not as a mature as relational database technology.

### **OLTP and OLAP systems**

Within databases, there are two primary methods for organizing information: online transaction processing, or OLTP, and online analytical processing, or OLAP. Two different systems are needed, based on how the resources supporting the database are being used. Smaller databases have a tolerance for simultaneous write and read operations. However, in large databases, you are forced to sacrifice the performance of read operations to allow high-performance write operations—or sacrifice performance of write operations to allow for high-performance read operations. The solution is to have an OLTP database optimized for write operations and an OLAP database optimized for read operations.

#### **OLTP SYSTEM**

Transactional databases are called OLTP databases. In an OLTP system, the most common queries are called lookup queries. OLTP is associated more with queries that return entire rows. The filters on this data are generally based on the key columns in that table. In this type of system, you might query to get details for a specific order.

#### **OLAP SYSTEM**

In an OLAP system, the most common queries are aggregate queries. These queries take large numbers of rows and reduce them to a single row by aggregating the values in one or more columns. OLAP is associated more with queries that only need a few columns. In this type of system, you might query to find out the total number of items sold on a specific date.

### **Row-based and columnar data storage**

Data within a database should be indexed to allow a query to quickly find the data it needs to produce a result. Indexes control the way data is physically stored on disk. They physically group records into a predictable order based on the key values within the table. This plays a huge part in the speed and efficiency of queries. Both OLTP and OLAP systems can use either indexing method.

The following table breaks down some of the characteristics of each storage type.

| **Characteristic** | **Row-based** | **Columnar** |
| --- | --- | --- |
| **Storage on disk** | Row by row | Column by column |
| **Read/write** | Best at random reads and writes | Best at sequential reads and writes |
| **Best for** | Returning full rows of data based on a key | Returning aggregations of column values |
| **Implementation** | Transactional systems | Analytical processing |
| **Data compression** | Low to medium compression can be achieved | High compression is the norm |

In practice, data is written to the OLTP database with a very high frequency. Records from that system are copied over to an OLAP system on a scheduled basis. In many solutions, the data being copied to the OLAP database is also transformed to provide clearer answers for analytic questions or pre-aggregated to enable rapid query results. This is all done with extract, transform, and load (ETL) operations.

### **Comparing data storage methods**

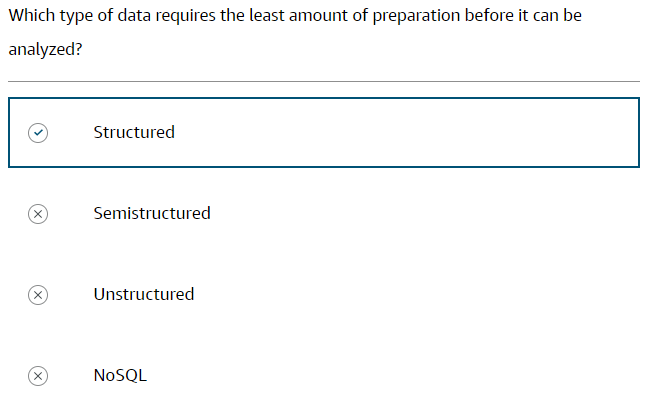
Following are the key characteristics of the two database types.

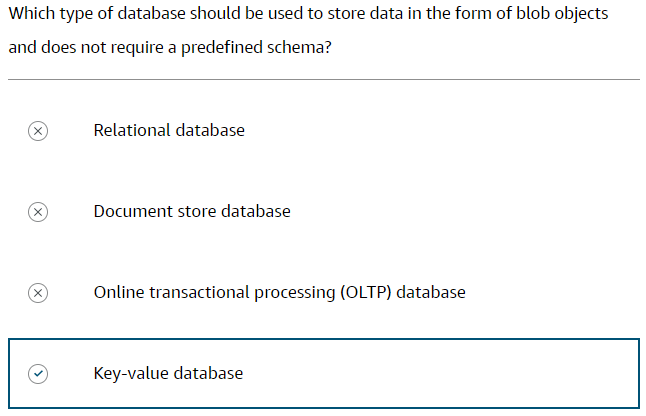
| **Characteristic** | **Row-based** | **Columnar** |
| --- | --- | --- |
| **Representation** | Multiple tables, each containing columns and rows | Collection of documents Single table with keys and values |
| **Data design** | Normalized relational or dimensional data warehouse. | Denormalized document, wide column, or key value |
| **Optimized** | Optimized for storage | Optimized for compute |
| **Query style** | Language: SQL | Language: Many Uses object querying |
| **Scaling** | Scale vertically | Scale horizontally |
| **Implementation** | OLTP business systems | Analytical systems |

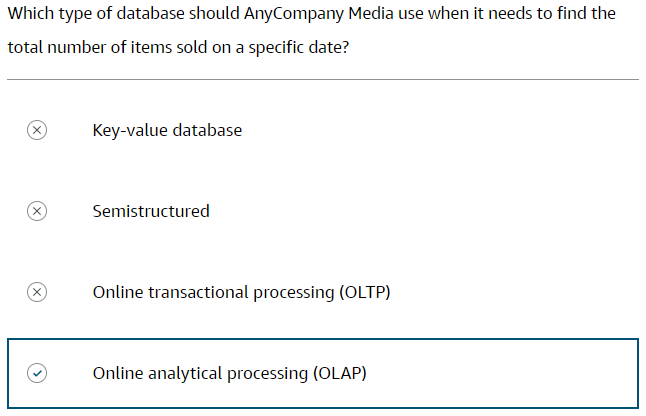
### **Purpose-built data stores**

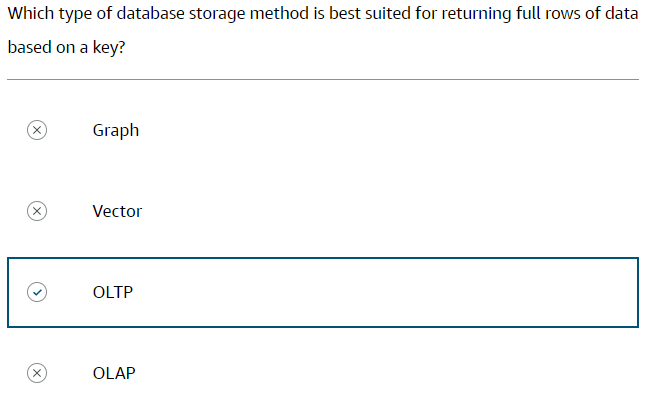
| **Databases** | **Description** |
| --- | --- |
| **Amazon Aurora** | High performance, high availability, scalable, proprietary serverless relational database management system (RDBMS) with full MySQL and PostgreSQL compatibility |
| **Amazon Relational Database Service (Amazon RDS)** | Managed relational database service in the cloud with various database engine options |
| **Amazon Redshift** | Cloud-based data warehousing with ML to deliver the best price performance at any scale |
| **Amazon DynamoDB** | Fast, flexible, and highly scalable NoSQL database |
| **Amazon ElastiCache** | Fully managed, cost-optimized, highly scalable data caching service for real-time performance |
| **Amazon MemoryDB for Redis** | Redis-compatible, durable, in-memory database for ultra-fast performance |
| **Amazon DocumentDB (with MongoDB compatibility)** | Fully managed, scalable JSON document database |
| **Amazon Keyspaces (for Apache Cassandra)** | Scalable, high availability, serverless, managed Apache Cassandra compatible database service |
| **Amazon Neptune** | High availability, scalable, serverless graph database |
| **Amazon Timestream** | Fast, scalable, and serverless time-series database |
| **Amazon Quantum Ledger Database (QLDB)** | Fully managed, cryptographically verifiable ledger database |
| **AWS Database Migration Service (DMS)** | Automated managed migration and replication service to move database and analytics workloads to AWS with minimal downtime and zero data loss |

### **Knowledge check**









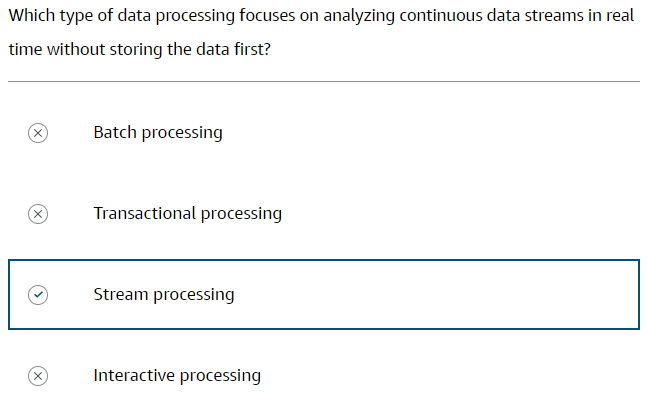
## **1.6. Velocity**

### **Data processing velocities**

There are four velocities for processing data.

* 1. **Scheduled** – Scheduled batch processing represents data that is processed in a very large volume on a regularly scheduled basis. For instance, once a week or once a day. It is generally the same amount of data with each load, making these workloads predictable.
  2. **Periodic** – Periodic batch processing is a batch of data that is processed at irregular times. These workloads are often run after a certain amount of data has been collected. This can make them unpredictable and hard to plan around.
  3. **Near real-time** – Near real-time processing represents streaming data that is processed in small individual batches. The batches are continuously collected and then processed within minutes of the data generation.
  4. **Real-time** – Real-time processing represents streaming data that is processed in very small individual batches. The batches are continuously collected and then processed within milliseconds of the data generation.

### **Knowledge check**



## **1.7. Veracity**

### **Data integrity and preventing potential issues.**

Data changes over time. As it is transferred from one process to another, and through one system to another, there are chances for the integrity of the data to be negatively impacted.

##### **THE IMPORTANCE OF DATA INTEGRITY**

You must ensure that you maintain a high level of certainty that the data you are analysing is trustworthy. Data veracity is contingent on the integrity of the data. Data integrity is all about making sure your data is trustworthy. Understanding the full lifecycle of your data and knowing how to protect it effectively will greatly strengthen the integrity of your data.

##### **IDENTIFYING DATA INTEGRITY ISSUES**

A data analyst might be called to perform data integrity checks. During this process they look for potential sources of data integrity problems. Data can come from both internal and external sources. It is highly unlikely that they will influence data generated outside of the organization. However, within the organization, they might have the ability to make recommendations on improvements for the data sources they will be interacting with.

### **Transforming data with the ETL process**

Extract, transform, and load (ETL) is the process of collecting data from raw data sources and transforming that data into a common type. This new data is loaded into a final location to be available for analysis and inspection. The ETL approach uses a set of business rules to process data from several sources before centralized integration.

**The purpose of ETL process:**

* To ensure the data has the required accuracy, precision, and depth.
* To bring together data from different sources to gain a complete picture.
* To build purpose-built data sets to answer key business questions.

**Extract data** –The extraction phase of this process is arguably the most important of all phases. The data required for most analytics transformations will likely come from multiple locations and be of multiple types, such as transaction logs, product databases, public data sources, or application streams.

There are four key areas you must plan for:

1. You must identify **where** all of the source data resides. This may be data stored on-premises by your company but can also include data that is found online.
2. You must carefully plan **when** the extraction will take place due to the potential impact of the copy process on the source system.
3. You must plan for **where** the data will be stored during processing. This is generally referred to as a staging location.
4. You must plan for **how often** the extraction must be repeated.

After you have determined where your data is coming from and what you want, you will extract that information and place it into a staging location.

**Transform data** –Transforming your data into a uniform, query-able format is really the heart of the ETL process. This phase involves using a series of rules and algorithms to massage the data into its final form. Data cleansing also occurs during this part of the process.Transformations can be basic, such as cleaning data to update formats or to perform data substitutions. This could be replacing NULL values with a zero or replacing the word female with the letter F. These seemingly small changes can have a huge impact on the usefulness of this data to analysts later, in the visualization process.

Transformations can also be more advanced, including applying business rules to the data to calculate new values. Filtering, complex join operations, aggregating rows, splitting columns, and data validation are all very common types of transformations applied at this phase.

**Load data** –The final phase of the ETL process is where you store the newly transformed data. The planning steps you took in the extraction phase will dictate the form the final data store must take. This could be a database, data warehouse, or data lake. After the process has successfully completed, the data in this location is ready to be analysed.

### **ELT process steps**

In modern cloud-based environments, the extract, load, and transform (ELT) approach loads data as it is. It then transforms it at a later stage, depending on the use case and analytics requirements. The steps are similar to those in the ETL process, performed in a different order, but with similar results. The ELT process requires more definition at the beginning. Analytics must be involved from the start to define target data types, structures, and relationships.

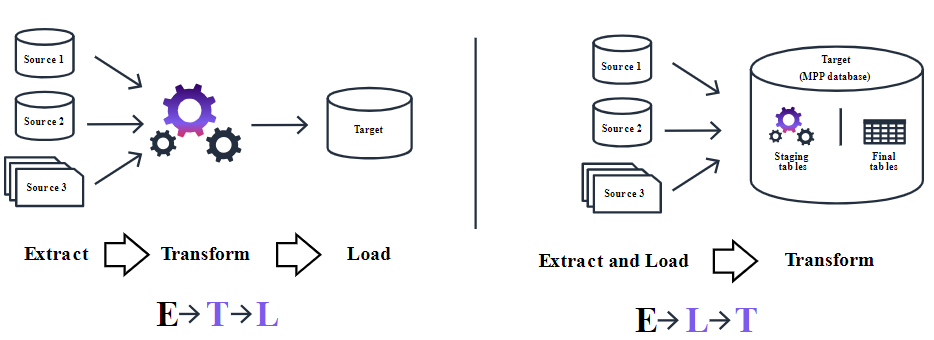
These are the three steps of ELT:

1. Extract raw data from various sources.
2. Load it in its natural state into a data warehouse or data lake.
3. Transform it as needed while in the target system.

With ELT, all data cleansing, transformation, and enrichment occur within the data warehouse. You can interact with and transform the raw data as many times as needed.

### **A comparison of the ETL and ELT processes**

Data scientists mainly use ETL to load legacy databases in the data warehouse, and use ELT with modern databases.



## **1.8. Value**

### **Querying and Reporting**

You have learned about data collection and the speed of collection, data storage, capacity, processing, and analytics. All of these have one thing in common: they do not evaluate the data for usefulness. This is the importance of value. Before making decisions, it is important to extract value from your data. The process of extracting, filtering, and customizing your data can be accomplished by creating queries. Value can be derived from data by querying the data and generating meaningful reports. Analytical reporting is used to transform data into actionable information that empowers organizations to make informed decisions, optimize processes, and achieve strategic objectives. Building a solid report that will provide consumers with what they need to make critical decisions is a bit of an art form.

There are a few steps to be successful:

1. Gather the data, facts, action items, and conclusions.
2. Identify the audience, expectations they have, and the proper method of delivery.
3. Identify the visualization styles and report style that will best fit the needs of the audience.
4. Create the reports and dashboards.

Querying and reporting provide decision-makers with relevant and timely information. They bridge the gap between raw data and actionable insights, so businesses can use their data assets effectively and gain a competitive advantage.

### **Visualizing data**

Analytics exists to help you get the most value and useful insights from raw data. With reporting tools, you can create visual representations of data, such as charts, graphs, and dashboards. Visualization makes complex data more accessible and understandable, helping users quickly identify trends, patterns, and anomalies.

With well-built visual data, you can describe the data, why it is important, and how to move forward with the information provided. To build an effective visual, identify the types of analytics that meet the needs of the analytics request.

Reporting comes in many shapes and sizes. Organize your reports to meet the needs of the consumers of the reports. When creating reports and dashboards, use charts, tables, and graphs to answer questions. The clearer the questions, the better the answers the report or dashboard will provide.

You can break reports and dashboards into pages or views. These pages should have a single theme for all the report elements within them. Provide filters that the report consumer can apply to either the whole page or to individual elements within the page.

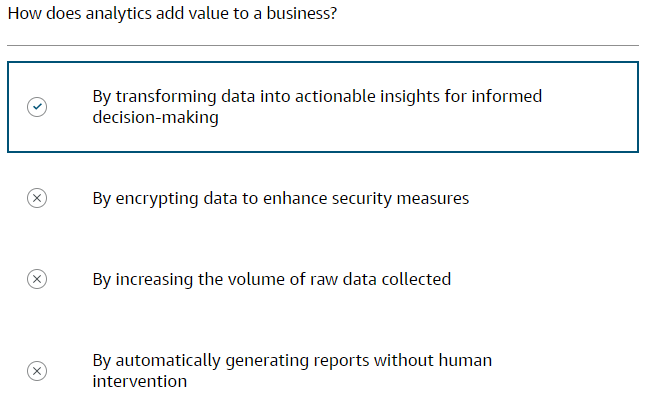
There are three broad types of visual reports: static, interactive, and dashboards.

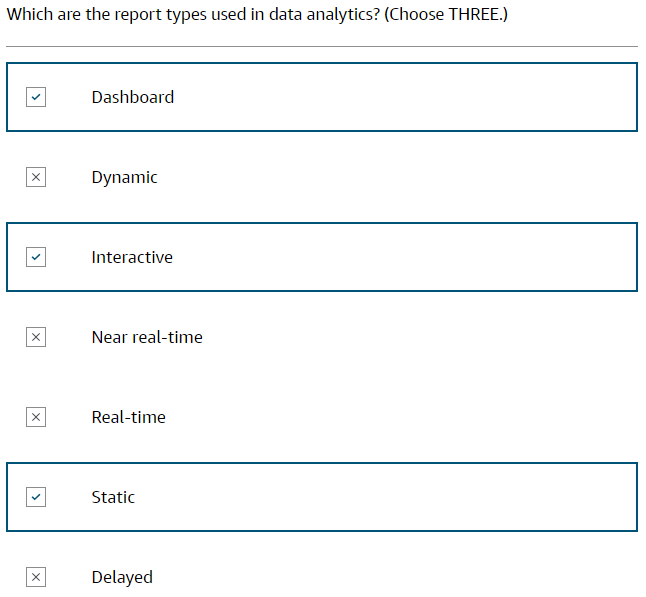
**Static reports** –These types of reports are found in the form of PDFs and PowerPoint slides and can often be accessed through web portals and software interfaces.

**Interactive reports** –These types of reports generally fall under the heading of self-service business intelligence. These reports often take on a print-based report style but have the advantage that consumers can apply filters to charts and graphs, change the scales, and even group and sort values within the reports. A consumer can then tell their own story using the foundation laid by the report builder.

**Dashboards** –This type of visualization is another very popular reporting tool. Whether dashboards are interactive depends on the software used. Consumers find the greatest benefit in dashboards when they focus on high-level roll-ups of key business factors.

### **Knowledge Check**





# **2. AWS Services for Analytics**

The backbone of any good data infrastructure is a scalable and secure storage solution. It must support different types of data collected from multiple sources, and ideally it should reduce expenses.

## **2.1. AWS Services for Volume**

### **AWS services**

* 1. **Amazon S3** - Store any amounts of object storage with scalability, availability & security.
  2. **AWS Lake Formation** - Build, manage, and secure data lakes faster and conveniently.
  3. **Amazon Redshift** - Use cloud data warehousing with the best price performance.

### **Amazon S3**

Secure, reliable, and price effective object storage. Amazon S3 is scalable. It's durable and can be used to store and retrieve any amount of data at anytime from anywhere on the web. It can store the widest variety of data types and data sources—structured, semi-structured, and unstructured. Data can be moved from cloud and on-premises locations to Amazon S3. From there, data can be accessed by AWS and third-party analytics, AI, and ML services.

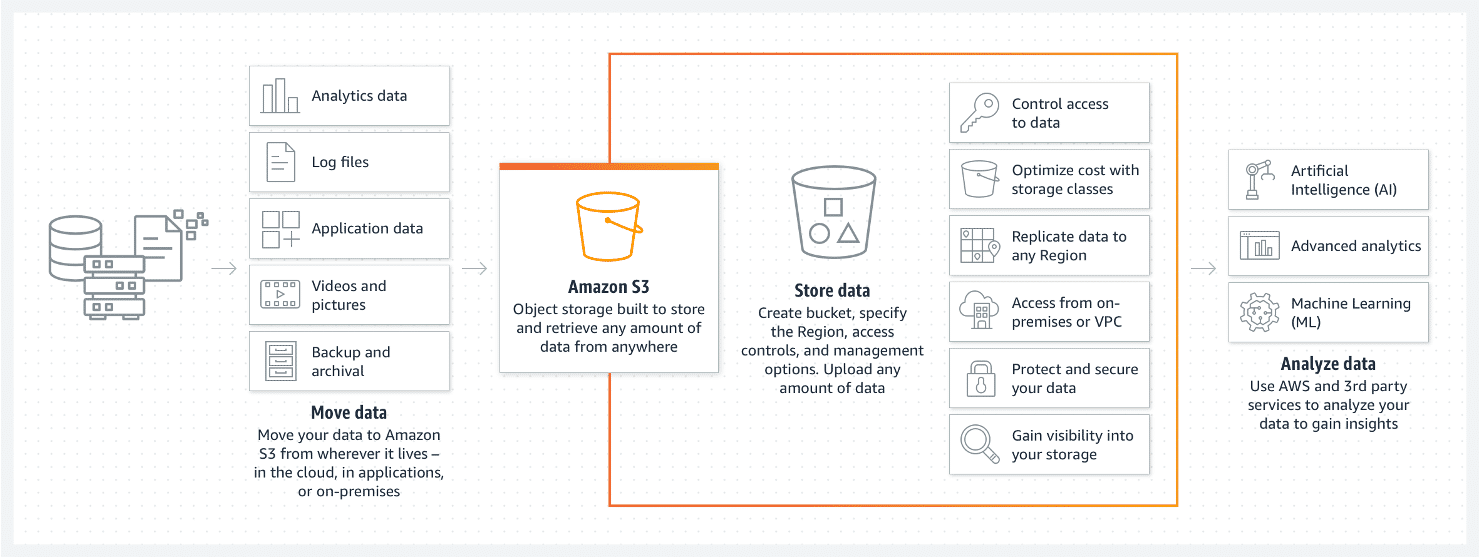
##### **FEATURES**

* 99.999999999% durability
* Global resiliency
* Highly configurable access policies
* HTTP access
* Centrally manage data at scale

##### **BENEFITS**

* Store anything from any source
* Highly scalable
* Secure object storage
* Cost-effective object storage

The following diagram shows where Amazon S3 sits in a generic analytics workflow.



There are three ways Amazon S3 helps as a storage solution for high-volume data needs.

* 1. **Decoupling** – You can separate the way you store data from the way you process it. This is known as decoupling storage from processing. You might have separate buckets for raw data, temporary processing results, and final results.
  2. **Parallelization** – You can access any of these buckets from any process, in parallel, without negatively impacting other processes.
  3. **Centralize datasets** – Amazon S3 becomes a central location to store analytical datasets, providing access for multiple analytic processes at the same time. This allows the solution to avoid the costly process of moving data between the storage system and processing system.

### **AWS Lake Formation**

Data lake—a centralized repository to store all structured and unstructured data. AWS Lake Formation makes it convenient to ingest, clean, catalogue, transform, and secure data and make it available for analysis and ML. With the AWS Lake Formation console, you can discover data sources, set up transformation jobs, and move data to an Amazon S3 data lake and also centralize job orchestrations, and monitor job processing.

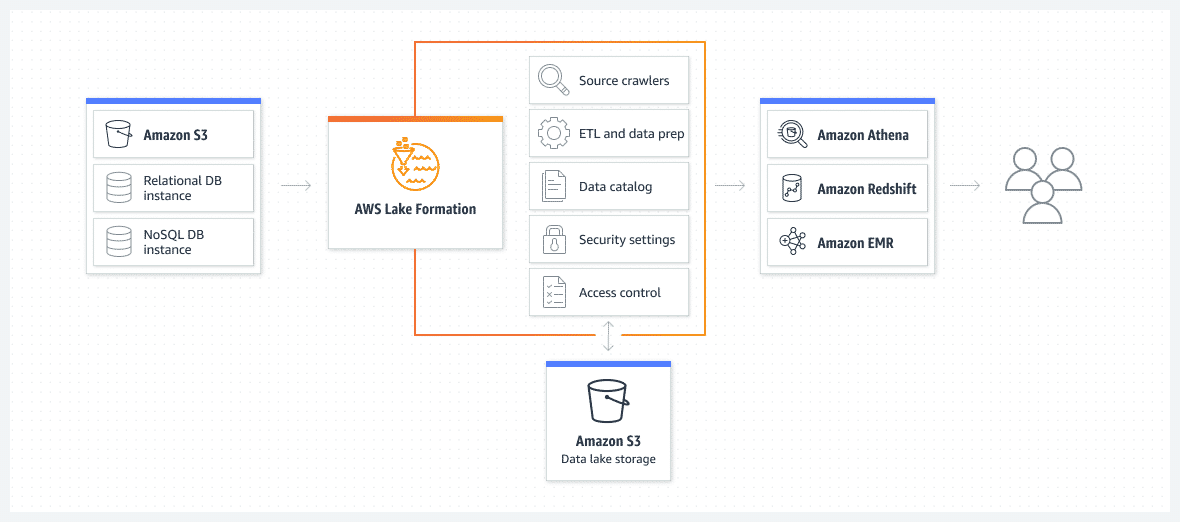
##### **FEATURES**

* Data lake foundation on Amazon S3
* Streamlined and centralized data management
* Straightforward data governance and security
* Enforce permissions with built-in integrations for data integration and big data processing.
* Database style fine-grained permissions on resources.
* Unified Amazon S3 permissions

##### **BENEFITS**

* Build data lakes quickly.
* Remove server management.
* Centrally manage access to datasets
* Streamline security and governance at scale.
* Data movement between data lakes and purpose-built data and analytics services
* Conveniently share your data securely within and outside your organization

Lake Formation in relation to Amazon S3 and Amazon analytics services.



### **Amazon Redshift**

Massive numbers of analytical requests easily overload transactional systems. When large numbers of queries are run against the database, it dramatically slows down insert, update, and delete operations. A specific type of database for large-volume structured data is a data warehouse.

Amazon Redshift is a fast, scalable data warehouse that makes it convenient and cost effective to analyse all your data across your data warehouse and data lake. Amazon Redshift delivers faster performance than other data warehouses by using ML, massively parallel query processing, and columnar storage on high-performance disks. You can run queries across petabytes of data in your Amazon Redshift data warehouse, and exabytes of data in your data lake built on Amazon S3. Amazon Redshift implements columnar indexing to achieve the right performance for analytical workloads.

Data lakes combined with Amazon Redshift data warehouses are an effective way to complement data stored in the data warehouse.

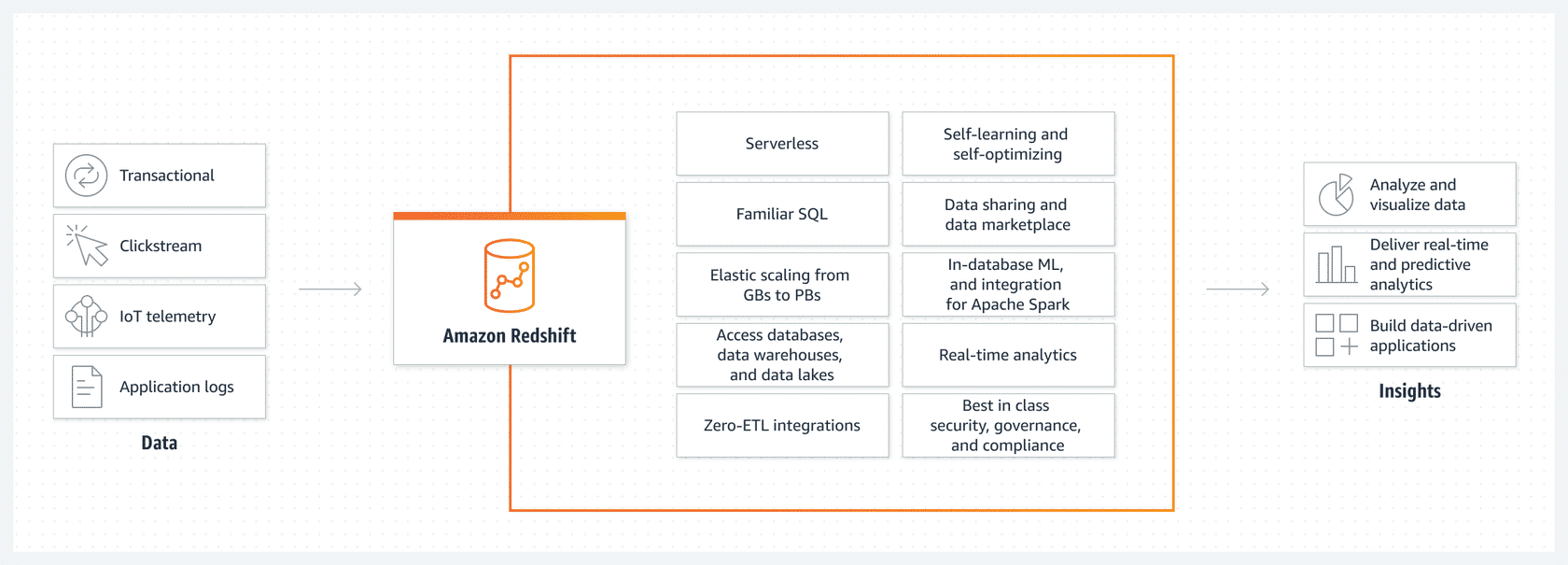
##### **FEATURES**

* Rapidly query datasets ranging in size from gigabytes to petabytes.
* Visualize queries and analysis and share anywhere.
* Automate building, training, and tuning ML models for business intelligence.

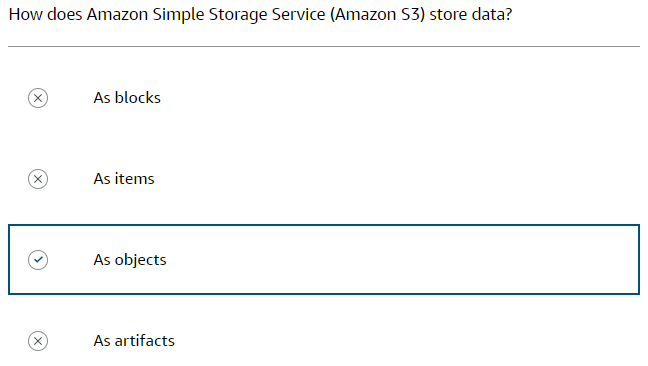
##### **BENEFITS**

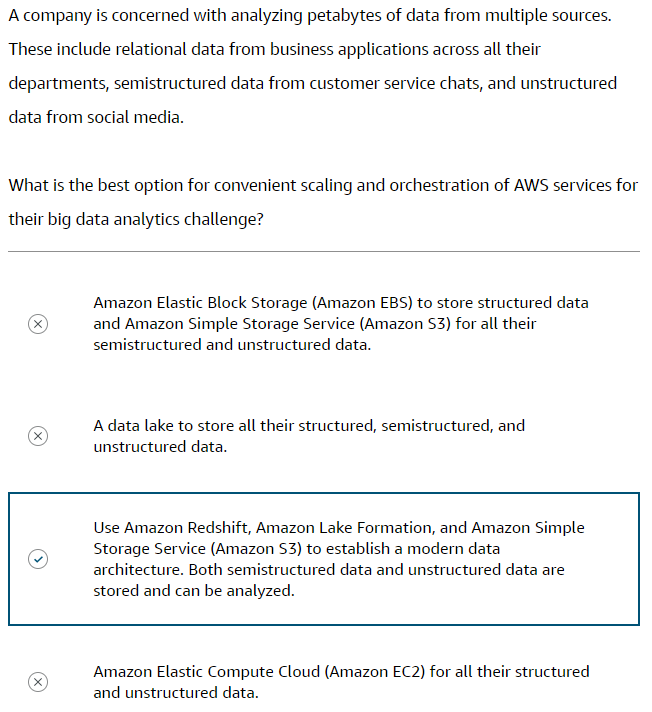
* Efficient storage and high-performance query processing
* Agile scaling of analytics workloads on all your data without managing your data warehouse infrastructure
* Price-performance at scale

The following illustration shows the roles Amazon Redshift plays in analysis workflows.



### **Knowledge check**





## **2.2. AWS Services for Variety**

Traditional systems can run into performance issues when running analytics on a variety of databases (relational, NoSQL) encountered in a production environment. The situation worsens as the volume of different data structures (structured, semi-structured) needed for collection, storage, and processing scales up. Companies need database storage systems that can scale as the data grows, and without sacrificing performance for analytical workloads.

### **AWS solutions**

* 1. **Amazon RDS** - Cloud-based relational database for convenient setup, operation, and scaling
  2. **Amazon Redshift** - Best price-performance for cloud data warehousing
  3. **Amazon OpenSearch Service** - Real-time search, monitoring, and analysis of business and operational data
  4. **Amazon DynamoDB** - Fast, flexible, fully managed NoSQL database for high performance at any scale

### **Amazon RDS**

Amazon RDS solves data variety challenges in terms of vendors, scalability, and performance. Amazon RDS streamlines replication to enhance database availability and improve data durability. You can build a highly efficient OLTP database. The service implements row-based indexing.

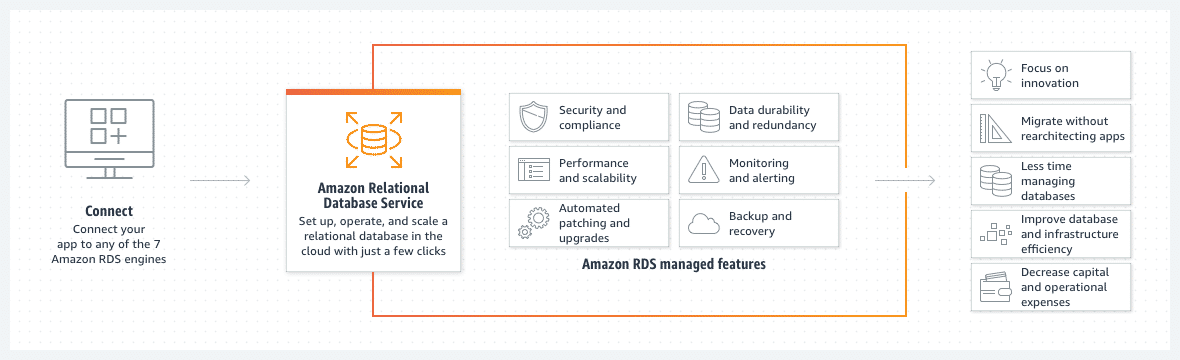
##### **FEATURES**

* Supports for database products you are familiar with (Amazon Aurora, MySQL, PostgreSQL, MariaDB, Oracle, and SQL Server)
* High availability, high throughput, and scalable storage
* Automated hardware provisioning and database setup
* Fully managed backups, software patching, automatic failure detection & recovery.
* Deployment of relational databases in the cloud or on-premises (RDS on Outposts).

##### **BENEFITS**

* Conveniently deploy and scale relational databases in the cloud or on premises.
* Flexible pay-per-use pricing to fit many application use cases.
* Fully managed databases to save time, reduce complexity, and reduce expenses.

The following diagram shows the key features and benefits of Amazon RDS.



### **Amazon Redshift**

Amazon Redshift solves data variety challenges when companies need to query and analyse data across all their departments. Amazon Redshift is a cloud-based data warehouse service on AWS. Amazon Redshift analyses structured and semi-structured data from databases, data lakes, and data warehouses to deliver the best price performance at any scale. You can use Amazon Redshift's capabilities to run analytics on structured data within the data warehouse. When using additional features like Amazon Redshift Spectrum, you can also analyse semi-structured data stored in Amazon S3 data lakes, providing a comprehensive analytics solution. All data becomes visible and available for reports and dashboards. Another key advantage of the AWS services is the ability to use everything in the data lake for ML and real-time analytics to accurately predict future outcomes.

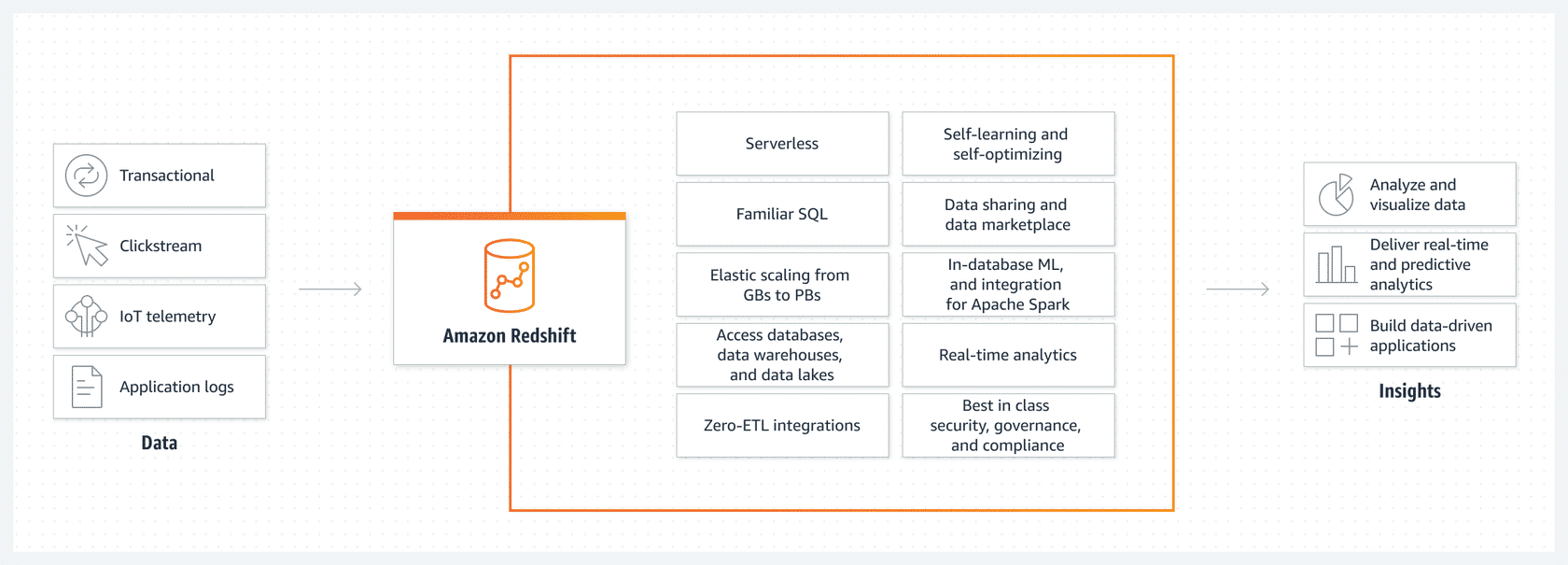
##### **FEATURES**

* Analyses your data across operational databases, data lakes, and data warehouse.
* Provides AWS services integration with AWS databases, analytics, and ML services.
* Queries live data across organizations, accounts, and Regions
* Automatically optimizes data warehouse.
* Provides fully managed, serverless options.

##### **BENEFITS**

* Ability to run and scale analytics quickly on all your data without managing your data warehouse infrastructure.
* Data sharing of live data with users
* Price-performance at scale
* Self-service analytics
* Straightforward data ingestion
* Data science and ML
* Secure and reliable analytics

The following illustration shows some features of Amazon Redshift. You can use SQL to analyse a variety of structured and semi-structured data across data warehouses, operational databases, and data lakes. The outputs of Amazon Redshift are used to visualize data, make predictions, and help build applications. Using hardware and ML designed by AWS, Amazon Redshift delivers the best price performance at any scale.



### **Amazon DynamoDB**

Companies face big data variety challenges with storing their semi-structured data with on-premises NoSQL database management systems. Companies are not able to scale their NoSQL databases infrastructure effortlessly. Amazon DynamoDB solves NoSQL databases variety challenges. Amazon DynamoDB is the fully managed service used for NoSQL data storage and provides fast and predictable performance with seamless scalability.

Companies can use Amazon DynamoDB to create a database table that can store and retrieve any amount of data and serve any level of request traffic. They can scale up or scale down their tables' throughput capacity without downtime or performance degradation. Amazon DynamoDB automatically spreads the data and traffic for the table over a sufficient number of partitions. As the customer makes capacity requests, or the amount of data changes, this helps to maintain consistent and fast performance.

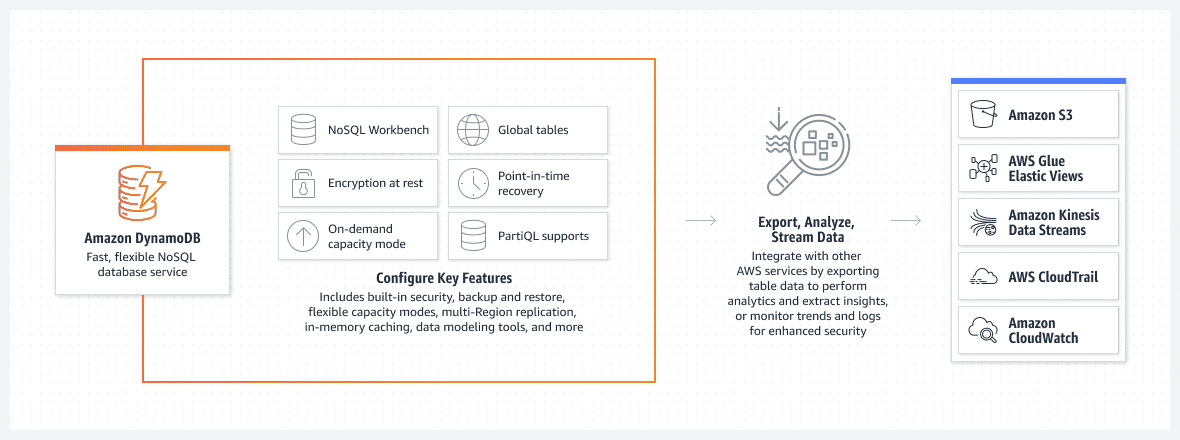
##### **FEATURES**

* Fast, flexible, scalable single-digit millisecond performance NoSQL database service
* Unlimited throughput and storage
* Automatic multi-Region replication
* Secure data with encryption at rest, automatic backup, and restore.
* Reliable single-digit millisecond performance and up to 99.999% availability

##### **BENEFITS**

* Straightforward to get started and use.
* Fully managed, serverless and highly scalable
* Built-in availability, durability, and fault tolerance that cannot be turned off.
* Active-active data replication for multi-Region resiliency
* Security at code level and encryption of data at-rest

The following diagram shows the key features of Amazon DynamoDB and integrations with other AWS services.



With Amazon DynamoDB, you can devote less time to undifferentiated database management tasks, and instead focus on making the database more useful. You can export Amazon DynamoDB table data to numerous other AWS services for storage, normalization, analysis, and monitoring.

### **Amazon OpenSearch Service**

Organizations with lots of variety of data to analyse might want to use an open-source search suite to provide fast response times to their customers' search requests. Amazon OpenSearch Service makes it convenient to perform interactive log analytics, real-time application monitoring, and website search. Amazon OpenSearch Service is an AWS managed service to run and scale OpenSearch clusters. With this service, companies do not have to worry about managing, monitoring, and maintaining infrastructure, or having to build in-depth expertise in operating OpenSearch clusters.

OpenSearch is an open source, distributed search and analytics suite for a broad set of use cases. OpenSearch provides a fast and highly scalable system for exploring and visualizing data, with convenient OpenSearch Dashboards. Amazon OpenSearch Service supports integration with streaming data from Amazon S3 buckets, Amazon Kinesis Data Streams, and DynamoDB Streams.

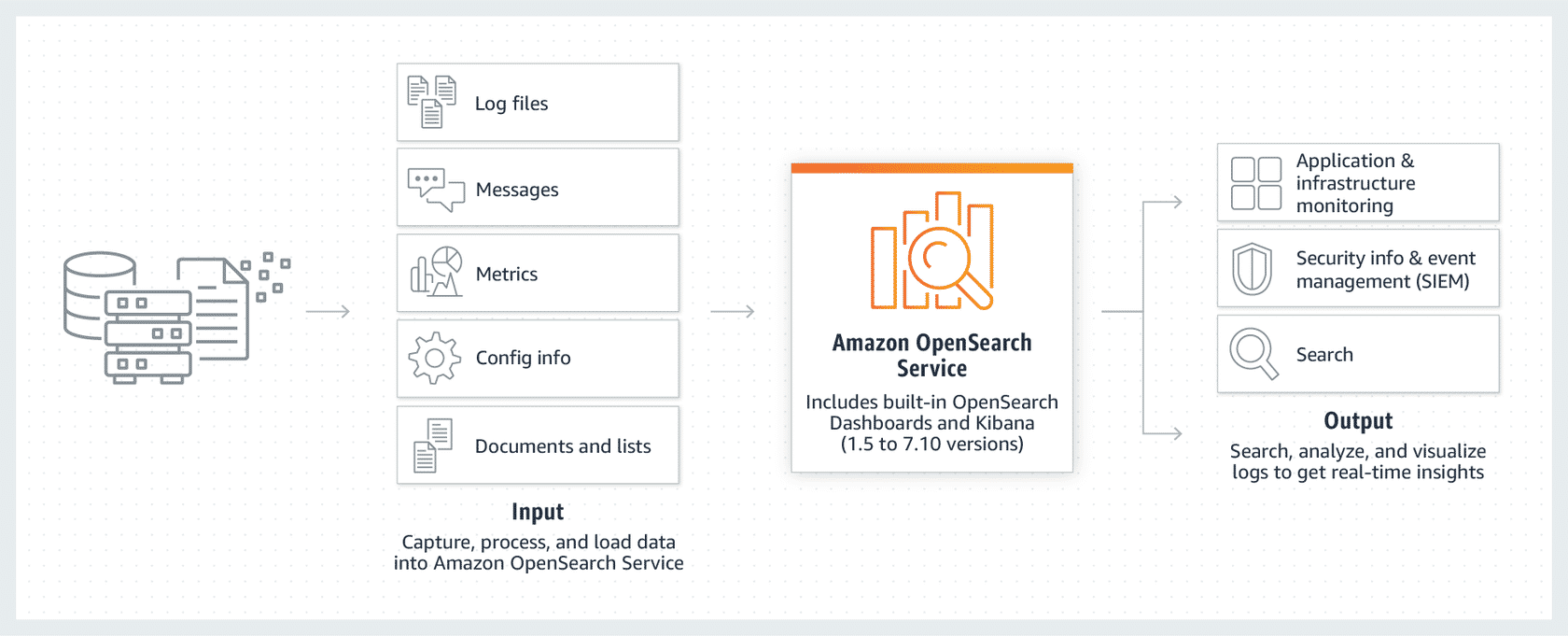
##### **FEATURES**

* Analyse activity logs for customer facing applications or websites.
* Analyse product usage data coming from other services and systems.
* Analyse social media sentiments, CRM data, and find trends for your organization or service.
* Monitor usage for mobile applications.

##### **BENEFITS**

* Use ML to detect anomalies in real time, autotune your clusters, and personalize your search results.
* Quickly search and analyse your unstructured and semi-structured data to effortlessly find what you need.
* Eliminate operational overhead and reduce cost with automated provisioning, software installation, patching, storage tiering, and more.

The following diagram shows how Amazon OpenSearch Service works.



Amazon OpenSearch Service captures and processes data from numerous sources, and provides analysis, visualization, and monitoring outputs.

## **2.3. AWS Services for Velocity**

### **AWS solutions**

* 1. **Amazon EMR** - Big data solution for petabyte-scale data processing, interactive analytics, and ML
  2. **Amazon MSK** - Fully managed, highly available Apache Kafka service
  3. **Amazon Kinesis** - Cost-effective service to processes and analyse streaming data.
  4. **AWS Lambda** - Serverless, event-driven compute service that lets you run code.

### **Amazon EMR**

Amazon EMR helps you handle velocity challenges by providing a scalable, managed, big data platform that efficiently processes and analyses high volumes of data at varying velocities. Streamlines the processing of large datasets and manages the infrastructure.

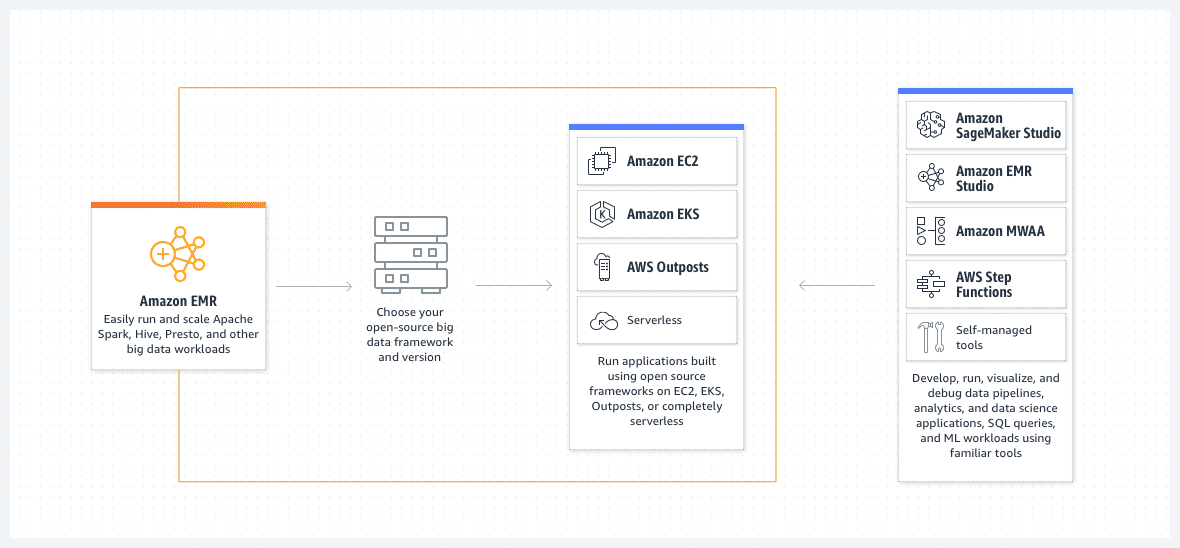
##### **FEATURES**

* Runs big data applications and petabyte-scale analytics quickly.
* Seamlessly integrates with Amazon SageMaker.
* Performs machine learning tasks on large datasets.
* Uses open-source big data frameworks (Spark, Hadoop, HBase, Hive, Hudi, Presto) to distribute data processing tasks.
* Uses parallel processing capabilities to ensure that data can be ingested, transformed, and analysed rapidly.

##### **BENEFITS**

* Highly scalable, fully managed service
* Integrates with AWS services.
* Optimized performance
* Robust security

The following diagram shows to use Amazon EMR to process & analyse big data workloads.



Use Amazon EMR for processing massive data sets on parallelized compute clusters, including Amazon EC2 instances, Amazon Elastic Kubernetes Service (Amazon EKS) clusters, and AWS Outposts on premises. Amazon EMR integrates with Amazon ML services, and can also be used in event-initiated, serverless workflows.

### **Amazon MSK**

With Amazon MSK, you can process high-velocity data streams with convenience and reliability. It is a fully managed service that helps you build and run applications that use Apache Kafka for real-time streaming and event-driven architectures. Apache Kafka is an open-source, high performance, fault-tolerant, and scalable stream-processing platform designed to process real-time data sources.

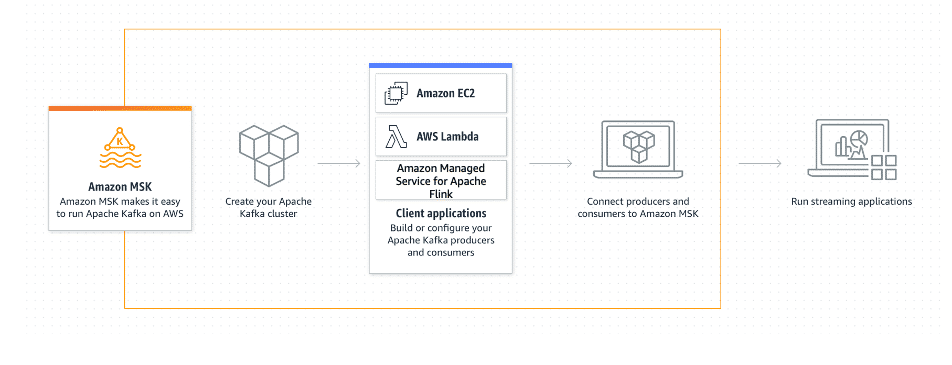
##### **FEATURES**

* Provisions servers, configures Apache Kafka clusters, and replaces servers when they fail.
* Orchestrates server patches and upgrades.
* Architects clusters for high availability and ensures data is durably stored and secured.
* Sets up monitoring and alarms.
* Runs scaling to support load changes.
* Provides the operations for creating, updating, and deleting Apache Kafka clusters.

##### **BENEFITS**

* Fully managed service
* Automatically scales.
* Highly secure
* Configurable clusters

The following diagram shows how you can use Amazon MSK to ingest and process log and event streams, run streaming applications and power event-driven systems.



Amazon MSK streamlines large-scale data streaming analysis, using Apache Kafka platforms on Amazon EC2 clusters. You can process and analyse real-time data streams for content producers and other distribution channels.

### **Amazon Kinesis**

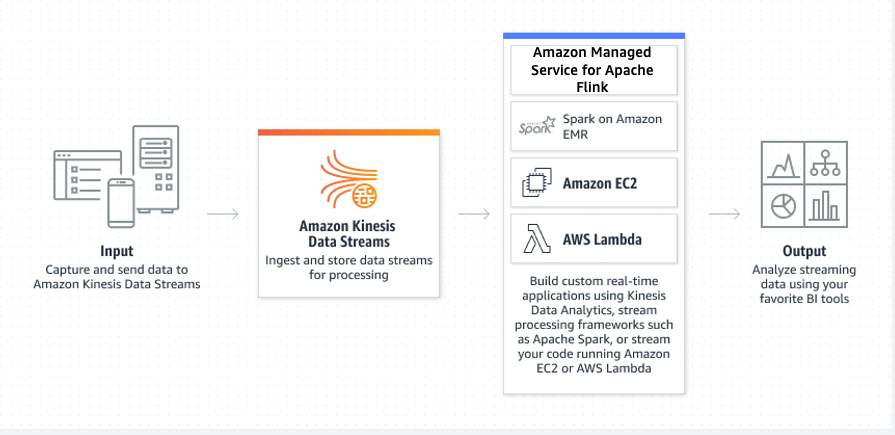
Amazon Kinesis is a fully managed platform that addresses velocity challenges by ingesting, processing, and analysing real-time data. You can collect, process, and analyse data as it is generated. This high-velocity data can then be used for ML, real-time analytics, and other applications to gain valuable insights. Amazon Kinesis Data Streams ingests and stores data from real-time streams so it can be prepared for analysis by downstream services. Amazon Kinesis Data Streams works with AWS Lambda to create event-driven, serverless workflows.

##### **FEATURES**

* Collects, processes, and analyses real-time streaming data.
* Helps you receive timely insights and react quickly to new information.
* Makes it convenient to capture, process, and store data streams at any scale.

##### **BENEFITS**

* Serverless
* Highly available and durable
* Low latency
* Secure and compliant
* Dedicated throughput per consumer



##### **Data streaming services**

| **Service** | **Description** |
| --- | --- |
| Amazon Kinesis Data Streams | A data streaming service that continuously captures data in real time from hundreds of thousands of sources. |
| Amazon Kinesis Data Firehose | Near real-time analytics with existing business intelligence tools by capturing, transforming, and loading data streams into AWS data stores. |
| Amazon Managed Service for Apache Flink | Build and run Apache Flink applications, and query and analyze streaming data without setting up infrastructure and clusters. |

### **AWS Lambda**

AWS Lambda is a serverless compute service that helps address velocity challenges in data processing with real-time and event-driven data processing. Lambda runs code in response to events, such as changes to data in an Amazon S3 bucket or updates in a DynamoDB table. Events and changes in data can generate quick responses.

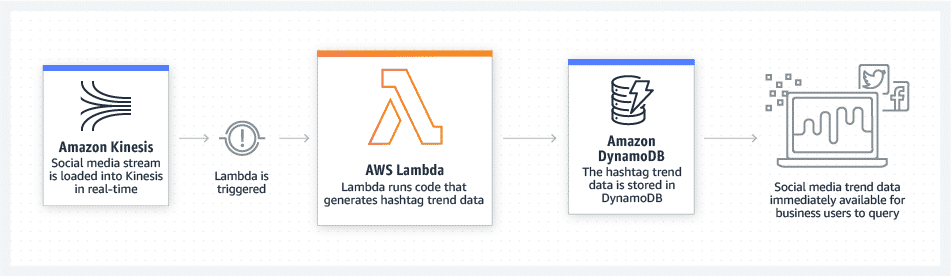
##### **FEATURES**

* Promotes an event-driven architecture, so you can design applications that respond to data changes, user interactions, or other events.
* Builds responsive systems that react to changes in data velocity without the need for constant monitoring or intervention.

##### **BENEFITS**

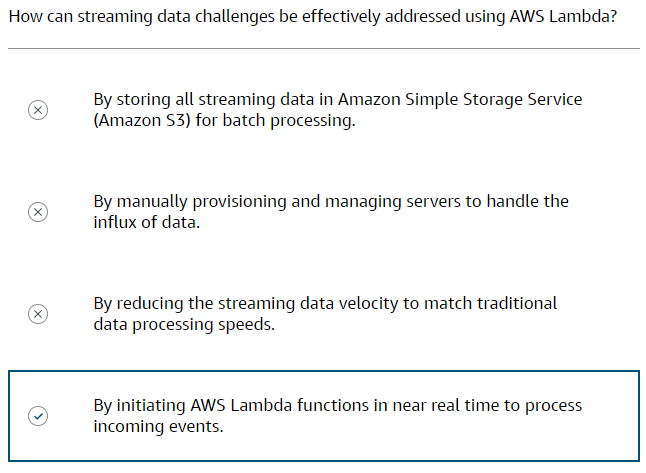
* Built-in fault tolerance
* Automatic scaling
* Connects to relational databases.
* Allows for building custom backend services.

Lambda processes data in real-time, scales automatically for varying data velocities, and integrates with other AWS services to build robust data processing pipelines and applications.



You don't need to invoke or configure servers. Instead, events (such as the presence of a data stream) initiate a workflow, such as ingestion into an Amazon DynamoDB instance. From there, data can be output for analysis, query, and other business uses.

### **Knowledge Check**



## **2.4. AWS Services for Veracity**

Organizations need to ensure the integrity of their data at all phases of the data lifecycle. Organizations must have accurate data as it enters their system by going through a data cleansing process. Also, organizations must secure the cleansed data and prevent it from being tampered with as it is shared and analysed.

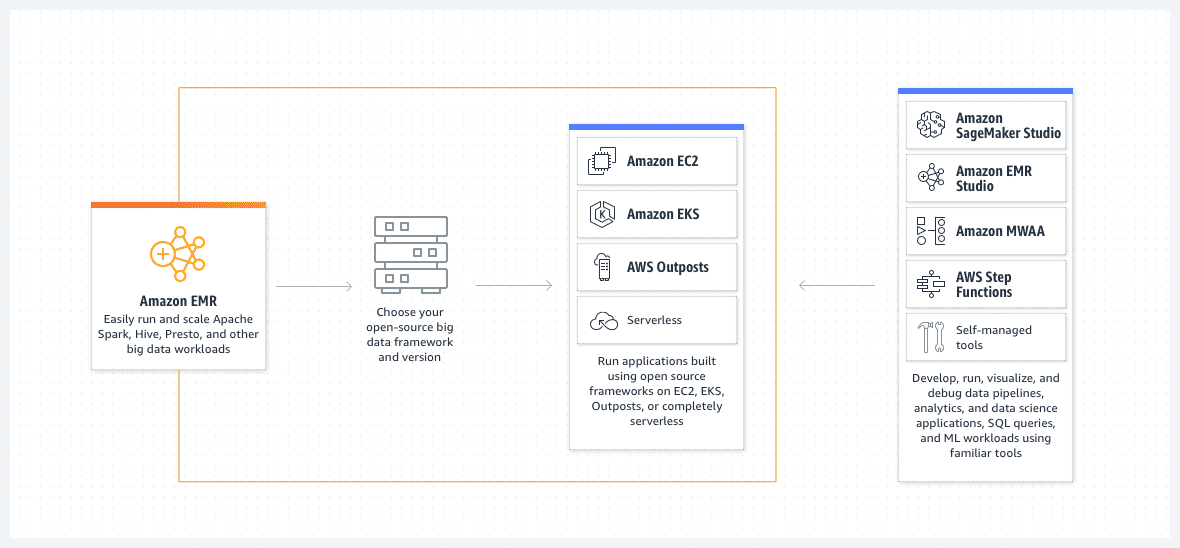
### **AWS solutions**

1. **Amazon EMR** - Streamline collection and processing data for big data workloads.
2. **Amazon Glue** - Prepare and integrate all your data at any scale.
3. **AWS Glue DataBrew** - Clean and normalize data faster and more efficiently.
4. **Amazon DataZone** - Share data across your organization with built-in governance.

### **Amazon EMR**

Amazon EMR helps you overcome data quality, accuracy, and integrity challenges. This service provides a robust data collection and processing platform to analyze vast amounts of data. Amazon EMR helps you focus on transforming and analysing your data, so you don't have to worry about managing infrastructure. Amazon EMR is a hands-on approach to creating your data pipeline and requires your team to have strong technical know-how. You can create a more customized pipeline to fit your business needs.

The following diagram shows how Amazon EMR connects with other AWS services, and which tools are available to build data pipelines, analytics, and ML workloads.



Use Amazon EMR for processing massive data sets on parallelized compute clusters, including Amazon EC2 instances, Amazon EKS clusters, and AWS Outposts on premises. Amazon EMR integrates with Amazon ML services, and can also be used in event-initiated, serverless workflows.

### **AWS Glue**

AWS Glue helps you overcome data quality and data integrity challenges. AWS Glue is a serverless data integration and managed ETL service. This service provides a more streamlined experience than Amazon EMR. AWS Glue makes it convenient to clean and normalize data directly from data lakes, data warehouses, and databases. AWS Glue can consume from streaming sources, clean, and transform it in-flight, and share it for analysis in seconds in your desired data store. You can process event data like Internet of Things (IoT) event streams, clickstreams, and network logs, and run a variety of complex analytics and ML operations.

AWS Glue can be used as a metastore for your final transformed data by using the AWS Glue Data Catalog. You can manage data quality in datasets with AWS Glue Data Quality. This service analyses your chosen dataset and recommends data quality rules that you can optimize. You can set up data quality rules in your data catalogue to manage data quality reactively and also from your data pipelines for proactive data quality.

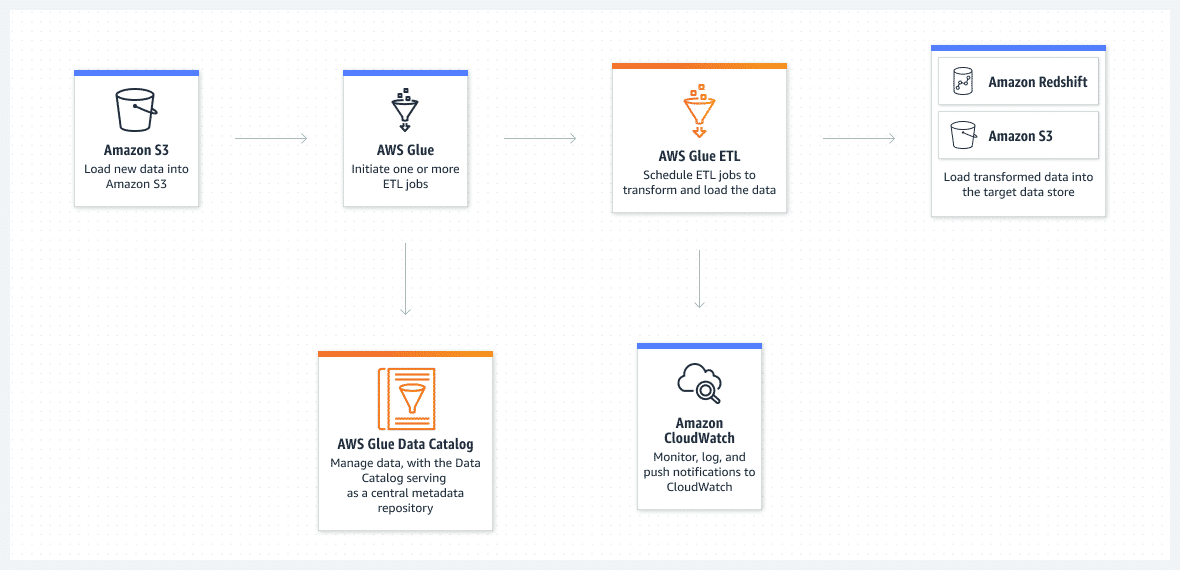
##### **FEATURES**

* Serverless, scalable, high-performing data integration service from multiple sources
* Highly integrated with many AWS services
* Supports multiple workloads, including batch, micro-batch, and streaming.
* Rule recommendations
* Built-in data quality rules and actions
* Data Quality Definition Language (DQDL)

##### **BENEFITS**

* Save time having to optimize the number of resources.
* Reduce costs of idle resources.
* Improve your data quality.
* Clean and prepare data for analysis without ML expertise.
* Streamline data quality rule authoring.

The following diagram illustrates how AWS Glue functions with other AWS services in ETL.



AWS Glue can run your ETL jobs as new data arrives. For example, you can configure AWS Glue to initiate your ETL jobs to run as soon as new data becomes available in Amazon S3.

### **AWS Glue DataBrew**

AWS Glue DataBrew helps with data veracity challenges. It is a visual data preparation tool that helps data analysts and data scientists clean and normalize data to prepare it for analytics and ML. With AWS Glue DataBrew, you can visually map the lineage of your data to understand the various data sources and transformation steps that the data has been through.

AWS Glue DataBrew gives prebuilt transformations to automate data preparation tasks, all without the need to write any code. You can automate filtering anomalies, converting data to standard formats, and correcting invalid values, and other tasks. After your data is ready, you can immediately use it for analytics and ML projects.

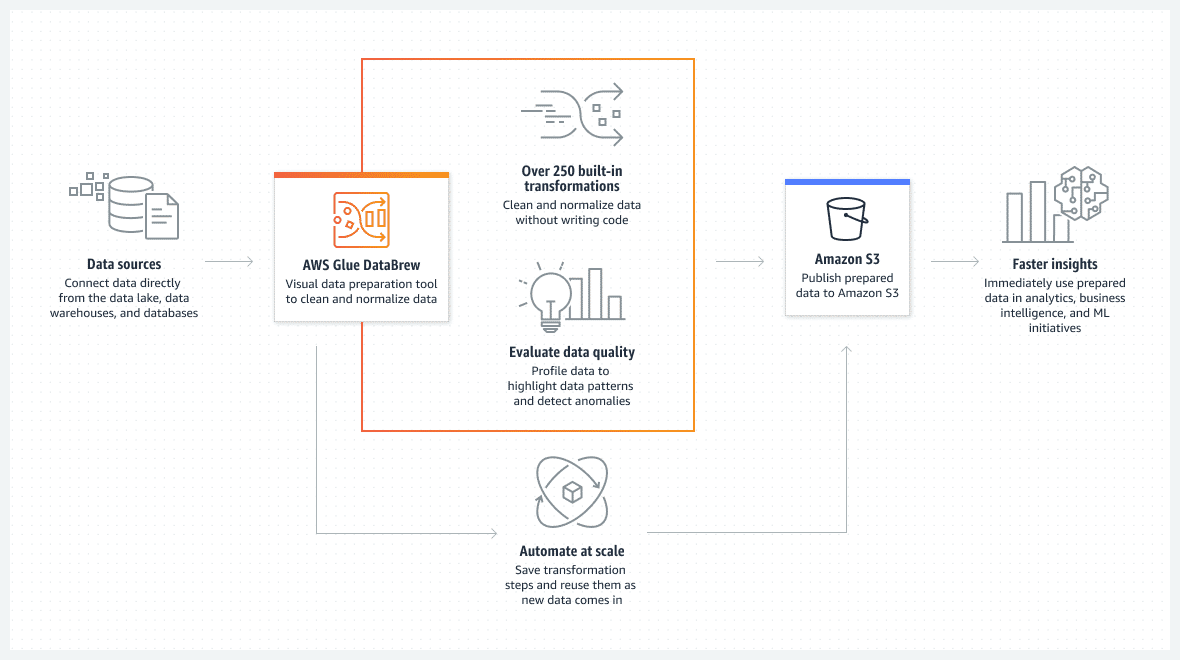
##### **FEATURES**

* Connects directly to AWS services, including Amazon S3, Amazon Redshift, AWS Lake Formation, Amazon Aurora, and Amazon RDS
* 250+ prebuilt transformations to automate data preparation tasks using no-code.
* Automate filtering anomalies, convert data to standard formats, and correct invalid values.

##### **BENEFITS**

* Convenient to clean and normalize data for analytics and ML.
* Pay for what your use with no upfront commitment.

The following diagram illustrates the interaction of AWS Glue DataBrew with data lake, data warehouses, and databases.



AWS Glue DataBrew helps you to explore your data that connects directly from your data lake, data warehouses, and databases.

### **Amazon DataZone**

Amazon DataZone is a data management service to catalogue, discover, govern, share, and analyze your data. An integrated analytics portal gives you a personalized view of all your data while enforcing your governance and compliance policies at scale. Administrators and data stewards who oversee your organization's data assets can manage and govern access to data using fine-grained controls.

With Amazon DataZone, you can share and access your data across accounts and supported Regions. Amazon DataZone extends governance controls through AWS Glue Data Catalog, AWS Identity and Access Management (IAM), and AWS Lake Formation. The service operates within your infrastructure without relying on individual credentials.

##### **FEATURES**

* Search for published data and request access to work on projects.
* Collaborate with project teams through data assets.
* Manage and monitor data assets across projects.
* Ensure the right data is accessed with a governed workflow.
* Access analytics with a personalized view for data assets through a web-based application or API.

##### **BENEFITS**

* Convenient to discover, prepare, transform, analyze, and visualize data.
* Centralized governance of your data according to your organization’s security regulations.
* Convenient to search, share, and access data stored on AWS, on premises, or with third-party providers.

The following diagram shows how Amazon DataZone works to share, search, and discover data at scale across organizational boundaries. Data producers’ catalogue and share data with decentralized and federated ownership to Amazon DataZone.



Amazon DataZone integrates with Amazon Redshift and AWS Lake Formation to provide access control. You can use analytics tools such as Amazon Athena query editor and the Amazon Redshift query editor.

## **2.5. AWS Services for Value**

In workloads that generate vast amounts of data, you need tools to derive valuable insights from that data.

### **AWS solutions**

1. **Amazon QuickSight** - Unified business intelligence at cloud-scale.
2. **Amazon SageMaker** - Build, train, and deploy ML models for any use case with fully managed infrastructure, tools, and workflows.
3. **Amazon Bedrock** - Build and scale generative AI applications with foundation models.
4. **Amazon Athena** - Analyze petabyte-scale data where it lives.

### **Amazon QuickSight**

Amazon QuickSight is a generative business intelligence (BI) and data visualization tool that helps you gain value from data through analytics. By using Integrations with ML tools, you can display predictions and other ML insights within interactive dashboards.

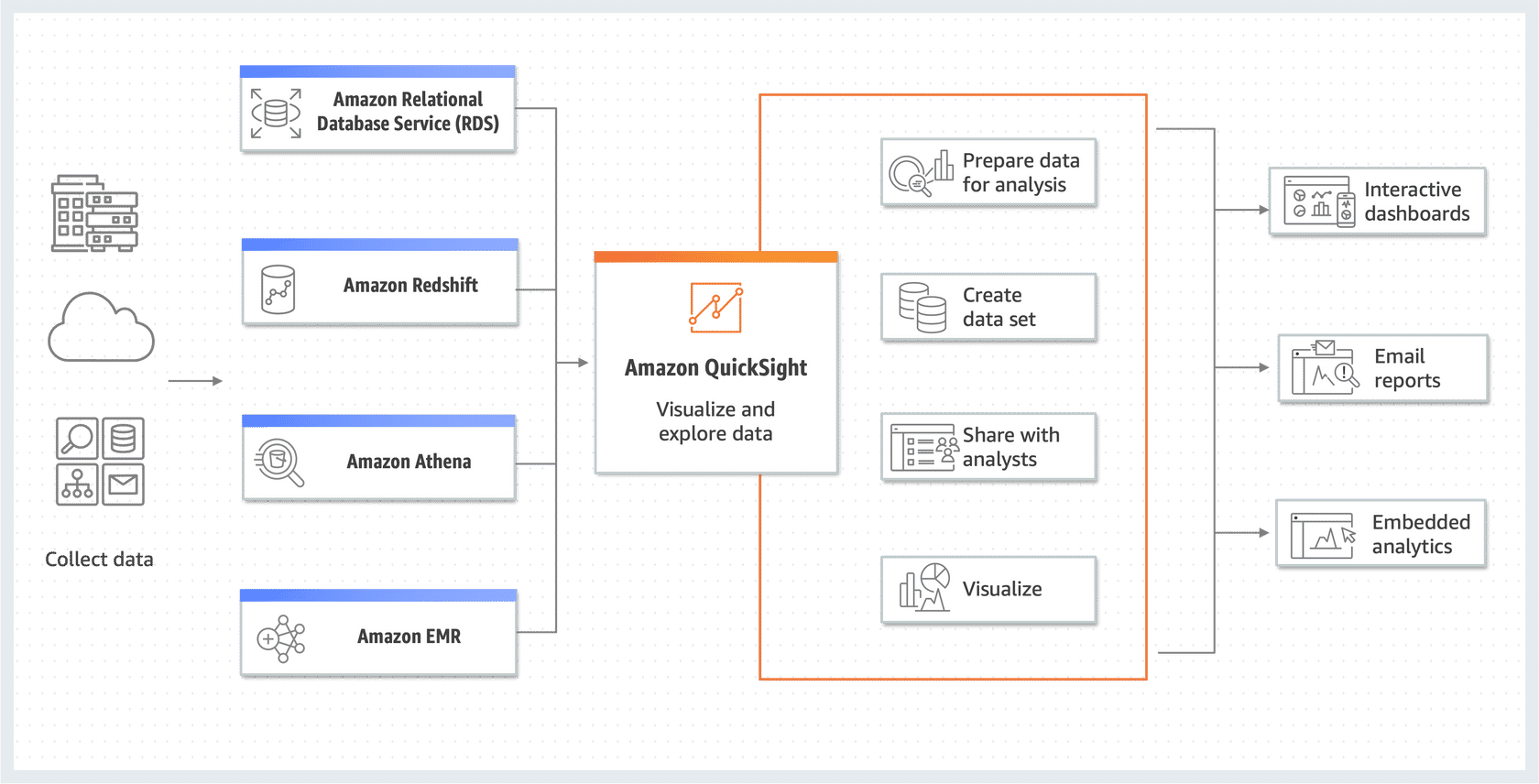
##### **FEATURES**

* Conveniently create interactive dashboards, paginated reports, embedded analytics, and use natural language queries.
* Supports integration with several data sources, including Amazon S3, Amazon Redshift, Amazon RDS, Amazon Athena, and third-party databases.
* Can connect to on-premises data sources.
* Cleans, transforms, and shapes data before creating the visualizations.
* Combine multiple visualizations into interactive dashboards.
* Integrates with Amazon SageMaker to incorporate ML models into visualizations.

##### **BENEFITS**

* Scales to thousands of users
* Built-in security and compliance
* Extensive API capabilities
* Broad data source support

Below diagram illustrates where Amazon QuickSight Workflow sits in analytics workflows.



### **Amazon SageMaker**

Amazon SageMaker helps you solve data challenges by providing a comprehensive ML platform that helps build, train, and deploy ML models. After building and integrating ML algorithms, SageMaker provides tools to monitor the performance of deployed models, track metrics, and set up alarms. This ensures the predictive models that have been created continue to provide accurate results over time. With SageMaker, you can extract valuable insights and drive innovation from their data efficiently. It streamlines the process of building and deploying predictive models, allowing organizations to harness the power of ML for data-driven insights.

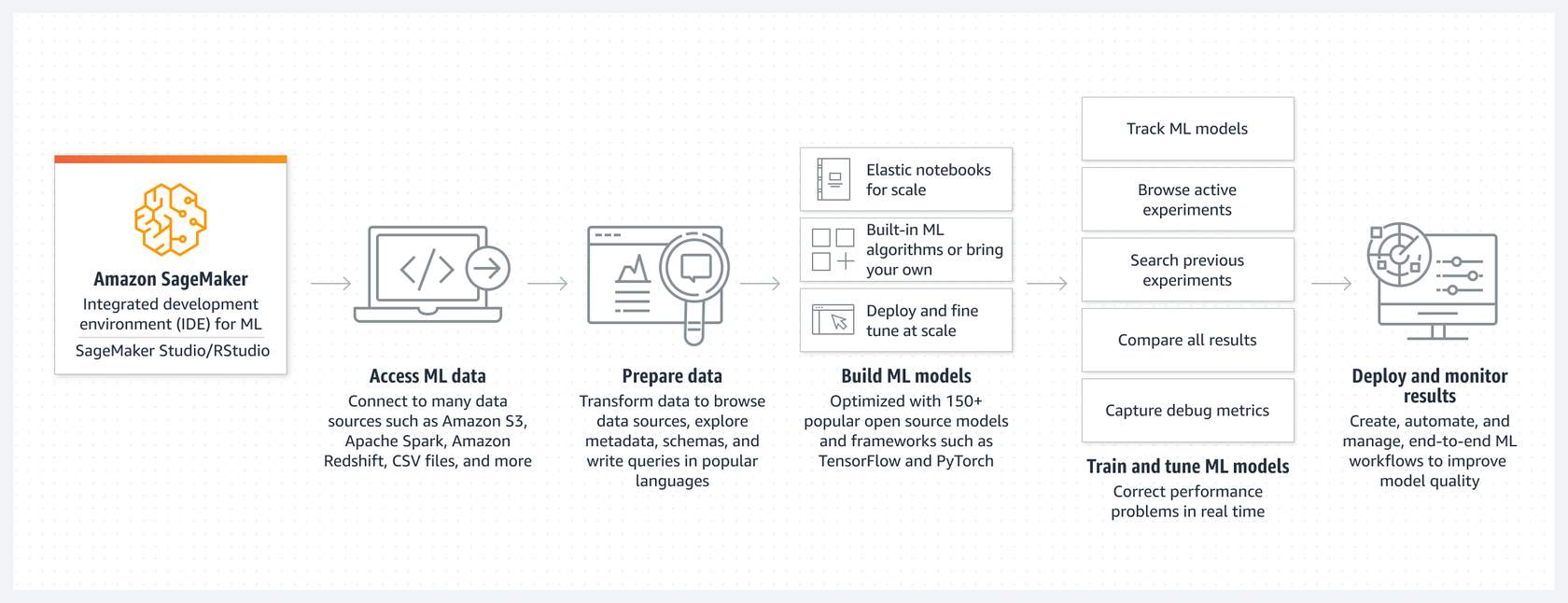
##### **FEATURES**

* Data cleaning and transformation is taken care of using data preprocessing tools.
* Access, label, and process large amounts of structured data and unstructured data to ensure it's ready for ML.
* Offers built-in ML algorithms that are optimized for performance and can be conveniently applied to predictive analytics tasks.
* Provides the option to create custom ML models.
* You can deploy predictive models as a highly available and scalable endpoint.
* Conveniently integrate your model predictions into other applications or websites to make predictions on new data.

##### **BENEFITS**

* Fully managed service
* Highly scalable
* Built-in data processing
* Integrations with AWS services
* Secure and compliant
* ML model tuning and optimization
* Multi-model endpoints

The following diagram shows the key features of Amazon SageMaker.

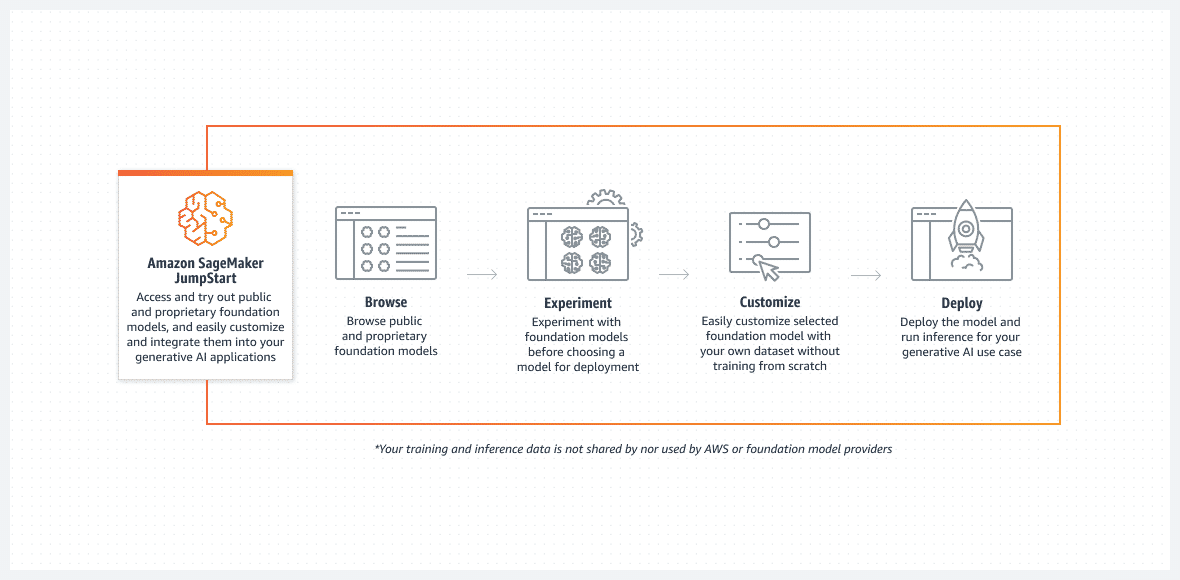


With Amazon SageMaker, you can connect with multiple data sources, and then prepare and transform the data.

##### **Amazon SageMaker JumpStart**

Amazon SageMaker JumpStart is an ML hub with foundation models, built-in algorithms, and prebuilt ML solutions. SageMaker JumpStart offers several foundation models from various model providers and provides hundreds of built-in algorithms with pre-trained models from model hubs. SageMaker JumpStart also offers ready-to-use solutions for common use cases, so you can quickly implement ML applications. These solution templates and built-in algorithms promote faster innovation, empowering you to focus on problem solving, rather than infrastructure and code.

The following diagram shows how SageMaker JumpStart helps you access and deploy foundation models.



### **Amazon Bedrock**

With Amazon Bedrock, organizations can build and scale generative AI applications with foundation models. Amazon Bedrock is a fully managed service and offers base models from leading AI companies (such as AI21 Labs, Anthropic, Cohere, Meta, Stability AI) with a single API. Since Amazon Bedrock is serverless, organizations can securely integrate and deploy generative AI into their applications using AWS services. With the power of AI, organizations can focus on automatically gaining value from their data.

##### **FEATURES**

* Choose from a variety of foundation models.
* Privately customize foundation models.
* Conveniently integrate using a single API.
* Connect foundational models to data sources.
* Automatic data source detection.

##### **BENEFITS**

* Data security and compliance certifications
* Convenient model customization
* Build fully managed agents that initiate business tasks.
* Convenient-to-use developer experience

### **Amazon Athena**

Amazon Athena is a serverless, interactive analytics service built on open-source frameworks. Athena helps solve data value challenges so you can conveniently query and analyze data stored in Amazon S3 using standard SQL queries. Athena is designed for interactive analytics, running queries, and getting results in real time. This is especially valuable for one-time queries and exploratory data analysis. Athena supports rapid and cost-effective data exploration, which in turn helps you derive valuable insights. Athena's serverless and scalable structure, integration with other AWS services and support for various data formats make it a powerful tool for gaining value from data.

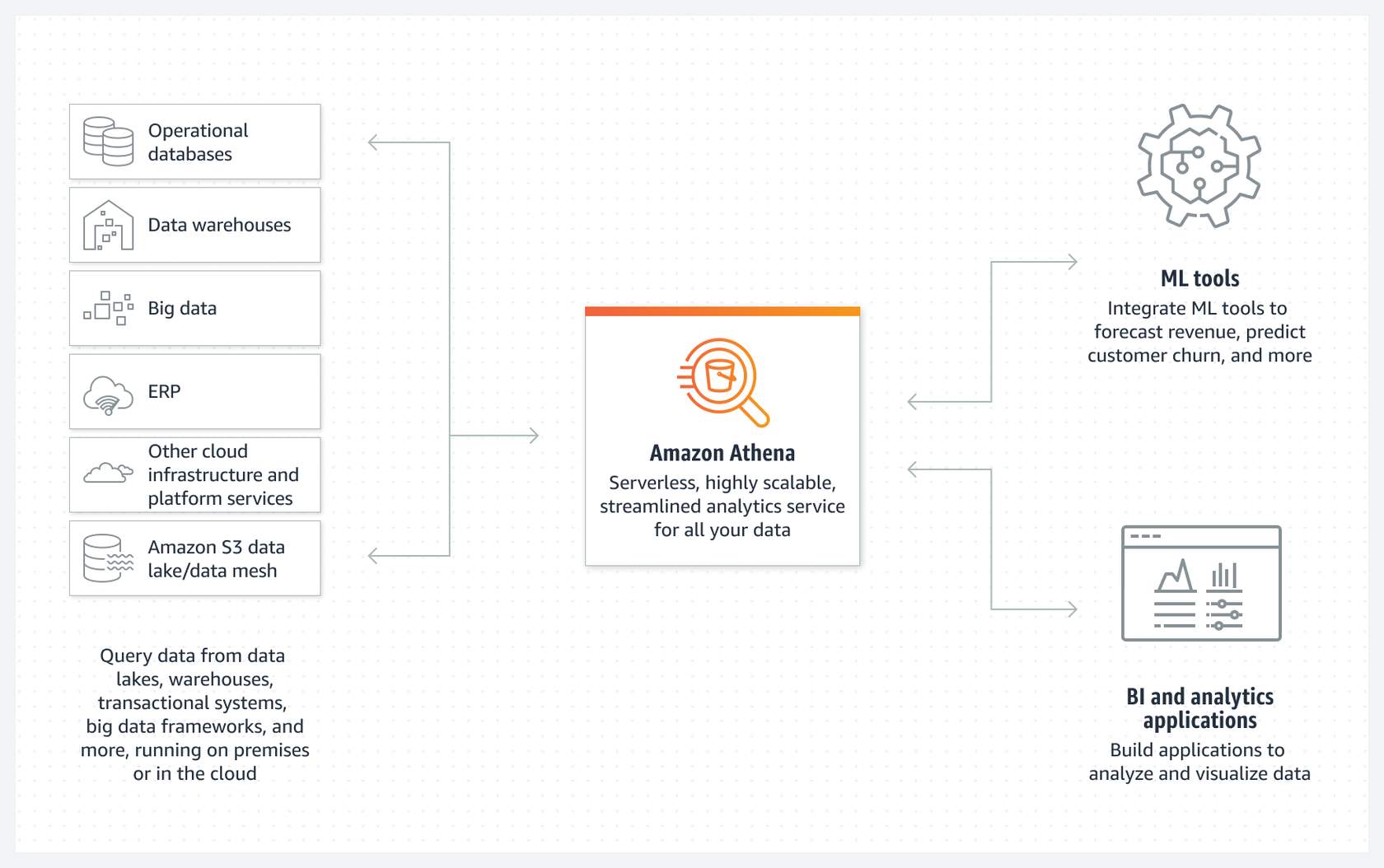
##### **FEATURES**

* Provides a streamlined, flexible way to analyze petabytes of data without the need to set up and manage infrastructure.
* Saves query history and results, making it convenient to review past queries, analyze past performance, and troubleshoot discrepancies.
* Integrates out-of-the-box with AWS Glue
* Controls access to data by using IAM policies.

##### **BENEFITS**

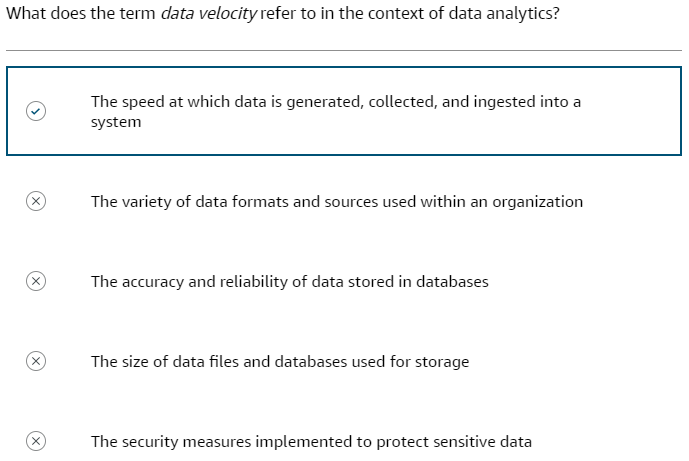
* Serverless
* Fast performance
* Secure
* Highly available and durable
* Convenient to query
* Integrates with other AWS services.

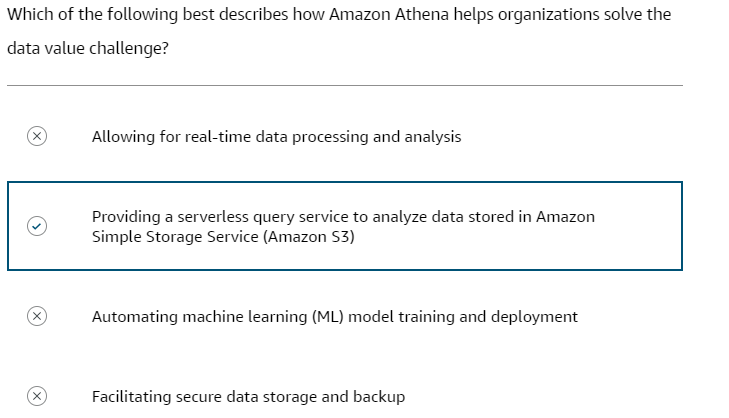
The following diagram shows how Amazon Athena analyses data where it lives.

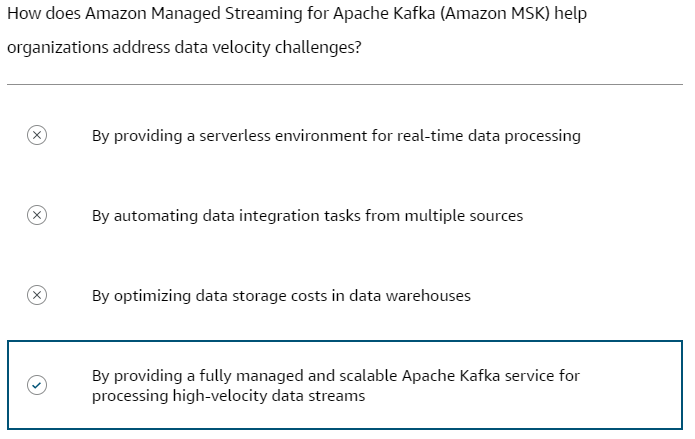


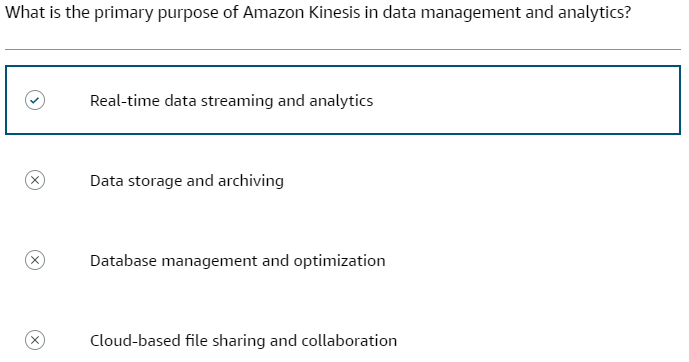
With Amazon Athena, you can query data where it lives, including data lakes, data warehouses, and big data frameworks. Amazon Athena integrates with ML tools, analysis applications, and visualization applications.

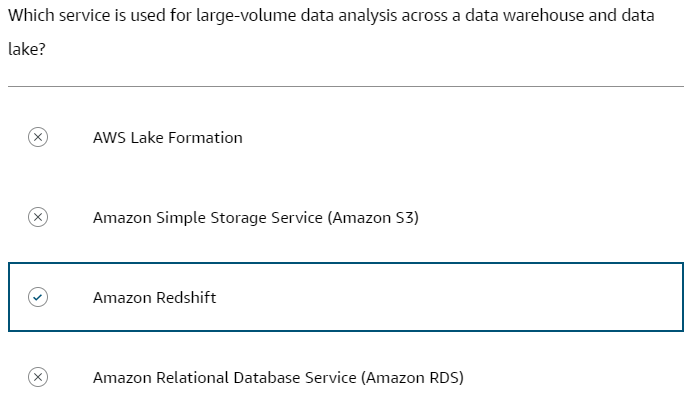
# **3. Assessment -1**

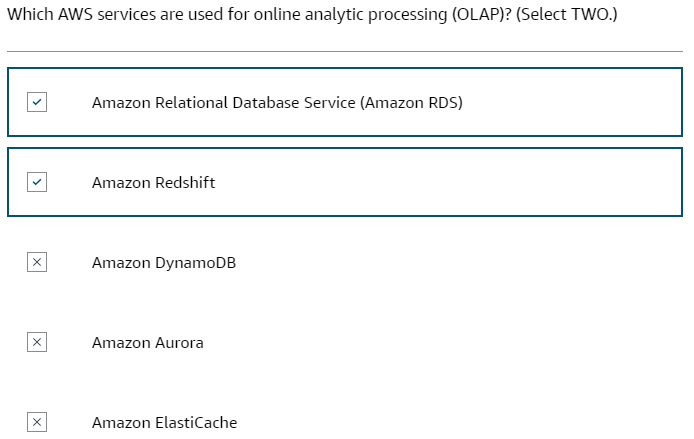


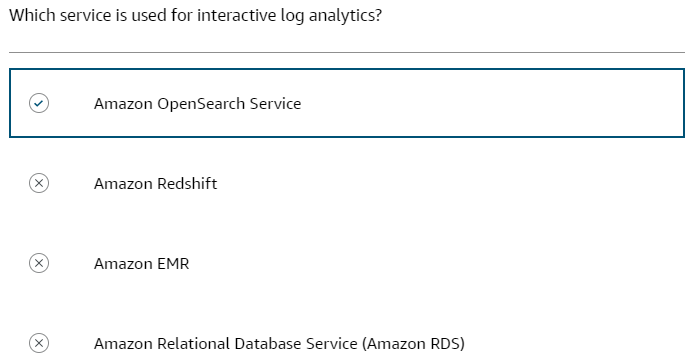


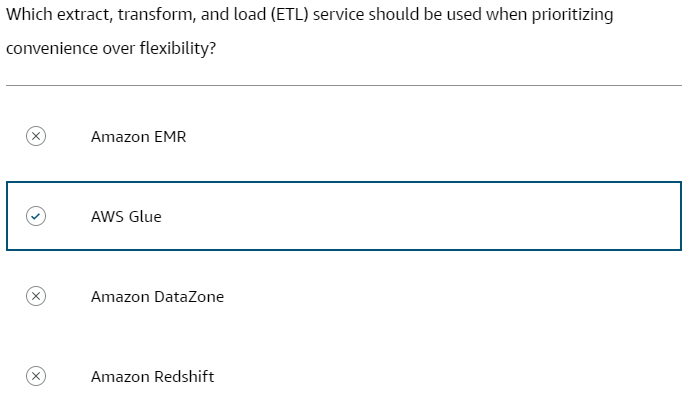


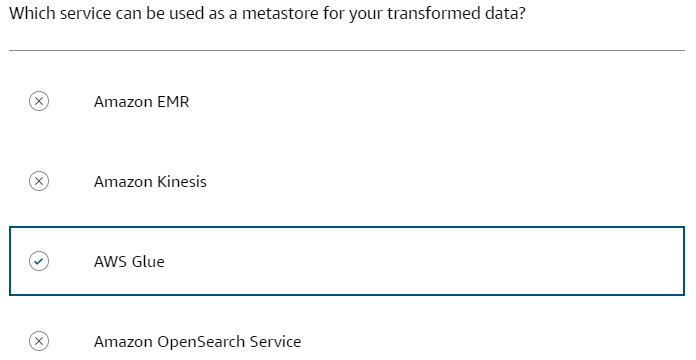












# **4. Architectures**

## **4.1. Introduction to Data Lakes**

### **What is a data lake?**

A data lake is a centralized repository that an organization can use to store data at scale, both structured and unstructured, in its original format. Data lakes ingest raw data from various sources such as databases, sensors, social media, documents, and applications. Prior to analysis, the raw data might undergo transformations such as cleansing, normalization, aggregation, and more. The large-scale and schema-on-read capabilities of a data lake provide opportunities for organizations to gain insights through big data analytics, machine learning, and artificial intelligence (AI).

### **Benefits of data lakes**

Data lakes on AWS help you break down data silos to maximize end-to-end value of your data.

**Scalability**

* Store massive amounts of data
* Scale storage up or down as needed.
* Store multiple types of data

**Cost-efficiency**

* Reduce costs with low-cost storage like Amazon Simple Storage Service (Amazon S3) or EMR File System (EMRFS)
* Develop schemas over time, which reduces modelling costs.
* Pay only for the storage used compared to the allocated capacity.

**Flexibility**

* Store structured, unstructured, and semi-structured data.
* Apply schemas later as needed for analysis.
* Conveniently load new data sources and types

**Faster Analysis**

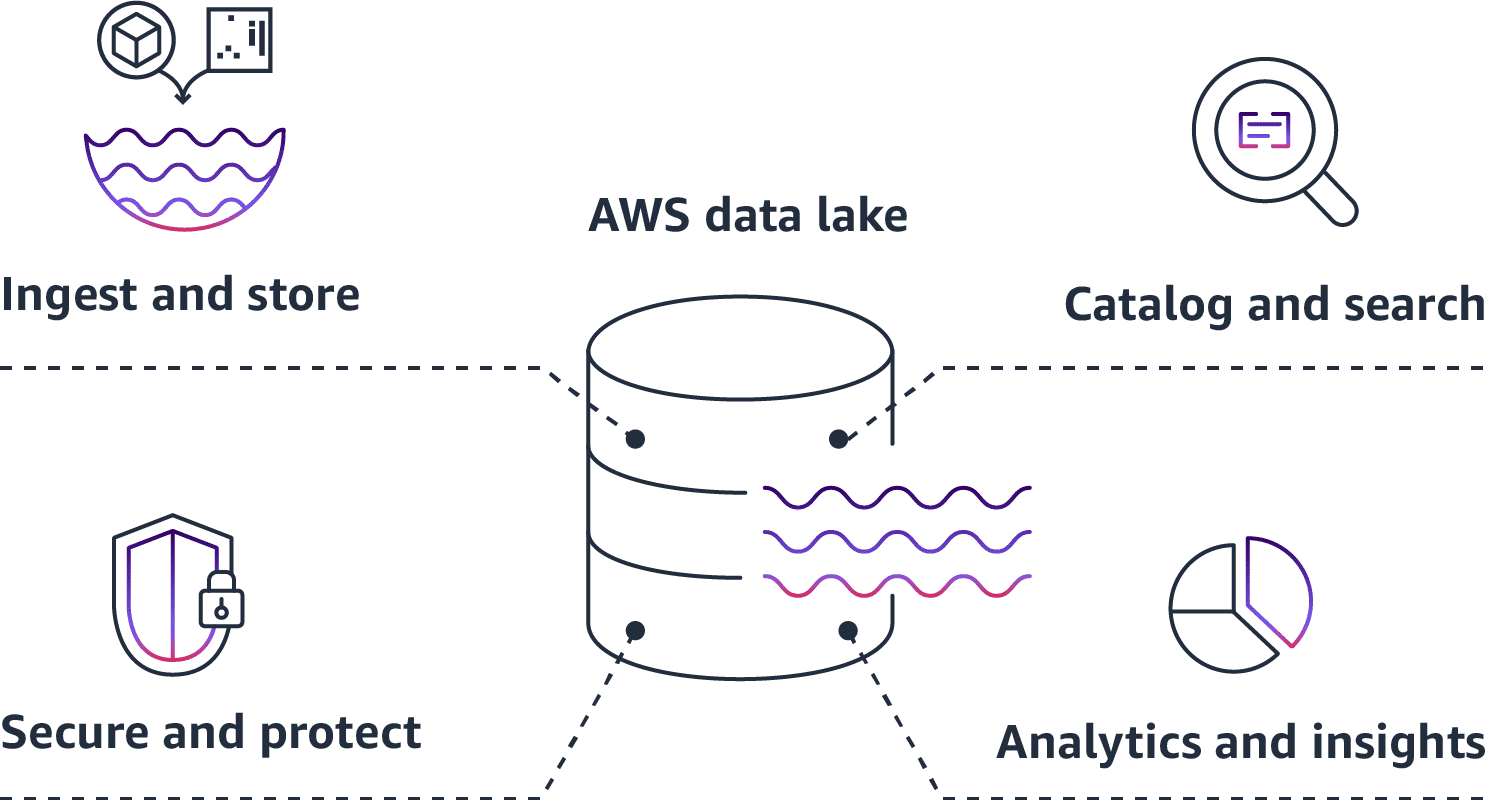
* Avoid the extract-transform-load (ETL) and extract-load-transform (ELT) processes, run and queries in place for impromptu analysis.
* Perform real-time and batch processing of data.

**Centralized View**

* View your data repository in a single view.
* Refer to a single source of truth instead of disparate systems.
* Apply common governance and security policies.

### **Data lake functions**

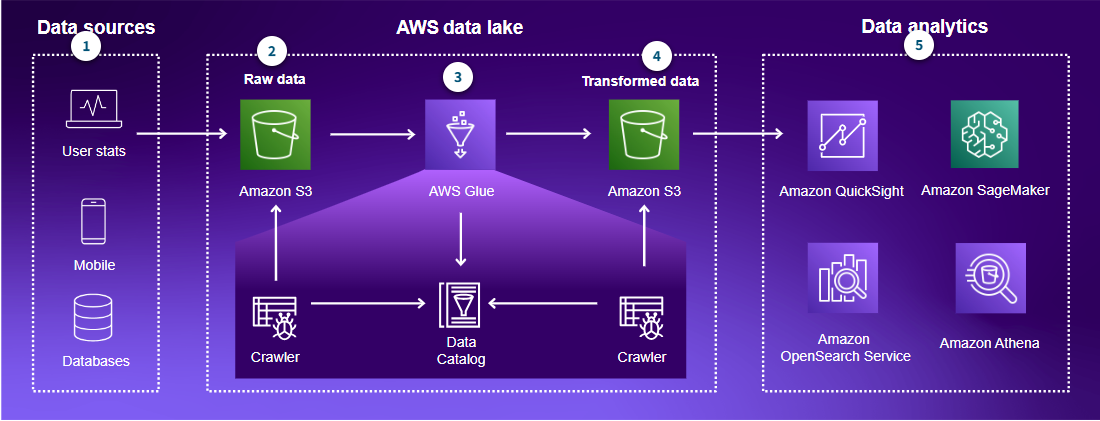
Data lakes are comprised of four main functions: data ingestion, cataloguing, securing, and insights.



* 1. **Ingest and Store** - Data is collected from multiple sources and ingested into the raw zone of the data lake in its original format. With this process, you can scale to data of any size and save time defining data structures, schema, and transformations upfront. The raw data can then be moved to a processed zone where it undergoes transformations to prepare it for analysis.
  2. **Catalog and search** - Data lakes can store different types of data, including unstructured, semi-structured, and structured data. A key advantage of data lakes is that they give you the ability to understand what data resides in the lake. Features like crawling, cataloging, and indexing help classify the contents of the data lake and make it easier to locate specific data assets for analysis. By supporting multiple data types and cataloging data, data lakes provide a centralized repository for the diverse data generated in modern organizations.
  3. **Secure and protect** - An AWS data lake secures and protects data through access controls, encryption, and auditing. Access controls restrict users and systems to only the data they need. Data is encrypted both at rest and in transit. It logs activity for access auditing.
  4. **Analytics and insights** - With data lakes on AWS, different roles in your organization can access data for analysis. Data scientists, developers, and analysts can use AWS analytics services like Amazon Athena, Amazon EMR, and Amazon Redshift. With these tools, you can run SQL, machine learning, and other analytics on data in Amazon S3 without moving it. AWS analytics services integrate with business intelligence tools. Users can build dashboards and visualizations using their preferred tools.

### **Basic data lake architecture**

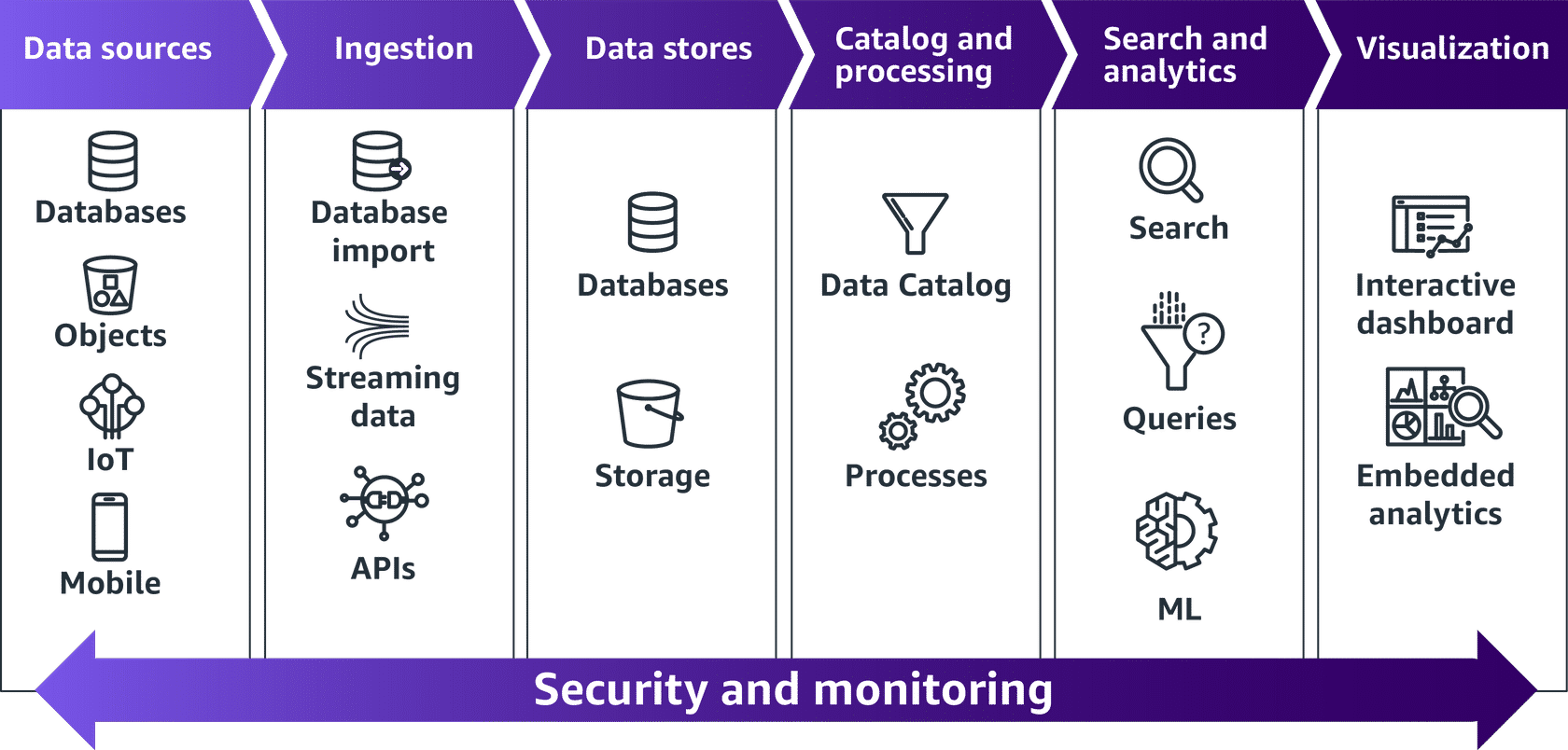
An AWS data lake architecture generally consists of Amazon S3 as the primary data store, where data from various sources is collected and stored in its original format. You can use AWS Glue crawl, catalog, and extract schema from the diverse data. Use additional AWS services like AWS Lambda, Amazon Kinesis, and Amazon EMR to process, transform, and analyze the data in the data lake. By using this architecture, you will have a central repository for collecting and organizing large amounts of data. That data could be structured, semi-structured, and unstructured from various sources for analytics and machine learning.



* 1. **Data sources** - Structured, semi-structured, and unstructured data can be ingested from many sources.
  2. **Raw data** - Raw data is ingested into the data lake and is stored in Amazon S3. Ingested data is cataloged using AWS Glue crawler. Amazon S3 is a cost- effective storage solution, but you can use Amazon Redshift and other Amazon databases in more advanced architectures.
  3. **AWS Glue** - AWS Glue ETL jobs are involved in transforming the data for a consuming use case.
  4. **Transformed data** - The transformed data zone in a data lake contains processed and enriched data that has been transformed from its raw form for analysis and reporting purposes. This zone holds consumable, trusted data assets that have been cleansed, validated, and structured in preparation for serving to downstream data consumers.
  5. **Data analytics** - Athena is representative of consuming the data from the data lake. Additional analytics, visualization, and machine learning tools can be integrated with the data lake, such as Amazon QuickSight for visualization or Amazon SageMaker for machine learning. Amazon OpenSearch Service can be integrated to develop log analytics and search solutions.

##### **AWS services used in the analytics pipeline.**

To make data stored in a data lake more useful, data lakes need governance, cataloging, and access controls. Following pipeline diagram shows which types of services are in each stage.



There are six stages in the analytics pipeline.

* 1. **Data sources stage:** Databases, object storage, IoT streams, mobile data.
  2. **Ingestion stage:** Database import, streaming data ingestion, API controls
  3. **Data stores:** Databases, various storage types
  4. **Cataloging and processing stage:** Data catalog, ETL processes
  5. **Search and analytics stage:** Search, queries, machine learning
  6. **Visualization stage:** Interactive dashboards, embedded analytics

Notice that security and monitoring are active throughout the entire pipeline.  Building a data lake can be challenging because many tools and services work together at different stages of the analytics pipeline. At every stage, you must secure and monitor your data.

### **Challenges in building a data lake.**

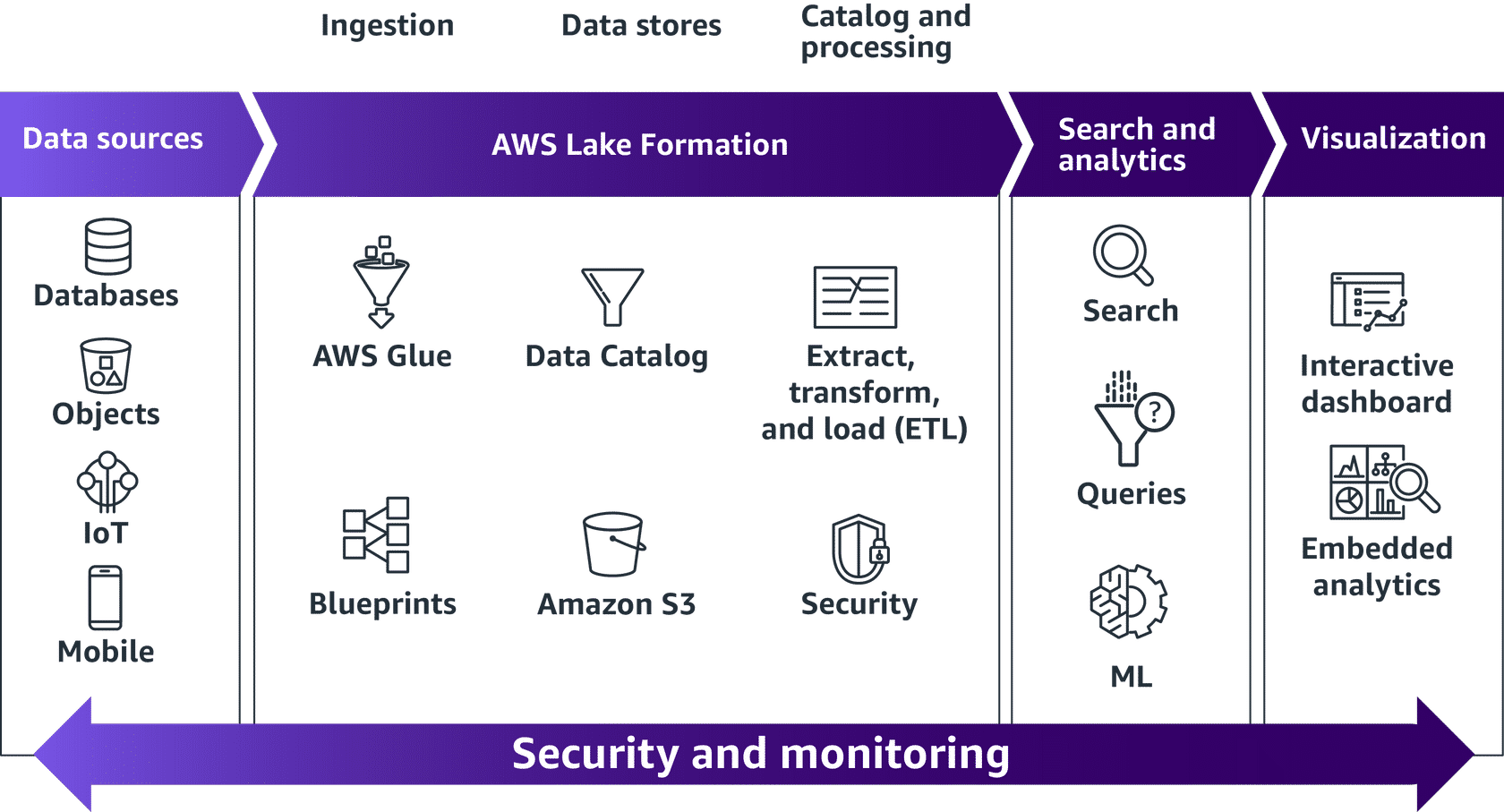
A major challenge with building a data lake is implementing effective data governance practices. As data from disparate sources flows into the lake, metadata can be lost or incorrect. With diverse users across an organization querying the data, it can be difficult to maintain data quality, security, and compliance without strong governance. With a robust governance framework, you can establish policies, roles, processes, and tools to properly ingest, catalog, protect, and manage the use of the varied data. Critical governance capabilities include lineage tracking, access controls, encryption, monitoring, auditing, and metadata management.

* 1. **Data governance** - Managing and governing huge volumes of disparate data from multiple sources is difficult. Establishing policies, access controls, data lineage tracking, and metadata management is critical but challenging.
  2. **Data quality** - Ingesting raw data from multiple sources leads to issues with data quality, inconsistencies, duplications, and integrity. Performing ETL and cleaning for huge data volumes is difficult.
  3. **Security** - Securing a data lake with large amounts of heterogeneous data is challenging. Implementing access controls, encryption, masking, and auditing at scale needs thoughtful planning.

### **Introduction to Lake Formation**

Lake Formation is a managed service that simplifies building, securing, and managing data lakes. It automates ingesting, cataloging, cleaning, and transforming data from diverse sources into a data lake on Amazon S3. Lake Formation uses machine learning and policies to automatically secure, organize, and catalog data. It provides a governed data lake by integrating with AWS services like AWS Glue, Athena, AWS Identity and Access Management (IAM), and AWS CloudTrail.

Lake Formation provides features for three stages in data lifecycle: ingestion, data storage, and catalog & processing. Below diagram shows the components of Lake Formation.



**Key benefits**

* Fully managed serverless service.
* Data lake developed in days, instead of weeks or months.
* Streamlined permission management.
* Monitored and audited access to verify compliance.
* Data sharing
* Integration with many analytics and machine learning tools

**Key features**

* **Automated build environment** – Lake Formation automates many of the complex manual steps required to create data lakes. These steps include collecting, cleansing, moving, and cataloging data, and securely making that data available for analytics and machine learning.
* **Persistently store dataset metadata** – The AWS Glue Data Catalog within the data lake persistently stores the metadata from raw and processed datasets as they are identified by AWS crawlers.
* **Orchestrate scripts and crawlers** – AWS Glue jobs encapsulate scripts, such as ETL scripts, which connect to source data, process it, and write it out to a data target. AWS Glue triggers can start jobs based on a schedule, event, or on demand. AWS Glue workflows orchestrate ETL jobs, crawlers, and triggers.
* **Centralized access controls** – Lake Formation provides centralized access controls for your data lake, including security policy-based rules for users and applications by role. Lake Formation uses the encryption capabilities of Amazon S3 for data in the data lake.

**References** - [AWS Lake Formation](https://aws.amazon.com/lake-formation/).

### **Lake Formation roles**

The Lake Formation permissions model includes three security roles.

1. **Lake Formation administrator**

* Has full read access to resources.
* Has data location permissions.
* Can grant or revoke access to resources, including self.
* Can create databases.
* Can grant permission to create databases.

1. **Database creator**

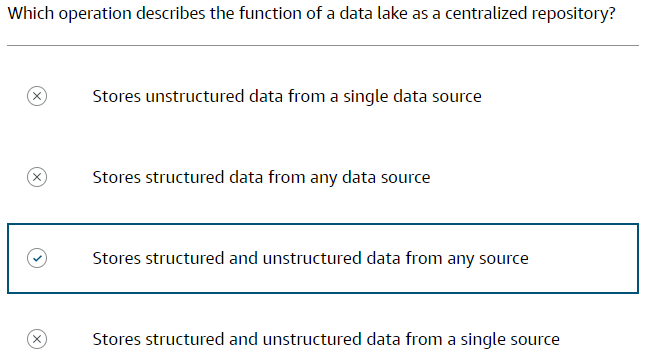
* Has all database permissions on databases that they create.
* Has permissions on tables that they create.
* Can use console or API to designate database creators.

1. **Table creator**

* Has permissions on tables that they create.
* Can grant permissions on tables that they create.
* Can view databases containing the tables that they create.

**NOTE:** Lake Formation permissions still require users to have appropriate IAM permissions.

### **Knowledge Check**



## **4.2. Introduction to Data Warehousing**

### **What is a data warehouse?**

A data warehouse is a central repository of information that is specially designed for analytics. Data flows into a data warehouse from business applications, databases, and other sources. Data warehouse users access the data through business intelligence (BI) tools, SQL clients, and other analytics applications. A data warehouse powers reports, dashboards, and analytics tools by storing data efficiently. It minimizes the input and output (I/O) of data and delivers query results quickly to hundreds and thousands of users concurrently.

### **Challenges with on-premises data warehouses**

Organizations often begin implementing a data warehouse on-premises and run into many challenges.

**High upfront costs** –On-premises data warehouses require large upfront investments in hardware, software licenses, and real estate among other expenses. This results in high fixed costs.

**Scalability issues** –It can be challenging to scale up or scale down an on-premises data warehouse because workload requirements change. Additional infrastructure and licenses might be needed to handle growth.

**Maintenance overhead** –On-premises data warehouses require significant resources to maintain. This includes administrators, database administrators (DBAs), infrastructure management, and others. This adds to ongoing operational expenses.

**Limited flexibility** –

On-premises data warehouses provide limited flexibility in expanding or adopting new technologies because they involve physical infrastructure. Moving or upgrading warehouses can be complex.

**Skill set requirements** –On-premises data warehouses require specialized skill sets such as database administration and system engineering. Recruiting and retaining such talent can be difficult.

**Lack of availability** –On-premises data warehouses are prone to downtime because of operational issues, outages, or disasters. Achieving high availability adds further complexity.

In summary, on-premises data warehouses involve major investment, ongoing maintenance costs, and inflexibility in scaling.

### **Data warehouse modernization using Amazon Redshift**

Amazon Redshift can help organizations set up and deploy a new data warehouse in minutes. It’s built to store and query datasets ranging from gigabytes to petabytes in size. Amazon Redshift provides a cloud-based, scalable, secure environment for your data warehouse. Many organizations deploy to Amazon Redshift to improve their query performance, reduce overhead, and lower the cost of analytics.

**Scalability** –With Amazon Redshift, organizations can scale compute and storage separately. Therefore, they can add resources on-demand to handle spikes in workloads. Amazon Redshift can query exabytes of data with fast performance.

**Agility** –Amazon Redshift rapidly spins up new data warehouses for new workloads. This provides flexibility to experiment and innovate quickly. Amazon Redshift integrates well with AWS data and analytics services.

**Cost-efficiency** –With Amazon Redshift, organizations only pay for the resources provisioned. It is very cost-effective compared to on-premises data warehouses. Automated monitoring tools optimize utilization to reduce costs.

**Performance** –Amazon Redshift provides fast query performance through columnar storage, running parallel queries, and advanced caching. Performance tuning features like workload management further boost throughput.

**Durability** –Amazon Redshift replicates data across multiple nodes to protect against failures. Data is stored redundantly across multiple availability zones for high durability.

**Security** –Amazon Redshift provides enterprise-grade security features like SSL encrypted connections, database encryption, virtual private clouds (VPCs), IAM policies, and audit logging. This helps meet strict regulatory requirements.

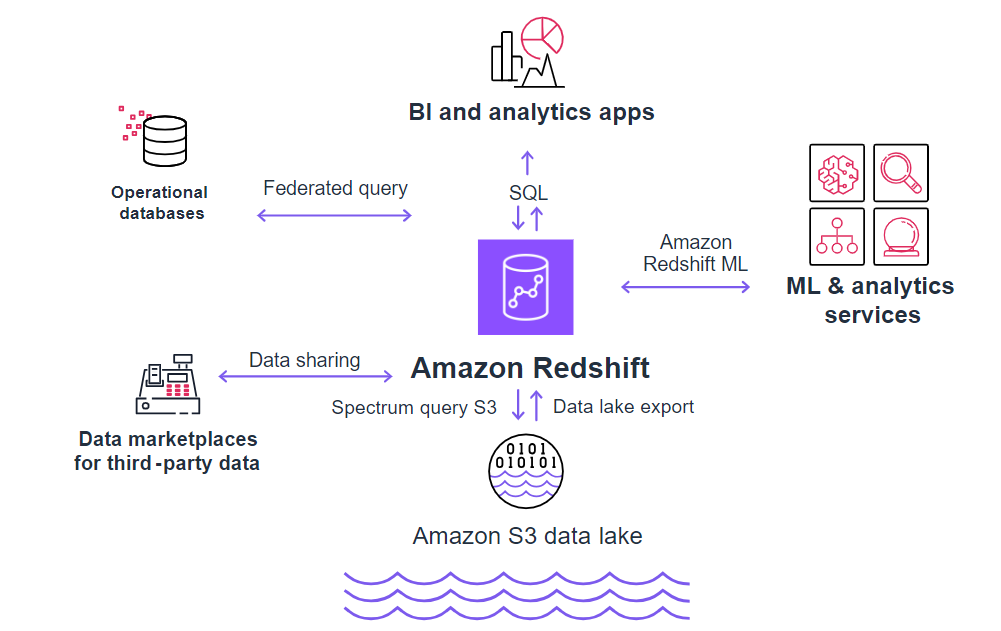
**Serverless** –Amazon Redshift Serverless makes it easy to run analytics workloads of any size without having to manage data warehouse infrastructure. With Amazon Redshift, you pay for compute and storage separately. Serverless automates capacity management by making the rightsizing decisions and scales the capacity elastically and automatically based on the workloads. For data, you pay for data stored in the managed storage of Amazon Redshift at the existing fixed gigabit-per-month rate.

**Machine Learning** –Amazon Redshift makes building and deploying machine learning (ML) models remarkably easy. Through simple SQL commands, users can leverage built-in algorithms like random forest and logistic regression to explore data, engineer features, and train models with no coding required. Amazon Redshift integrates seamlessly with SageMaker to import external models directly into Amazon Redshift. With Amazon Redshift, users of all skill levels can harness the power of machine learning across petabytes of data by abstracting away the complexity. With Amazon Redshift ML, machine learning is made accessible and easy to apply within your data warehouse.

**Automation** –Amazon Redshift self-regulates its data warehouse performance using ML. Amazon Redshift optimizes and automates administrative tasks like distributing physical data, designing the storage layer schema, and maintaining tables.

**Data sharing** –With Amazon Redshift data sharing, organizations can securely share live read-only snapshots of Amazon Redshift data lakes with other accounts or within their business for real-time analytics and reporting. Amazon Redshift applies continuous delta updates to keep shared snapshots up to date while encrypting data in transit and at rest for security. Data consumers can directly query the live snapshots using SQL without moving any data. Amazon Redshift data sharing minimizes data duplication by sharing source datasets while still providing centralized control through integrated permissions and audit logging.

This diagram shows Amazon Redshift as a key component in a modern data architecture. You will learn more about modern data architecture in the next lesson. Amazon Redshift can easily query data across all your data stores. Amazon Redshift can query an Amazon S3 data lake and write back data to the data lake in open formats. Amazon Redshift helps you to make live data queries in operational databases without requiring any data loading and zero-ETL. Amazon Redshift uses familiar SQL statements to combine and process data across all your data stores. An Amazon S3 data lake makes data available to other analytics and machine learning tools rather than locking it in a new silo.



### **Zero-ETL**

Amazon Redshift has built-in support for zero-ETL. Zero-ETL is a set of integrations that eliminates or minimizes the need to build ETL data pipelines. Typical ETL processes are time-consuming and complex to develop, maintain, and scale. Traditionally, moving data from a transactional database into a central data warehouse always required a complex ETL solution. Zero-ETL facilitates point-to-point data movement without the need to create ETL data pipelines. Zero-ETL instantly duplicates data from the transactional database to the data warehouse. The duplication mechanism uses change data capture (CDC) techniques built into the data warehouse. Applications store data in the transactional database and analysts query the data from the warehouse seamlessly. Zero-ETL helps with querying across data silos without the need for data movement. There is no requirement to stage the streaming data for transformation on any other storage service.

**Auto-copy from Amazon S3** –This feature automates data ingestion into Amazon Redshift. This feature continuously ingests data as soon as new objects are created in Amazon S3 with no custom coding or manual ingestion activities.

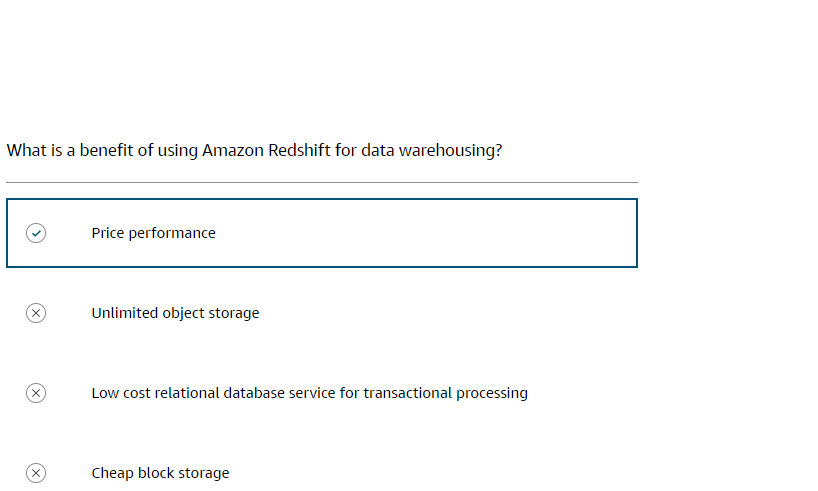
**Streaming ingestion** –The Amazon Redshift streaming ingestion feature ingests hundreds of megabytes of data per second from Amazon Kinesis Data Streams or Amazon Managed Streaming for Apache Kafka (Amazon MSK). You can define a schema or choose to ingest semi-structured data with the SUPER data type to query data in real time.

**Aurora zero-ETL with Amazon Redshift** –Amazon Aurora now supports zero-ETL integration with Amazon Redshift to enable near real-time analytics and ML using Amazon Redshift on petabytes of transactional data from Aurora. Within seconds of transactional data being written into Aurora, the data is available in Amazon Redshift.With this zero-ETL integration, you can analyze data from multiple Aurora database clusters in the same new or existing Amazon Redshift instance.

With near real-time access to transactional data, you can use capabilities of Amazon Redshift to derive insights from transactional and other data:

* Built-in ML
* Materialized views
* Data sharing
* Federated access to multiple data stores and data lakes

### **Knowledge Check**

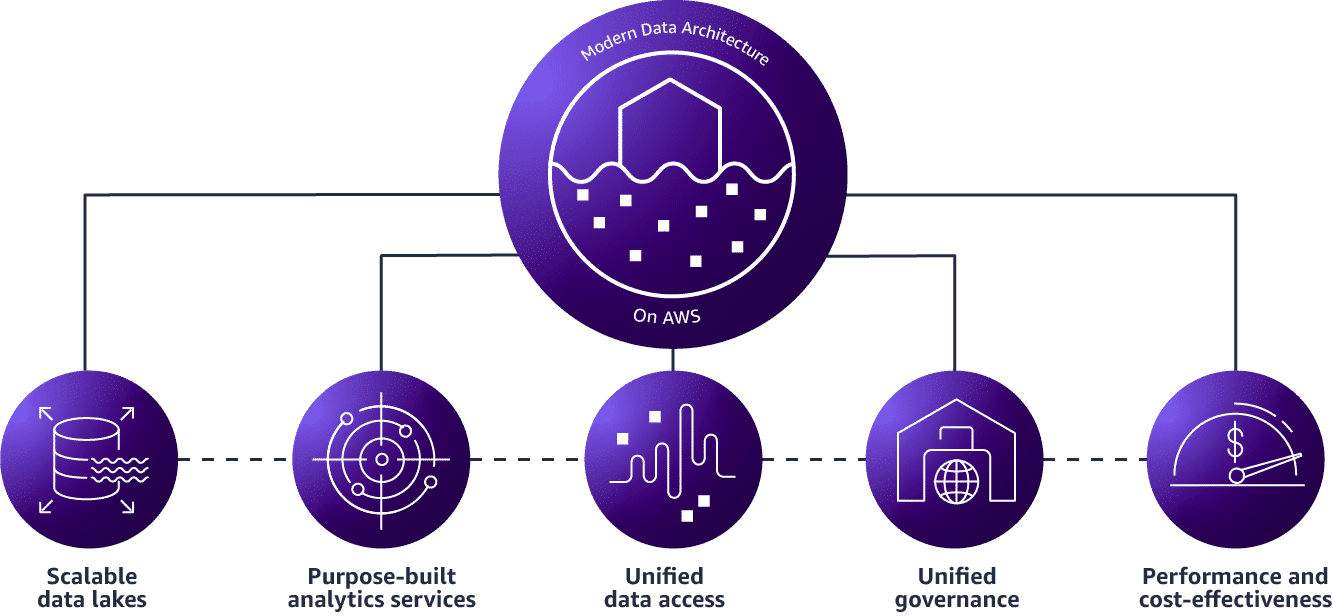


## **4.3. Introduction to modern data architecture**

A modern data architecture removes the boundaries between disparate systems and seamlessly integrates the data lake, data warehouse, and purpose-built data stores. Rather than a one-size-fits-all approach that compromises the analytics platform, a modern architecture recognizes different systems are optimal for different data needs. True integration goes beyond connecting data systems. It also encompasses unified data governance, security, metadata management, and easy movement of data between storage and processing. The flexible architecture can adapt to changing requirements and help you use the best technology for varying data infrastructure needs.

### **Pillars of a modern data architecture**

Modern data architecture aims to provide data insights, analytics, reporting, machine learning, and AI by efficiently processing large volumes of diverse data from multiple sources. It uses cloud infrastructure and modern big data tools like data lakes, stream processing, and cloud warehouses to ingest, store, process, and analyze data at scale. It also optimizes for speed, flexibility, and cost efficiency.



* 1. **Scalable data lakes** - A scalable data lake can handle growing amounts of data, users, and workloads. As volumes increase, a scalable data lake can expand its storage and processing capabilities without significant redesign or disruption. With a scalable storage solution, such as a file system like Amazon S3, organizations can store petabytes of data.
  2. **Purpose-built analytics services** - Although data lakes support queries and exploratory analytics, there are use cases that require high volume and high-performing analytics. Purpose-built data stores provide specialized tools for each job and are an integral component of modern data architecture. They optimize performance, scalability, and usability for specific use cases.
  3. **Unified data access** - As the data in your data lakes and purpose-built data stores continues to grow, you might need to be able to move a portion of that data from one data store to another. Unified data access is direct access to data where it resides and optimizes selective data movement as needed.
  4. **Unified governance** - One integral piece of a modern analytics architecture is the ability for customers to authorize, manage, and audit access to data. Unified governance provides the ability to manage access to all data across your data lakes and purpose-built data stores from a single place.
  5. **Performance and cost-effectiveness** - Analytics services should provide high performance at the lowest cost.

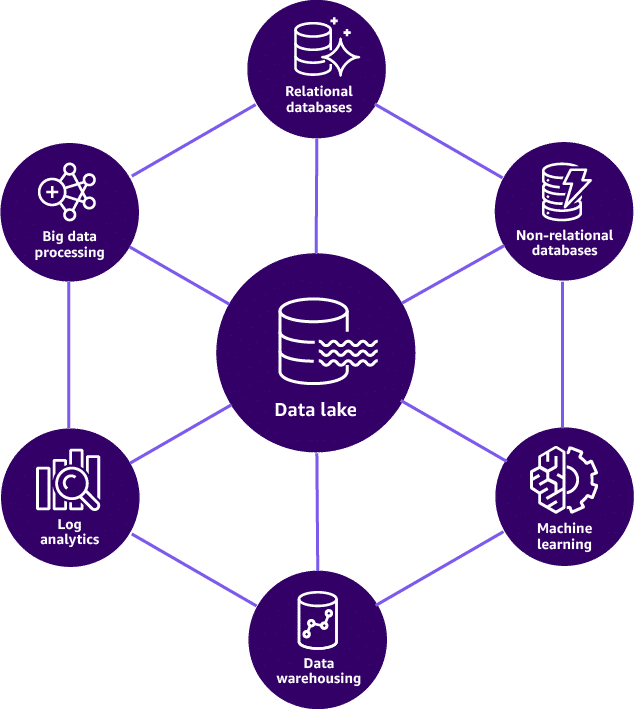
### **Concepts of data movement**

Efficiently managing and analyzing data at scale requires thoughtful architecture strategies to avoid unnecessary data movement that can incur substantial costs and latency.

Key concepts include the following:

* Minimizing transit
* Using data gravity
* Using zero-ETL or minimal extract-load-transform (ELT) workflows
* Orchestrating pipelines
* Streaming for continuous data processing

By architecting application and analytics workloads on the cloud using these best practices, we can optimize data movement to build responsive, cost-effective data platforms.



Modern data architecture components are made up of relational databases, non-relational databases, machine learning, log analytics, data warehousing, and big data processing. Common to all the data movement types is the data lake, in which all modern data architecture components store data and write data to.

**Types of data movement:**

* 1. **INSIDE-OUT**

Inside-out data movement keeps data in a central repository like a data lake and moves compute to it. Key enablers are technologies like serverless functions, query engines, and orchestration tools that collect processing workloads with the source data. A shared data catalog provides metadata to enable querying in place. By analyzing, transforming, and moving data only, when necessary, this pattern reduces duplication and movement costs while optimizing processing performance.

* 1. **OUTSIDE-IN**

Outside-in data movement refers to the process of bringing data from purpose-built systems into an organization's data lake. An example is when a customer copies query results for regional product sales from their data warehouse into their data lake to run product recommendation algorithms against a larger dataset using ML.

* 1. **AROUND THE PERIMETER**

Around the perimeter data movement is moving data from one purpose-built data store to another. An example is when a customer copies the product catalog data stored in their database to their search service. This makes it easier to look through their product catalog and offload the search queries from the database.

* 1. **SHARING ACROSS**

Sharing across data movement involves the seamless transfer of data between different applications. An example of sharing across data movement is using a data mesh architecture. Data as a product is a central design goal of data mesh. A business-aligned domain registers as a node in a mesh, publishes their data products to a central governance catalog, and discovers data products to consume through central governance services. The enterprise manages a central set of governance services to support enterprise-wide data discovery, reporting, and auditing.

* **Data mesh:** A data mesh architecture consists of data teams that build and run the platform. They build security controls, run the onboarding, and provide training.
* **Data producers:** Data producers consist of business domain teams that want to share their data. They have domain expertise and manage data ownership and governance. They ensure data quality and manage the metadata to make it easy to find their data.
* **Data consumers:** Data consumersmake use of the data mesh to fit their business function. They want to be able to find data easily. Data consumers use data to run business priorities and develop business analytics to find new insights.

### **Zero-ETL for zero data movement**

Traditional ETL processes are difficult and take a long time to develop, maintain, and scale. Instead, zero-ETL integrations facilitate point-to-point data movement without the need to create ETL data pipelines. Zero-ETL can also enable querying across data silos without the need for data movement.

**Reference** - [What is Zero-ETL?](https://aws.amazon.com/what-is/zero-etl)

### **Data mesh architecture for data sharing**

Organizations need to share data with consumers more easily and securely. The data mesh architecture can solve these organizational needs.

A data mesh is a decentralized data architecture and governance model that treats data as a product. A data mesh architecture consists of self-serve data platforms provisioned by domain or business unit teams. These autonomous data platforms provide domain-specific analytics, applications, and machine learning, while infrastructure teams manage shared data utilities like catalogs, observability, and platform services. Data is managed as products with an emphasis on discoverability, quality, and collaboration between domains and central IT teams. The domain-oriented decentralization aims to increase organizational agility and analytical speed while maintaining enough consistency and governance through core infrastructure.

A data mesh pattern separates consumers, producers, and central governance. Data products are stored in their own data lake. The data mesh provides physical separation. Each data lake is cataloged and schematized. Services that consume data are in a consumer application domain and are separate from each other and the data lakes. When data consumers need data from one or more data lakes, cloud services make the data visible to consumers and query-able directly from the data lakes. Data product-specific lakes that hold data and the application domains that consume lake data are interconnected to form the data mesh.

**Goals of a data mesh:**

* 1. **DATA PRODUCTS**

A data product in a data mesh is a curated set of data provided by a domain team to serve specific business needs. It encapsulates raw data with value-added content like cleaned data schemas, transformation code, and quality metrics to power analytics and applications. Data products provide discoverability and reusability of data across the organization.

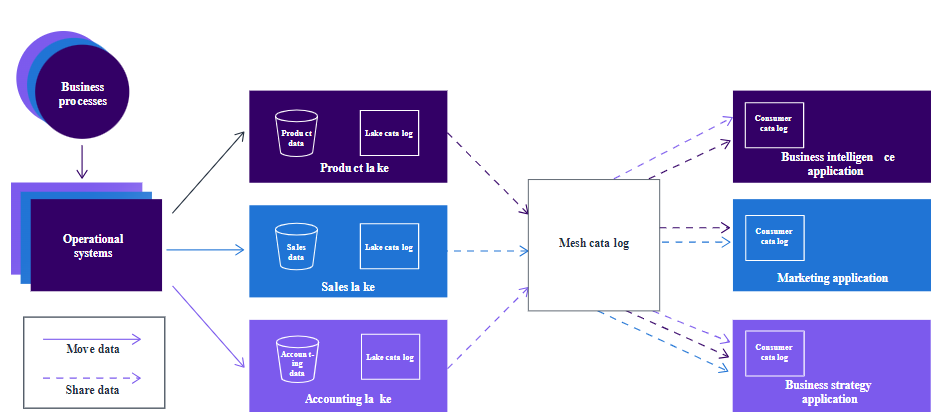
* 1. **CENTRAL DATA GOVERNANCE**

Central data governance in a data mesh on AWS provides common infrastructure and policies while empowering domain teams to build data products. It manages core data services like catalogs, pipelines, and data quality tools. Domain teams are accountable for their data products within the overall governance standards set by central IT.

* 1. **COMMON ACCESS**

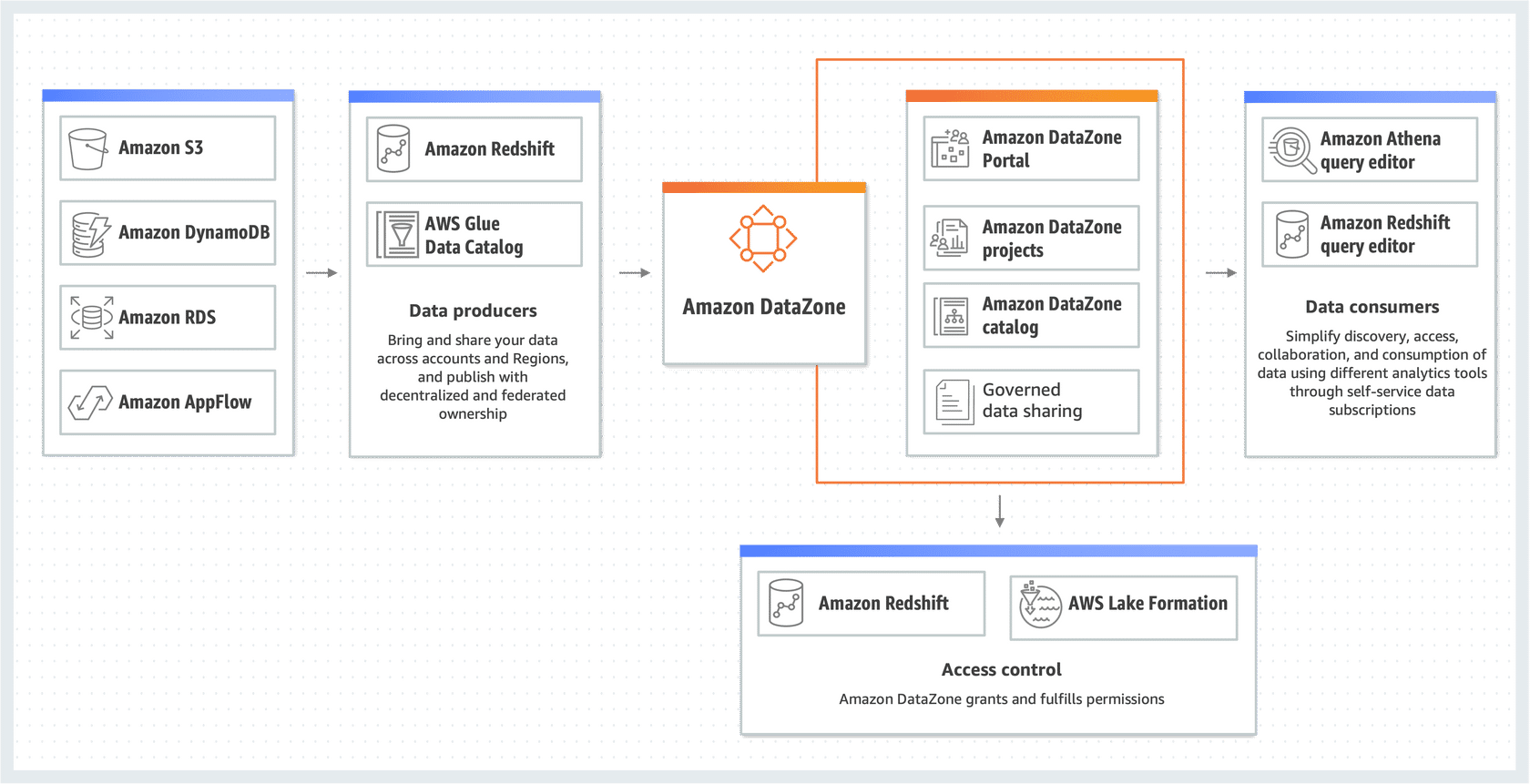
Common access provides consistent discovery and consumption of decentralized data products across domains. Standard tooling like catalogs and data portals provide unified access. Usage monitoring, access controls, and infrastructure are centrally governed however schema, APIs, and pipelines create a common interface to domain data products.

The following diagram illustrates a data mesh architecture.



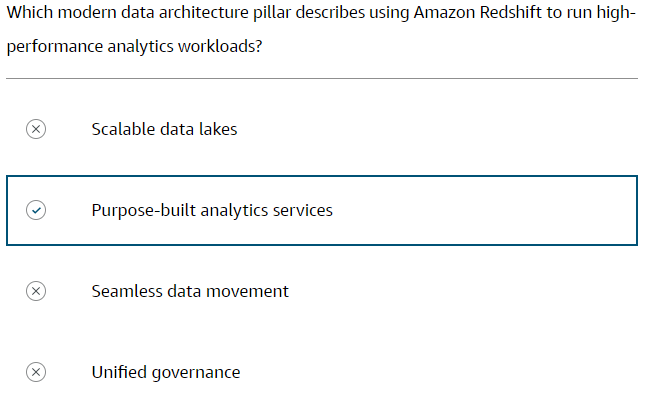
### **Amazon DataZone**

AWS implements the data mesh pattern through Amazon DataZone. This service helps organizations manage, govern, and share data assets across AWS, on-premises sources, and third-party sources. It provides fine-grained access controls to ensure the right users can access the right data. Amazon DataZone also helps organizations discover, catalog, and collaborate around data to drive insights.  
Amazon DataZone integrates with various AWS services. It can publish data assets from sources like AWS Glue Data Catalog, Amazon Redshift, and Amazon S3 into the Amazon DataZone catalog. Amazon DataZone supports querying data through Athena and Amazon Redshift. It also uses Lake Formation and Amazon EventBridge to control access to data assets and integrate with other services.



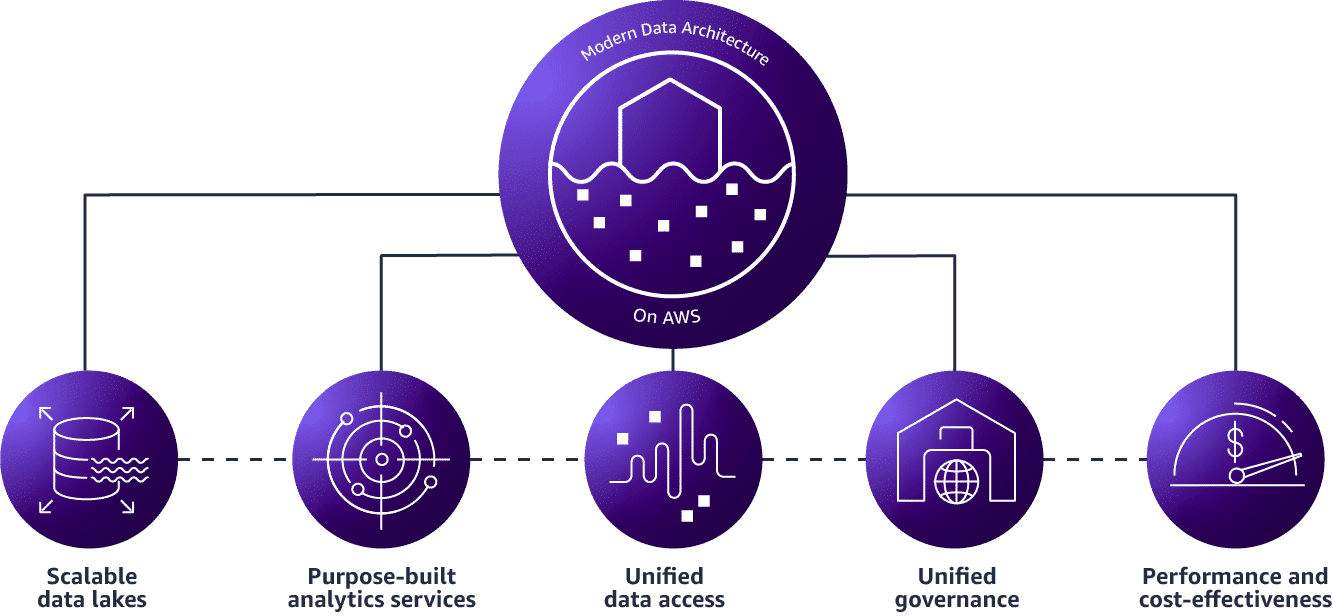
**Reference** - [Amazon DataZone](https://aws.amazon.com/datazone/).

### **Knowledge Check**



## **4.4. AWS Services for Modern Data Architecture**

### **AWS services for pillars of a modern data architecture**



On AWS, these features integrate through managed services, separating concerns while providing scalability and flexibility. The cloud-first architecture abstracts infrastructure complexities while automating data pipelines. With AWS services, you can quickly build modern data architectures that deliver analytics and insights.

### **AWS services for scalable data lakes**

A modern data architecture uses scalable data lakes that can handle growing amounts of data and workloads. Data lakes can expand effortlessly to store petabytes of data.

##### **Amazon S3**

Amazon S3 is a scalable, high-speed, web-based cloud storage service designed for online backup and archiving of data and applications. You can store large amounts of data in Amazon S3, and it can handle any amount of data.

**Reference** - [Amazon S3](https://aws.amazon.com/s3/).

##### **AWS Lake Formation**

Lake Formation is used to easily build a secure and scalable data lake. Lake Formation helps you centrally secure and manage your data lakes in the cloud.

**Reference** - [AWS Lake Formation](https://aws.amazon.com/lake-formation/).

##### **AWS Glue Data Catalog**

This is a centralized metadata catalog for all data assets across various data sources. Quickly discover and search multiple AWS datasets without moving the data.

**Reference** - [AWS Glue Features](https://aws.amazon.com/glue/features/).

### **AWS services for purpose-built analytics**

AWS offers purpose-built analytics services that provide capabilities for processing, exploring, visualizing, and building predictions from data as part of a modern cloud-based analytics architecture. These services are all designed to help accomplish complex analytics goals.

##### **Amazon Managed Service for Apache Flink**

Query and analyze streaming data using Apache Flink. This managed service helps you get started with Apache Flink to quickly build and run real-time stream processing applications.

**Reference** - [Amazon Managed Service for Apache Flink](https://aws.amazon.com/managed-service-apache-flink/).

1. **Amazon QuickSight**

QuickSight is a business analytics service that makes it easy to build visualizations, perform targeted analysis, and provide business intelligence at any time on any device. Business analysts can use the Athena or Amazon Redshift interactive SQL interface to power QuickSight dashboards with data in modern data architecture storage.

**Reference** - [Amazon QuickSight](https://aws.amazon.com/quicksight/).

1. **Amazon OpenSearch Service**

OpenSearch Service is a fully managed service you can use to deploy, operate, and scale OpenSearch, a popular open-source search and analytics engine. It offers easy-to-use APIs and real-time analytics capabilities to deliver fast search, visualizations, and dashboards for log analytics, application monitoring, and operational data workloads.

**Reference** - [Amazon OpenSearch Service](https://aws.amazon.com/opensearch-service/).

1. **Amazon Redshift**

Amazon Redshift is a cloud data warehouse. With federated queries, you can query and analyze data across operational databases, data warehouses, and data lakes. Amazon Redshift provides a powerful SQL capability designed for blazing-fast online analytical processing of very large datasets. Perform in-place querying of data in a data lake. You can also run Redshift SQL queries against massive amounts of data.

**Reference** - [Amazon Redshift](https://aws.amazon.com/redshift/).

1. **Amazon EMR**

Amazon EMR is a highly distributed computing framework used to process vast amounts of data using open-source tools. With Amazon EMR, you can run big data applications and petabyte-scale data analytics faster and at less than half the cost of on-premises solutions.

**Reference** - [Amazon EMR](https://aws.amazon.com/emr/).

1. **Amazon SageMaker**

A fully managed ML service, SageMaker provides features to build, train, and deploy ML models. Data scientists can connect SageMaker to the Lake House storage layer and access training feature sets.

**Reference** - [Amazon SageMaker](https://aws.amazon.com/sagemaker/).

1. **AWS services for AI**

AWS has deep expertise in AI and provides the most comprehensive set of services, tools, and resources for customers. AWS offers many AI services to build and scale your AI applications and for your use cases.

**Reference** - [Artificial Intelligence](https://aws.amazon.com/ai)

The following are some examples of AI services.

* **Amazon CodeGuru Security**

Use this service to detect, monitor, and fix code security vulnerabilities.

**Reference** - [Amazon CodeGuru Security](https://aws.amazon.com/codeguru/).

* **Amazon Fraud Detector**

Use this service to detect online fraud and enhance detection models.

**Reference** - [Amazon Fraud Detector](https://aws.amazon.com/fraud-detector)

* **Amazon Rekognition**

Use this service to automate, streamline, and scale image recognition and video analysis.

**Reference** - [Amazon Rekognition](https://aws.amazon.com/rekognition/).

* **Amazon Sagemaker**

Use this service to build, train, and deploy ML models.

**Reference** - [Amazon SageMaker](https://aws.amazon.com/sagemaker/).

* **Amazon Textract**

Use this service to extract printed text, analyze handwriting, and automatically capture data **Reference** - [Amazon Textract](https://aws.amazon.com/textract).

* **Amazon Transcribe**

Use this service to convert speech to text, extract key business insights from video files, and improve business outcomes.

**Reference** - [Amazon Transcribe](https://aws.amazon.com/transcribe).

1. **AWS services for generative AI**

AWS makes it easy to build and scale generative AI with security, privacy, and responsible AI.

**Reference** - [Generative AI](https://aws.amazon.com/generative-ai/).

The following are examples of generative AI services.

* **Amazon Bedrock**

Amazon Bedrock is a fully managed service that offers a choice of high-performing foundation models (FMs) from leading AI companies with a single API. Amazon Bedrock is serverless, so you don't have to manage infrastructure. You can securely integrate and deploy generative AI capabilities into your applications using the AWS services you are already familiar with.

**Reference** - [Amazon Bedrock](https://aws.amazon.com/bedrock/).

* **Amazon CodeWhisperer**

Amazon CodeWhisperer generates code in real time in your integrated development environment (IDE) based on your comments and existing code. You can select from over 15 programming languages, your favourite IDEs, and your favourite command lines. It supports CLI completions and natural-language-to-bash translation in the command line. Generative AI-powered code suggestions help you increase productivity and enhance code security.

**Reference** - [Amazon CodeWhisperer](https://aws.amazon.com/codewhisperer/).

* **Amazon QuickSight Generative-BI**

Generative BI in QuickSight helps you build, discover, and share insights in seconds. Generative BI in QuickSight helps business teams interact with data using natural language. Business users can author dashboards quickly using natural language and generate data stories automatically.

**Reference** - [Amazon QuickSight Generative-BI](https://aws.amazon.com/quicksight/generative-bi)

1. **Amazon Athena**

Use Athena for interactive querying, analyzing, and processing petabytes of data where it resides. With Athena, you can analyze data directly in Amazon S3 using standard SQL.

**Reference** - [Amazon Athena](https://aws.amazon.com/athena/).

1. **Amazon RDS**

Amazon RDS is a managed relational database service that provides resizable capacity while automating time-consuming administration tasks like hardware provisioning, database setup, patching, and backups. It offers high availability and failover support for popular database engines like PostgreSQL, MySQL, MariaDB, Oracle, and SQL Server.

**Reference** - [Amazon RDS](https://aws.amazon.com/rds/)

1. **Amazon Aurora**

Amazon Aurora is serverless and scalable MySQL and PostgreSQL compatible OLTP database. It is high performance, highly available, and fully managed. It has many integrations with other AWS services to perform analytics and extract insights.  
**Reference** - [Amazon Aurora](https://aws.amazon.com/rds/aurora/)

1. **Amazon DynamoDB**

Amazon DynamoDB is a serverless NoSQL key-value database. It is fast, scalable, and fully managed. It is highly integrated with many AWS services and you can export table data to performance analytics and gain insights.

**Reference** - [Amazon DynamoDB](https://aws.amazon.com/dynamodb/)

### **AWS services for unified data access**

A core capability of modern data architecture is to combine, move, and replicate data across multiple data stores and your data lake.

##### **AWS Glue**

AWS Glue is a fully managed ETL service that automates the difficult, time-consuming steps of data preparation for analytics. AWS Glue crawls data sources, identifies data formats, suggests schemas and transformations, and generates ETL code to run your data processing and loading jobs.

When you’re connecting to data sources and destinations, handling datasets, and initiating common transforms, AWS Glue on Ray is a straightforward methodology for using Ray to solve problems transforming Ray datasets.

**Reference** - [AWS Glue](https://aws.amazon.com/glue/).

##### **Amazon Kinesis Data Firehose**

Kinesis Data Firehose is an ETL service that reliably prepares and loads streaming data into data lakes, data stores, and analytics services. You can stream data into Amazon S3 and convert data into required formats for analysis without building processing pipelines.

**Reference** - [Amazon Kinesis Data Firehose](https://aws.amazon.com/kinesis/data-firehose/).

### **AWS services for unified governance**

An important feature of modern data analytics architecture is to help customers authorize, manage, and audit access to data.

##### **Amazon DataZone**

Amazon DataZone helps organizations break down data silos and derive more value from their data. It provides purpose-built solutions to discover data across silos, catalog and build context around distributed data, apply machine learning to enrich data, and activate and share data at scale through fine-grained access controls.

**Reference** - [Amazon DataZone](https://aws.amazon.com/datazone/).

##### **AWS Lake Formation**

Use Lake Formation to centrally manage your security, access control, and audit trails. You can get fine-grained row-level security for your data.

**Reference** - [AWS Lake Formation](https://aws.amazon.com/lake-formation/).

### **Modern data architecture on AWS**

A modern data architecture on AWS solves analytics challenges, delivers analytics and insights, and is high performance and cost effective.

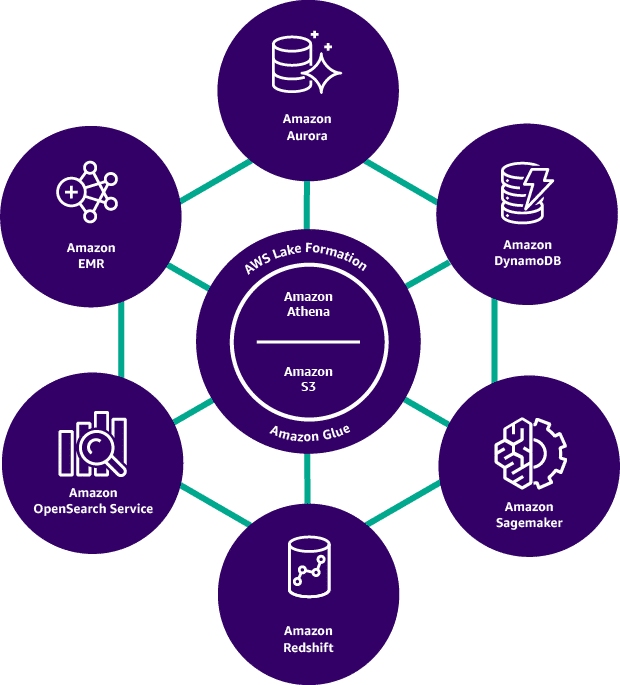
AWS services that support scalable data lakes are:

* Amazon S3 for scalable storage
* Amazon Glue for unified data access
* AWS Lake Formation for management and unified governance
* Amazon Athena for interactive analytics directly in S3 using SQL.

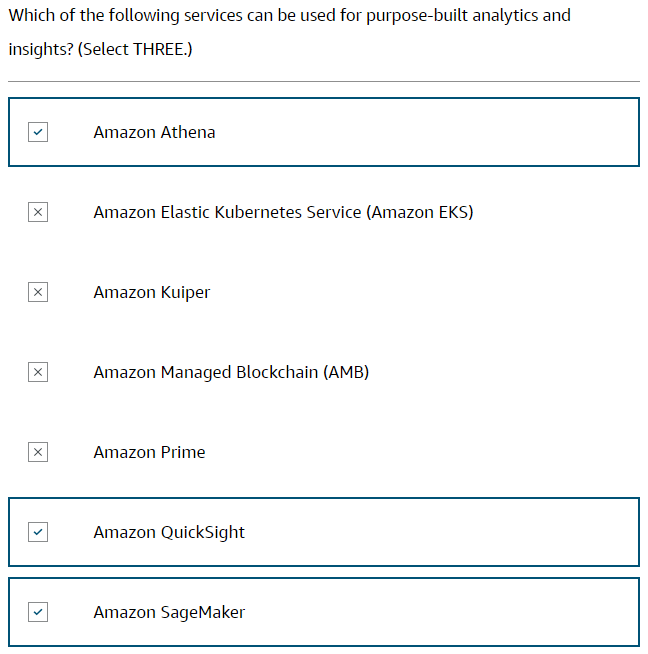
AWS services for purpose-built analytics can be:

* Amazon Redshift for cloud data warehousing
* Amazon EMR for big data applications and petabyte-scale data analytics
* Amazon Aurora for MySQL and PostgreSQL compatible OLTP databases
* Amazon DynamoDB for NoSQL key-value databases
* Amazon OpenSearch Service for open-source search and analytics
* Amazon Sagemaker for building, training, and deploying ML models.

The following diagram shows a modern data architecture using AWS services.



### **Knowledge check**



## **5. Common Use Cases**

### **AWS Analytics for Health Care: Gather, Store, and Analyse vast Volumes of Data**

##### **Challenges**

The healthcare industry generates large amounts of data.

The following are examples of healthcare data:

* Medical imagery
* Patient records
* Scientific and research metrics
* Financial information and accountancy records

That data is most useful when it can be aggregated and analysed in its totality.

Accordingly, one of the greatest challenges the healthcare industry faces is the isolation, or siloing, of information. Data silos require their own physical infrastructure and analyst teams to extract and analyse the data. Also, data types in one silo are often incompatible with data types in other silos. These factors can make data analysis prohibitively slow and expensive. Each data silo can require its own staff, and data formats between silos may be incompatible.

##### **Solution**

For healthcare data to be maximally useful, the data must meet the following conditions:

* Securely stored in a centralized repository
* Accessible to authorized personnel across many geographical locations
* Readily available to other specialized storage repositories and analytics tools

Data lakes on AWS meet all these requirements and more. The data lake, which can store almost any type of data, is at the centre of an architecture that enables cleansing, cataloguing, and analysis.

##### **Benefits**

Using data lakes on AWS, healthcare providers, researchers, and finance personnel can access aggregated data in near real-time and use it in a wide array of AWS services, including the following:

* AWS Glue helps to discover, prepare, and move data for machine learning and analytics applications.
* Amazon Redshift helps to analyse and process massive amounts of structured and semi-structured data.
* Lake Formation helps to provide unified governance and management of the data lake.

##### **References:**

* [Data Lakes on AWS](https://aws.amazon.com/big-data/datalakes-and-analytics/datalakes/)
* [Modern Data Architecture on AWS](https://aws.amazon.com/big-data/datalakes-and-analytics/modern-data-architecture/)
* [AWS Glue](https://aws.amazon.com/glue/)
* [Amazon Redshift](https://aws.amazon.com/redshift/)
* [AWS Lake Formation](https://aws.amazon.com/lake-formation/?gclid=CjwKCAiAx_GqBhBQEiwAlDNAZrDYjuxE1QSOJcNNHtXhooYo9WrackY1EshHyd1vMyAvq0ih6As-nhoCV9kQAvD_BwE&trk=30a5c4e0-0d3e-437e-b82b-d5463ae61868%E2%89%BBchannel=ps&ef_id=CjwKCAiAx_GqBhBQEiwAlDNAZrDYjuxE1QSOJcNNHtXhooYo9WrackY1EshHyd1vMyAvq0ih6As-nhoCV9kQAvD_BwE:G:s&s_kwcid=AL!4422!3!658520966861!!!g!!!19852662473!149878728220)

### **AWS Analytics for Automotive: Making Analysis Results meaningful and accessible.**

##### **Challenges**

With the increasing use of Internet of Things (IoT) devices, automotive companies generate massive amounts of real-time data in their design, prototyping, testing, and production lines. Much of this data is time-sensitive and needs to be analysed and acted on quickly. AWS data lakes, data warehouses, and a wide range of analytics services make this possible and affordable.

However, if the outputs of data analytics are not accessible or understandable by a wide enough range of staff, much of its value is lost. To reap the full benefits of data analytics, you need to display the results of the analysis clearly and meaningfully. Displaying technical data graphically, particularly with interactive graphics, can lead to deeper insights and improve your business outcomes.

Sharing data with a broader population over a wider geographical area presents another potential issue—maintaining security.

##### **Solution**

AWS offers a powerful array of services to solve these challenges. You can visualize your data in real-time as graphs, tables, charts, and more.

QuickSight is a business intelligence service with built-in machine learning integration. With QuickSight, you can pose natural language questions to analyse your data. QuickSight displays the results in clear, understandable formats.

You can use Amazon SageMaker Canvas to generate accurate predictions without having to write code of your own. You can access prebuilt foundation models (FMs) from Amazon, publicly available FMs, or customize your own models. Use SageMaker Canvas for sentiment analysis, object detection, or document analysis.

Amazon DataZone helps you build and manage a secure and efficient environment so that your staff can share data and insights across boundaries.

##### **Benefits**

Displaying data analytics results graphically can help you do the following:

* Reveal trends.
* Discover potential trouble spots in production and testing processes.
* Diagnose causes of inefficiencies.
* Make more accurate projections.

Making data analytics more accessible to a wider selection of staff can eliminate bottlenecks and speed up the path to business insights.

##### **References:**

* [Amazon QuickSight](https://aws.amazon.com/quicksight/)
* [Amazon SageMaker Canvas](https://aws.amazon.com/sagemaker/canvas/)
* [Amazon DataZone](https://aws.amazon.com/datazone/)

### **AWS Analytics for Energy: Using Data Lakes and Analytics for Energy assets.**

##### **Challenges**

Keeping renewable energy grids efficient and profitable has the following challenges:

* IT and OT (operational technology—the physical plant and machines) must be tightly integrated.
* Remote control centres and generation sites make data ingestion and storage complex and costly.
* Performance analysis and predictive maintenance are difficult because of the diverse and incompatible tools.
* Varied and non-standard formats make data sharing difficult.

##### **Solution**

Data lakes on AWS provide a centralized, secure repository for the massive data streams generated daily by renewable energy IoT devices. This breaks down the barriers to sharing data among and from remote sites.

With the AWS IoT Core, you can connect billions of devices and route trillions of messages to your AWS services, all without having to manage infrastructure.

Kinesis Data Firehose reliably captures, transforms, and delivers those massive data streams to the data lake.

##### **Benefits**

Centralizing the data ingestion and analytics empowers your analysts to discover trends, locate and diagnose inefficiencies, and make more accurate predictions.

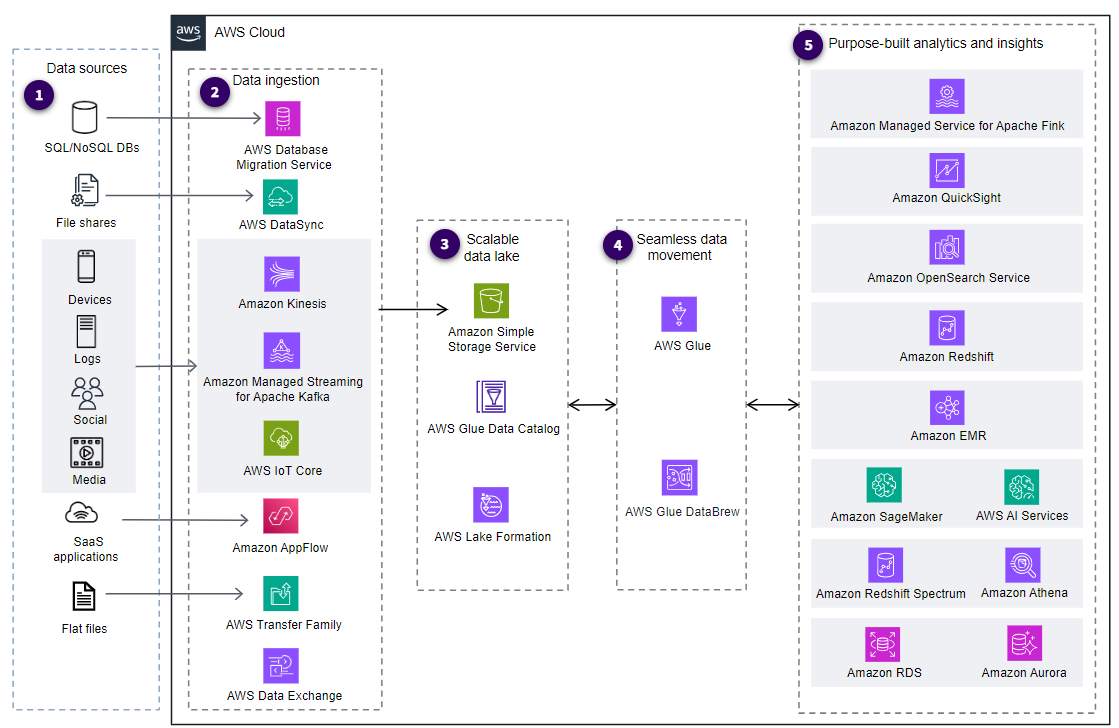
##### **References:**

* [AWS IoT Core](https://aws.amazon.com/iot-core/)
* [Amazon Kinesis](https://aws.amazon.com/pm/kinesis/?gclid=CjwKCAiA9ourBhAVEiwA3L5RFpY7kzzU11MwU2P2gqqxi6NQTvm_u8s8cE_eSEeqfcH1V-6tvzyvKxoCFNgQAvD_BwE&trk=1be29a9d-9928-42fd-a79b-ab6d005d22df%E2%89%BBchannel=ps&ef_id=CjwKCAiA9ourBhAVEiwA3L5RFpY7kzzU11MwU2P2gqqxi6NQTvm_u8s8cE_eSEeqfcH1V-6tvzyvKxoCFNgQAvD_BwE:G:s&s_kwcid=AL!4422!3!652240143316!p!!g!!kinesis%20firehose!19878797227!144181882021)

## **Reference Architecture**

### **Modern data and analytics reference architecture on AWS**

You can build your data and analytics pipelines using modern data and analytics reference architecture. The reference architecture shows the AWS services used at different stages of the data pipeline. You can use the specified AWS service based on your unique use cases.



### **Streaming data architecture (optional)**

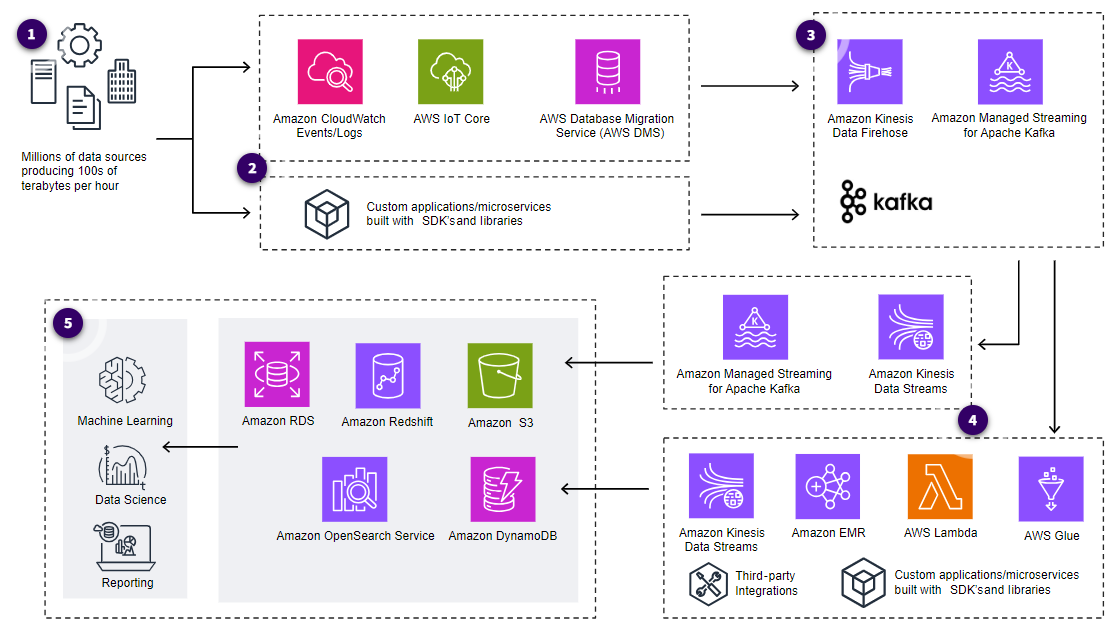
Streaming data architecture is a framework for handling stream data as it is generated or ingested. While modern data reference architecture encompasses various data processing frameworks, streaming data architecture specifically focuses on real-time or near real-time data processing.

Streaming data architecture is built on the following five core components:

* Data sources
* Stream ingestion
* Stream storage
* Stream processing
* Destinations

You can create and manage each of these components using AWS Managed Services (AMS). You also can deploy and manage them as a solution on Amazon EC2, Amazon Elastic Container Service (Amazon ECS), or Amazon Elastic Kubernetes Service (Amazon EKS).

* **Data sources** - Data sources include application and click stream logs, mobile apps, existing transactional relational and NoSQL databases, IoT sensors, and metering devices.
* **Stream ingestion and producers** - Data is collected from various sources and ingested into the streaming platform. Examples include custom streaming producers or pre-built producers such as Amazon Kinesis or Apache Kafka.
* **Stream storage** - Streaming data is stored for batch analytics, historical analysis, and compliance. Stream storage options include Kinesis Data Streams, Amazon MSK, and Apache Kafka.
* **Stream processing and consumers** - This component processes data as it arrives so it can be filtered, transformed, and enriched. Processing tools include Amazon EMR (Spark Structured Streaming and Apache Flink), AWS Glue ETL Streaming, Amazon Managed Service for Apache Flink, third-party integrations, and build-your-own custom applications using the AWS and open-source community SDKs and libraries.
* **Downstream destinations** - This is the end-users or applications that consume the processed streaming data. Examples include databases, data warehouses, purpose-built systems such as OpenSearch Service, data lakes, and various third-party integrations.

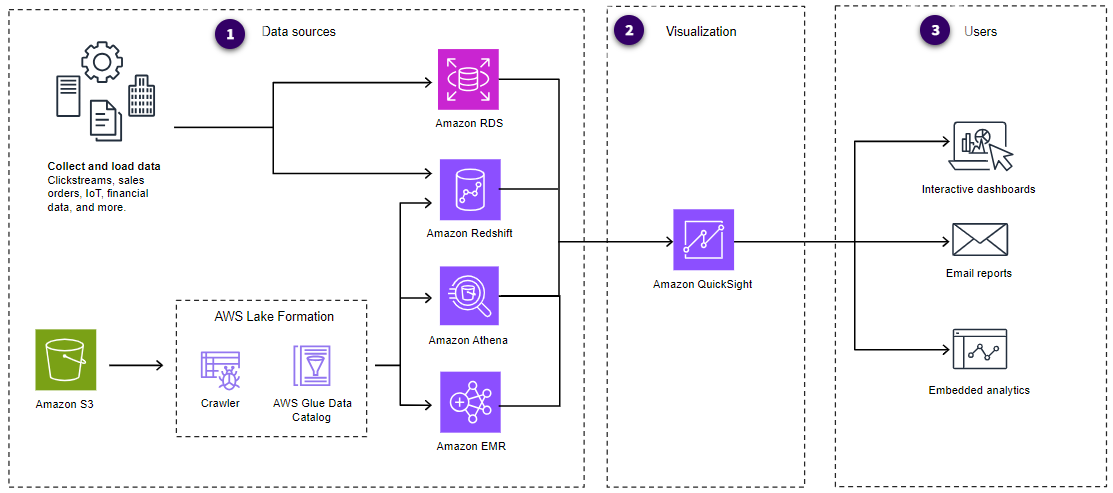


### **Data visualization architecture (optional)**

Data visualization architecture refers to the design and structure of the systems and components used to create, display, and interact with visual representations of data. Organizations can build a data visualization architecture to interpret information in an interactive, visual environment and accelerate data-driven insights that are easy to understand and navigate.

Data visualization architectures consist of the following core characteristics:

* **Scalability** - This characteristic ensures that the infrastructure can scale up vertically and horizontally to handle large volumes of data and growing numbers of users.
* **Connectivity** - Applications must be able to connect with data platforms such as traditional data warehouses and databases. Applications must also support connectivity to data lakes, modern data architectures, and non-traditional sources, such as SaaS applications.
* **Centralized security and compliance** - Robust security features, such as layered security, authentication authorization, and encryption are essential. This includes securing at the perimeter, securing the data in transit and data at rest, and restricting varying levels of access through fine-grained permissions for users. The application must also comply with the governmental and industry regulations.
* **Sharing and collaboration** - The architecture should support collaboration features and data democratization. It must have features that allow sharing of dashboards and visualization with others. Collaboration on data analysis is important to convey insights effectively.
* **Logging, monitoring, and auditing** - Built-in monitoring tools provide insights into user interactions, system performance, and usage patterns.  Applications must provide adequate mechanisms to monitor and audit usage for security (to prevent unwanted access to data assets and other resources) and troubleshooting.
* **Perform advance analytics** - Analytics allow organizations to analyse user behaviour and optimize visualizations based on user preferences. Applications must be able to discover hidden insights from the data, perform forecasting and what-if analysis, or add easy-to-understand natural language narratives to dashboards. Organizations should be able to perform analytics without deep statistical and machine learning knowledge.
* **Self-service interactions** - An important goal of all applications is to make the data more accessible to more people without extensive user training and technical understanding. Data must be available in all formats—raw, semi-processed, and processed. Self-service applications should allow users to interact with data on an as-needed basis without involving IT.



## **Assessment – 2**

