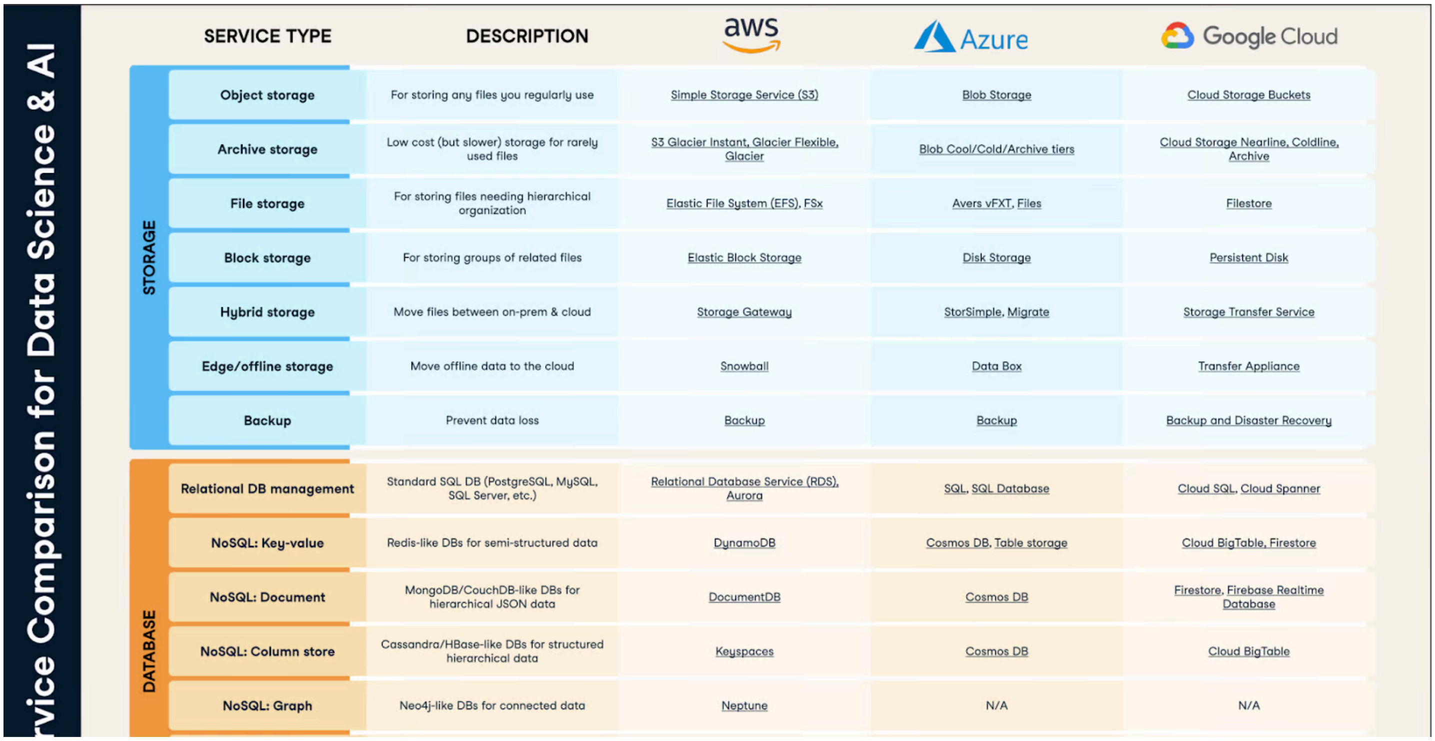
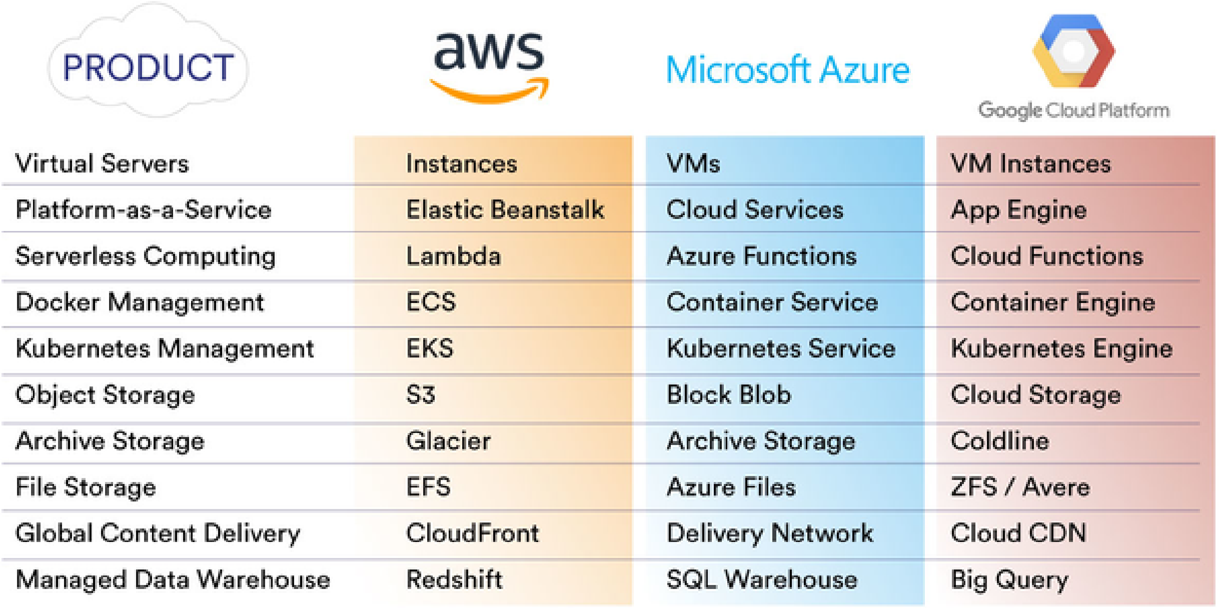
Day 2: Cloud-Based Data Engineering Applications



rvice Comparison for Data Science & AI

	SERVICE TYPE	DESCRIPTION	aws	Azure	Google Cloud
STORAGE	Object storage	For storing any files you regularly use	Simple Storage Service (S3)	Blob Storage	Cloud Storage Buckets
	Archive storage	Low cost (but slower) storage for rarely used files	S3 Glacier Instant, Glacier Flexible, Glacier	Blob Cool/Cold/Archive tiers	Cloud Storage Nearline, Coldline, Archive
	File storage	For storing files needing hierarchical organization	Elastic File System (EFS), FSx	Avers vFXT, Files	Filestore
	Block storage	For storing groups of related files	Elastic Block Storage	Disk Storage	Persistent Disk
	Hybrid storage	Move files between on-prem & cloud	Storage Gateway	StorSimple, Migrate	Storage Transfer Service
	Edge/offline storage	Move offline data to the cloud	Snowball	Data Box	Transfer Appliance
	Backup	Prevent data loss	Backup	Backup	Backup and Disaster Recovery
DATABASE	Relational DB management	Standard SQL DB (PostgreSQL, MySQL, SQL Server, etc.)	Relational Database Service (RDS), Aurora	SQL, SQL Database	Cloud SQL, Cloud Spanner
	NoSQL: Key-value	Redis-like DBs for semi-structured data	DynamoDB	Cosmos DB, Table storage	Cloud BigTable, Firestore
	NoSQL: Document	MongoDB/CouchDB-like DBs for hierarchical JSON data	DocumentDB	Cosmos DB	Firestore, Firebase Realtime Database
	NoSQL: Column store	Cassandra/HBase-like DBs for structured hierarchical data	Keyspaces	Cosmos DB	Cloud BigTable
	NoSQL: Graph	Neo4j-like DBs for connected data	Neptune	N/A	N/A



PRODUCT	aws	Microsoft Azure	Google Cloud Platform
Virtual Servers	Instances	VMs	VM Instances
Platform-as-a-Service	Elastic Beanstalk	Cloud Services	App Engine
Serverless Computing	Lambda	Azure Functions	Cloud Functions
Docker Management	ECS	Container Service	Container Engine
Kubernetes Management	EKS	Kubernetes Service	Kubernetes Engine
Object Storage	S3	Block Blob	Cloud Storage
Archive Storage	Glacier	Archive Storage	Coldline
File Storage	EFS	Azure Files	ZFS / Avere
Global Content Delivery	CloudFront	Delivery Network	Cloud CDN
Managed Data Warehouse	Redshift	SQL Warehouse	Big Query

Aspect	Data Analyst	Data Engineer

Role	Focuses on analysing data and interpreting data to generate insight for business decision	Builds and maintain the infrastructure for data collection, storage and processing
Key responsibilities	-Performs data cleaning and preprocessing -Develops dashboards and reports -use Statistical methods to extract insights -interprets trends and patterns -support decision-making with data driven insight.	-Design and building scalable data pipelines -Manage data ingestion, RTI/ELT processes -Optimise and maintain databases -Ensure data quality, integrity and security - Work with Big Data technologies and cloud platforms.
Skill Set	SQL, Excel, python(Pandas.Numpy) -Power Bi, Tableau -Stastical Analysis & Machine Learning -Business Intelligence and Data Story telling -Communication and Problem solving	SQL, python, Java, Scala -Apache Spark, kafka, Airflow - ETI Tools (**AWS- Glue**, **Azure- Data Factory**) -Cloud data Service(AWS, Azure GCP) - Big Data Storage and Processing(Hadoop, Delta Lake and Snowflake)
KRA(Key result Areas)	Accuracy and effectiveness of Data visualization and report Speed and efficiency in delivering insight Impact of insight on business decisions Collaboration with business teams	-Performance and scalability of data pipelines - Reliability and security of data infrastructure - Data availability and accessibility - Optimization of storage and processing costs.
Tools & technologies	Power BI, tableau, Excel and Python(Pandas and matplotlib) SQl	Apache Spark/Pyspark, Hadoop & Airflow AWS Glue , Azure data Factory and Google Big Query
END goal	Generate Actinable insight for business growth	Ensure high quality, well structured and accessible data for analyst and business users

🔹 1. Cloud-Native Data Analytics Services Overview

📖 Definition:

Cloud-native data analytics services are **fully managed** platforms provided by cloud vendors (AWS, Azure, GCP) that allow organizations to process, analyze, and derive insights from large volumes of data without managing underlying infrastructure.

🚀 Features:

✔ **Scalability** – Handle petabytes of data with on-demand scaling
✔ **Serverless & Managed** – No need to manage servers, automatic resource allocation
✔ **Integration** – Connects with various storage, ETL, and AI/ML tools
✔ **Multi-Format Support** – Works with structured, semi-structured, and unstructured data

🛠️ Best Practices:

✅ Use **columnar storage formats** (Parquet, ORC) for optimized query performance
✅ **Partition & index data** for cost-effective and faster querying
✅ Implement **security & access control** via IAM roles and encryption
✅ Optimize queries using **pre-aggregations and caching techniques**

📌 Case Study: Netflix & BigQuery

Netflix processes large-scale streaming data using **Google BigQuery** to run real-time analytics on user engagement, recommendation systems, and content optimization.

📊 Use Cases:

🔹 Real-time fraud detection using **AWS Athena** & **Kinesis**
🔹 Customer behavior analysis in **Google BigQuery**
🔹 Predictive analytics for sales forecasting using **Azure Synapse Analytics**

🔹 2. Data Storage & Processing in Cloud

📖 Definition:

Data storage in the cloud enables **scalable, durable, and highly available** storage solutions optimized for analytics, batch processing, and real-time workloads.

🚀 Features:

✔ **Durability & Availability** – Cloud storage ensures **99.999999999% (11 9s)** durability
✔ **Multi-Format Support** – Supports structured (SQL), semi-structured (JSON, Avro), and unstructured (images, logs) data
✔ **Data Lifecycle Management** – Tiered storage options (hot, cool, archive) for cost efficiency

🛠️ Best Practices:

✅ Store **raw data in cloud storage** (S3, Blob, GCS) and transform as needed
✅ Use **compression & optimized formats** (Parquet, Avro) for better performance
✅ Enable **versioning & data retention policies** for governance

📌 Case Study: Uber & Delta Lake

Uber leverages **Delta Lake on AWS S3** for managing historical trip data, ensuring **data consistency, performance, and reliability** for large-scale analytics.

📊 Use Cases:

- 🔹 Storing **real-time event logs** in **Amazon S3** for ML model training
- 🔹 Managing **IoT sensor data** in **Azure Data Lake Storage**
- 🔹 Creating a **data lakehouse** architecture with **Delta Lake on Databricks**

🔹 3. Serverless Data Analytics & Querying

📖 Definition:

Serverless data analytics allows users to query, process, and analyze large datasets without managing infrastructure, reducing costs and complexity.

🚀 Features:

✔ **No Infrastructure Management** – Fully managed query execution
✔ **Pay-per-Query Pricing** – Users are charged based on data scanned
✔ **High Performance** – Uses distributed computing for faster results

🛠️ Best Practices:

✅ **Optimize queries** by filtering unnecessary columns & rows
✅ **Use caching mechanisms** to store frequently accessed data
✅ **Set query limits & cost alerts** to avoid unexpected charges

📌 Case Study: Airbnb & AWS Athena

Airbnb leverages **AWS Athena** to analyze petabytes of **guest booking data**, reducing **query costs by 30%** while improving insight generation speed.

📊 Use Cases:

- 🔹 Running **ad-hoc analytics queries** on S3-stored data
- 🔹 Analyzing **clickstream logs for website optimization**
- 🔹 Generating **customer segmentation reports** in BigQuery

🔹 4. Hands-on Demo: Cloud-Based Data Engineering Workflow

💻 Practical Demo:

📌 **Scenario:** Querying Large Datasets on Cloud Storage
✅ Upload **CSV/Parquet** data to **Amazon S3**
✅ Run an **AWS Athena query** to analyze customer transactions
✅ Optimize query performance using **partitioning & indexing**

🎯 Outcome:

✔ Learn how to process & analyze data without managing infrastructure
✔ Understand cost-efficient data querying techniques
✔ Gain insights into optimizing cloud-based analytical workflows

---

📌 Summary Table: Key Concepts & Applications

Concept	Definition	Best Practices	Use Cases

Cloud Data Analytics	Fully managed platforms for large-scale data processing	Use **optimized storage formats** & partitioning	Fraud detection, Customer insights, Predictive modeling
Cloud Storage	Scalable storage for structured & unstructured data	Use **tiered storage** & compression for cost efficiency	Data lakes, Backup & disaster recovery, IoT data storage
Serverless Querying	On-demand query execution without infrastructure	Optimize **query execution & limit costs**	Ad-hoc reporting, Log analysis, Performance monitoring
Data Lakehouse	Hybrid of data lakes & warehouses for structured + unstructured data	Use **Delta Lake for versioning & consistency**	Unified analytics, Streaming + batch processing

---

📌 Conclusion & Next Steps

🔹 After this session, learners should:
✅ **Understand cloud-native analytics tools & data storage solutions**
✅ **Know best practices for cloud-based data engineering workflows**
✅ **Be ready to implement ETL & ELT pipelines using AWS & Azure**

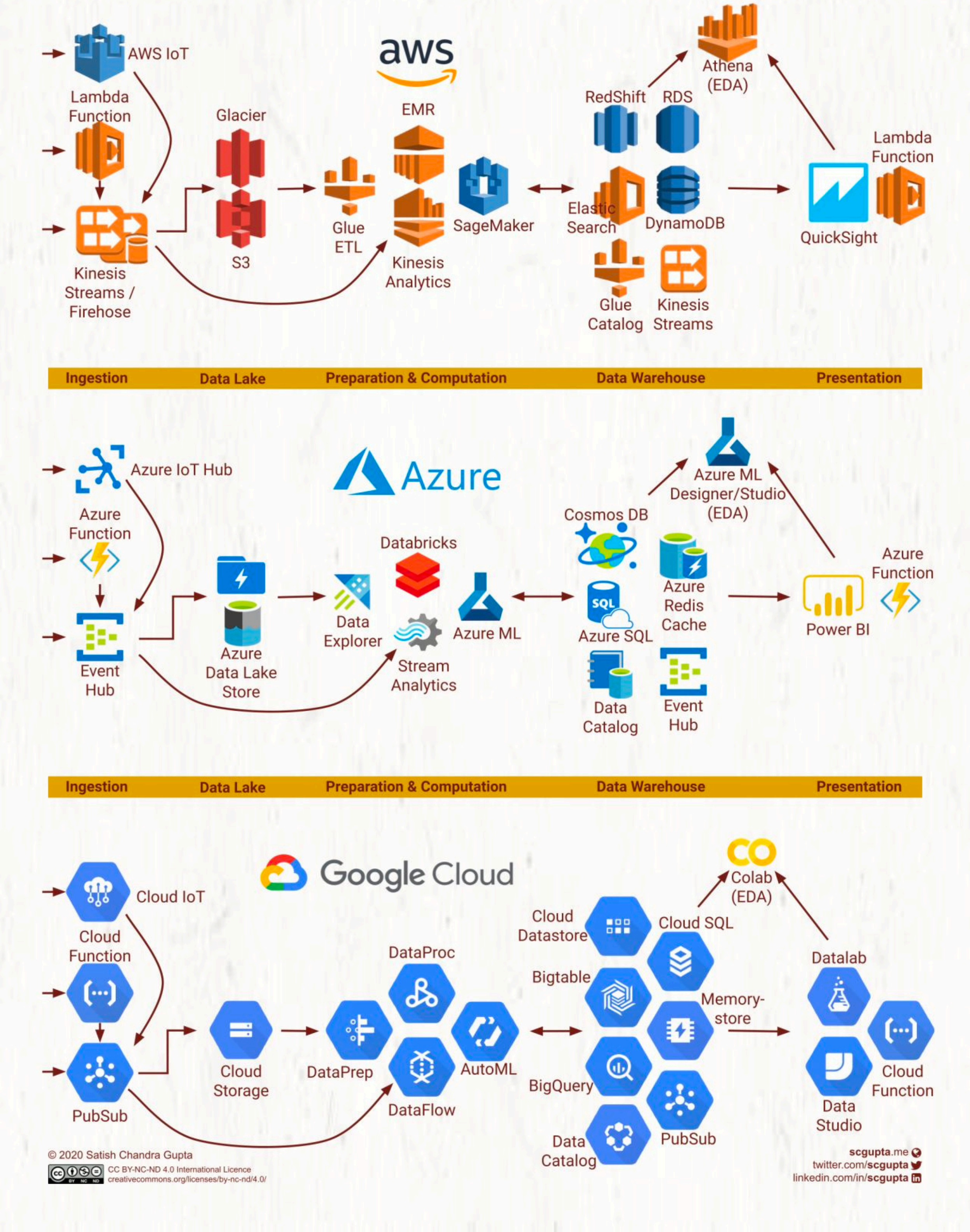
Day 2: Cloud-Based Data Engineering Applications

Topic	Key Points Covered

Cloud-Native Data Analytics Services Overview	- Understanding Managed Cloud Data Warehouses (Amazon Redshift, Google BigQuery, Azure Synapse Analytics) - Serverless vs. Provisioned Analytics Services (AWS Athena, Azure Data Explorer) - Cloud-based ETL & Data Pipeline Services (AWS Glue, Azure Data Factory, Google Dataflow)
Data Storage & Processing in Cloud	- **Data Lake vs. Data Warehouse** (Architecture & Use Cases) - **Cloud Storage for Analytics** (Amazon S3, Azure Blob Storage, Google Cloud Storage) - **File Formats for Analytics** (Parquet, Avro, ORC vs. CSV, JSON) - **Introduction to Delta Lake & Data Versioning**
Serverless Data Analytics & Querying	- **Introduction to Serverless Computing for Data** (AWS Lambda, Azure Functions, Google Cloud Functions) - **Data Querying without Infrastructure Management** (AWS Athena, Google BigQuery, Azure Data Explorer) - **Streaming Data Processing Basics** (Apache Kafka, Kinesis, Event Hub for real-time analytics)
Hands-on Demo (Interactive, Live Demo Preferred)	- **Querying Large Datasets in the Cloud Using Serverless Engines** *(Example: Run an SQL query on a large dataset in AWS Athena or Google BigQuery)* - **Storing & Processing Structured/Unstructured Data in Cloud Storage** *(Example: Upload sample data to Amazon S3 & query it using AWS Glue)*

Typical Data pipeline has following steps :

Step 1: Data Ingestion
Step 2: Data Lake
Step 3: Preparation and Computation
Step 4: Datawarehouse
Step 5: Data visualisation/presentation



© 2020 Satish Chandra Gupta

scgupta.me
twitter.com/scgupta
linkedin.com/in/scgupta