**Data Engineering:**

**Overview:**

* Big Data
* Data processing pipelines - How Data Flows

**What is Big Data?**

* Large datasets and Technology.
* The 3 Vs of big data.
  + The first one is **volume**, just the sheer scale of the data being handled by a system.
    - Megabytes
    - Gigabytes
    - Terabytes
    - Petabytes
    - Zettabytes
  + The next V is **velocity**. The speed at which data is being processed. That doesn't just mean being stored, but analyzed, visualized, or used in some productive way.
    - AD-hoc
    - Batched
    - Near real time
    - Real time
  + Finally, **variety**, the diversity of data from different sources in various different formats and a varying quality.
    - Key/value
    - Tabular data
    - Images
    - Audio
    - Video
    - Unstructured
    - Mobile
    - IOT

**Data Processing Pipeline:**

Let's define some further big data terms that will come up frequently in our paths as data engineers.

* + Data Warehouses and Data lakes.
  + OLTP versus OLAP or online transactional processing versus online analytical processing
  + SQL versus NoSQL
  + Batch versus streaming.

**Data Warehouses and Data lakes:**

These are both abstract ways of thinking about how data is stored.

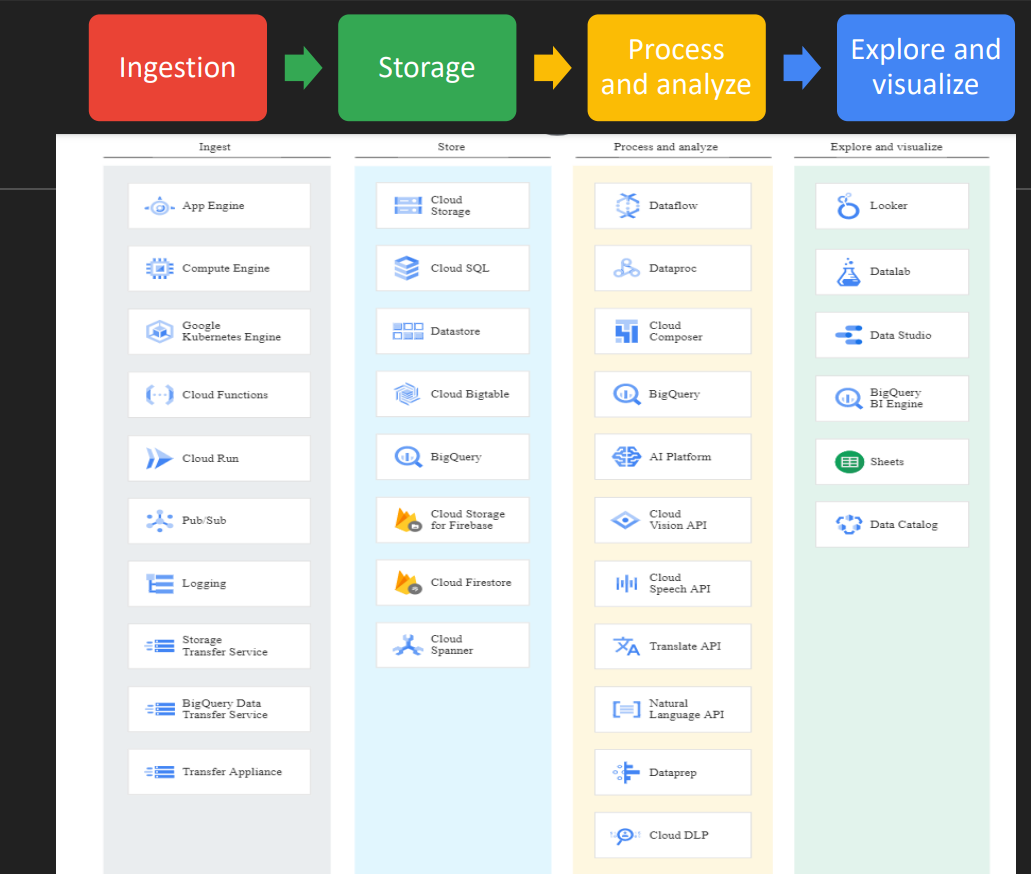
* **Data warehouses:**
  + Structured and/or processed.
    - Data may already have been processed or transformed to be stored in its structured place.
  + Ready to use.
    - It doesn't require any further transformation and can be consumed easily.
  + Rigid
    - Data maybe easier to understand but less up to date. Structures are hard to change.
* **Data Lake:**
  + Raw and/or Unstructured:
    - Data lake contains all raw, unprocessed data, before any kind of transformation and organization.
  + Ready to Analyze:
    - Data is more up to date but may require more advanced tools for analysis.
  + Flexible:
    - No structure is enforced , so new types of data can be added.

**OLTP versus OLAP or online transactional processing versus online analytical processing:**

* **OLTP** and **OLAP** are both online database systems.
* An **OLTP system** usually processes a high number of short transactions like select or insert statements in basic SQL terms. Queries should be fast and the system should maintain a high level of data integrity.
  + Example purchase transaction in SQL database is an OLTP system.
  + It will modify the database.
* An **OLAP system** by contrast will typically run a lower volume of transactions, but they could be longer running queries. An OLAP system might also run these queries over aggregated historical data, which could have been sourced from other OLTP systems, even multiple ones.
  + So following our previous example, this system could be used to produce a report on purchases and purchasing trends over time.
  + It will query the database.
* ETL frameworks will move data from OLTP system to OLAP system.

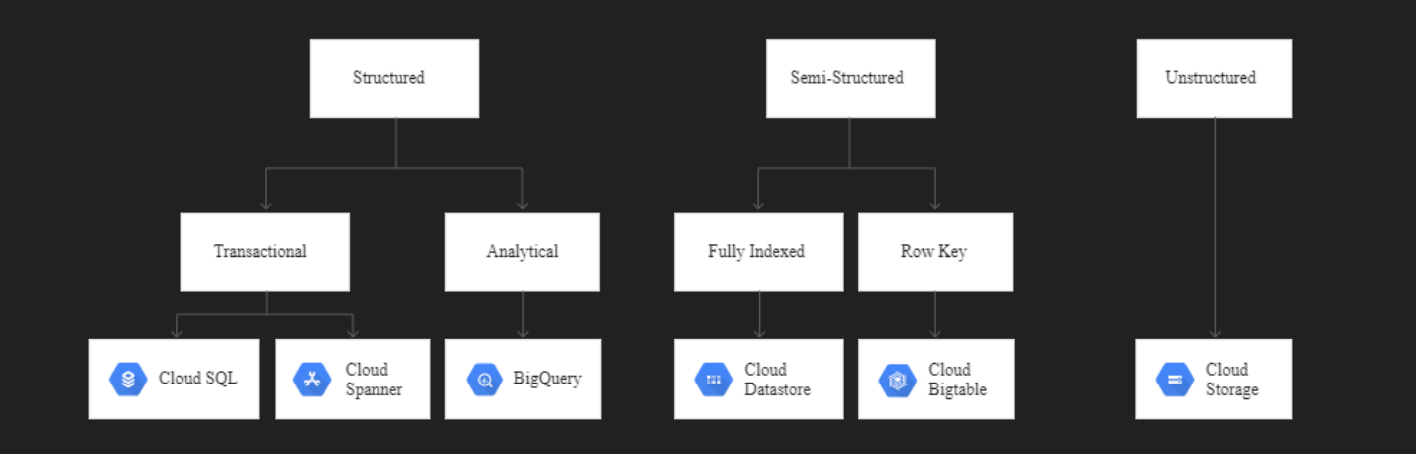
**Ingest:**

* Gather Data from multiple sources
* Data gather from App
  + Event Log,
  + Click stream Data,
  + ecommerce Transaction
* Streaming Ingest
  + Pub Sub
* Batch Ingest
  + Different Transfer services
  + GCS – gsutil



**Storage:**

* Cost efficient & durable data storage.



**Types of Data:**

1. Structured.
2. Semi-Structured
3. Unstructured

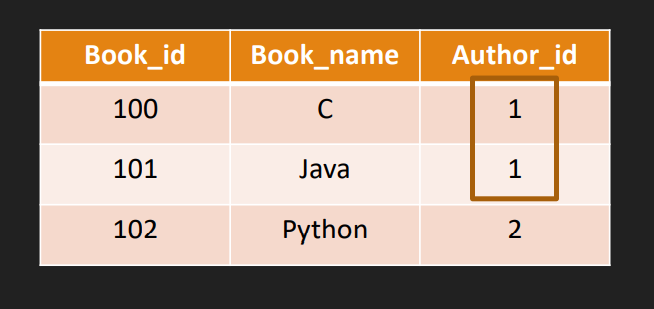
**Structured Data:**

* Tabular Data
* Represented by Rows & Columns
* SQL can be used to interact with data
* Fixed Schema
* Each row has same number of columns
* Relational Database are structured
* MySQL, Oracle SQL, PostgreSQL, MSSQL
* In GCP, Cloud SQL, Cloud Spanner

**SQL and NoSQL Databases:**

**SQL Databases:**

* SQL or SQL databases contains structured tabular data, which means rows and columns stored in tables.
* Take for example, storing a customer entity in a SQL database.
* The information we store is entered into these columns, which must follow a strict schema.
* SQL databases are relational databases, which means that entities and tables can contain relationships to other entities, enabling you to build relatively complex queries.



**Semi-Structured Data:**

* Each Record has variable number of Properties
* No Fixed Schema
* Flexible structure
* NoSQL kind of Data
* Store data as key-value pair
* JSON – Java Script object Notation are base way to represent semi structure data ¬ MongoDB, Cassandra, Redis, Neo4j
* In GCP, BigTable, Datastore, memoryStore.

**NoSQL Databases:**

* NoSQL databases include key value stores, JSON document stores, and a few other variants.
* The most common NoSQL systems you may have heard of are **MongoDB** or **Apache Cassandra**.

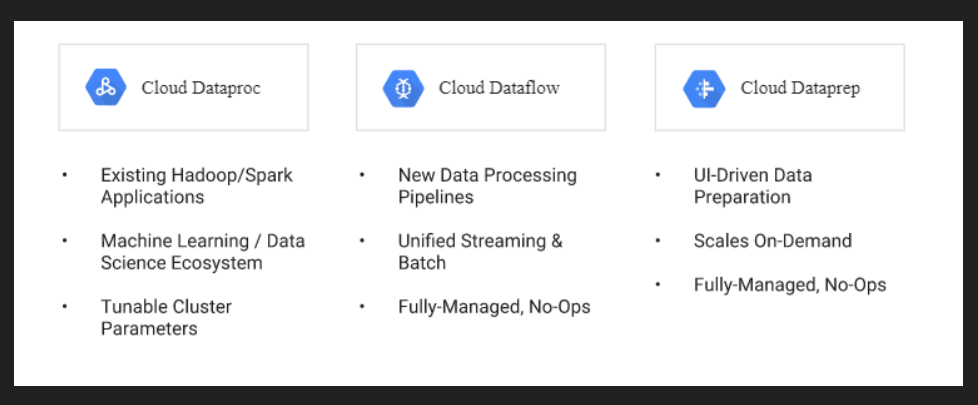


**Unstructured Data:**

* No Predefine Structure in Data
* Image
* video data,
* natural Language are example of unstructured data
* Google Cloud Storage, File store inside GCP to store Unstructure data.

**Process & Analyze:**

* What kind of outcome you want
* What analysis you want to perform
* Convert Data into meaning
* Analyze Data with BigQuery
* Apply ML with
  + BigQuery ML
  + Spark ML with DataProc
  + Vertex API
  + Build ML Model with Auto ML/Custom Model



**ETL and ELT:**

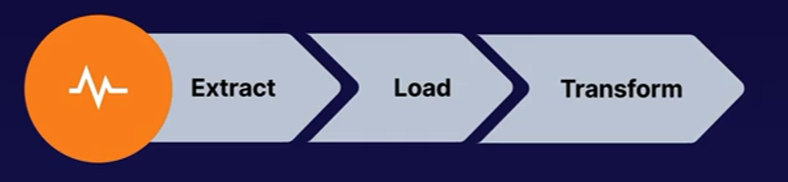
**ETL:**

* Data is taken from the source and manipulated to fit the parameters of the destination system.



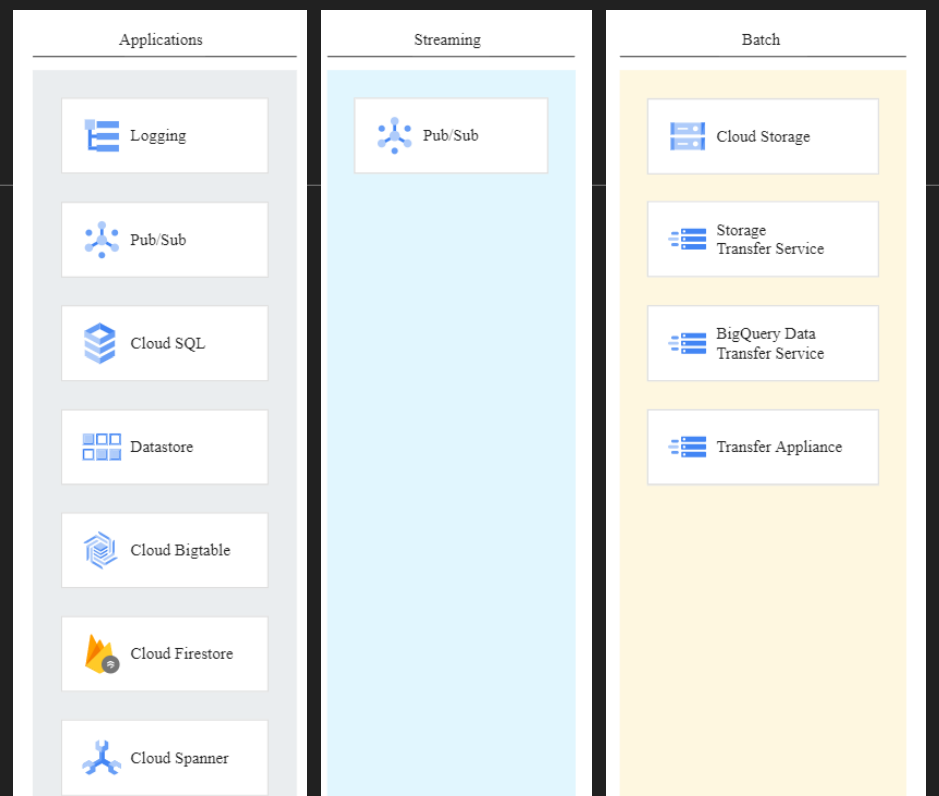
**ELT:**

* Data is loaded into data lakes and transformations can take place later.



**Common Transformations:**

* Formatting
* Labelling
* Filtering
* Validating



**Batch Data vs Streaming Data:**

**Batch Data:**

* Batch data is simply data that is gathered together, usually within a defined time window and loaded into a system, all in one go.
* Defined Start & End of data
* data size is known
* Processing High volume of data after certain periodic interval
* Long time to process data
* Payment processing

**Streaming Data:**

* Streaming on the other hand is just as it sounds, the continuous collection of data into a system.

Data is collected as it happens if you like every transaction or metric being sent immediately to the system. It can be harder to process streaming data because of the sheer volume of it.

* Unbounded, No End defined
* Data is processed as it arrives
* Size is unknown
* No much heavy processing
* take millisecond - seconds to process data
* Stock data processing

**Explore & visualize:**

* **Google Data Studio** - Easy to use BI Engine
  + Dashboard & Visualization
* **Datalab** 
  + Interactive Jupyter Notebook
  + Support for all Data Science Library
* ML Prebuilt API
  + Vision API
  + Speech API