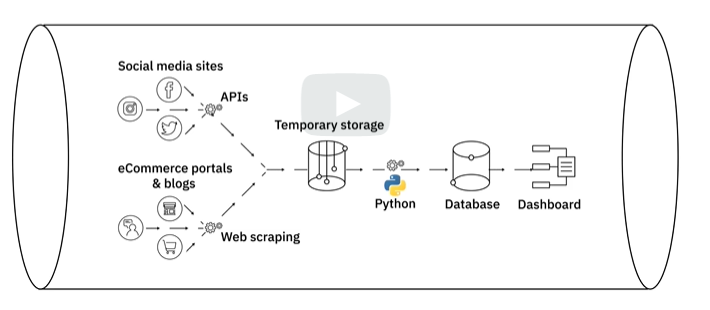
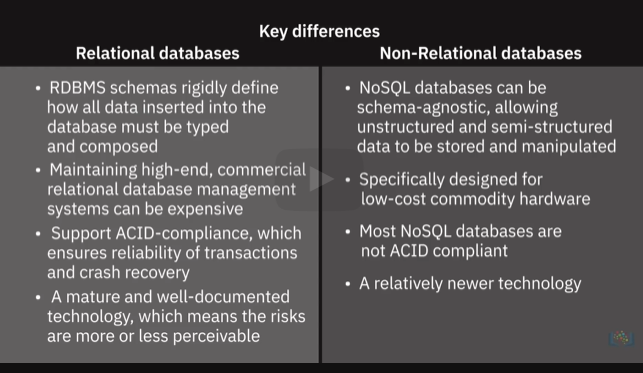
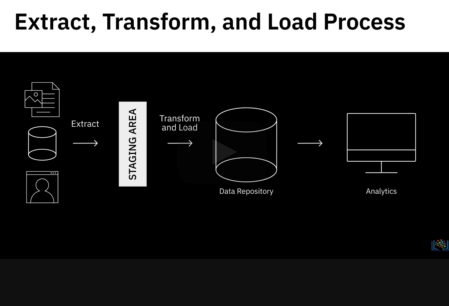
|  |  |
| --- | --- |
| **MODERN DATA ECOSYSTEM** | CHALLENGES |
| Integration  Insights  Stakeholders  Tool, applications and infrastructure to store, process and disseminate  Data Sources - structure, unstructured and semi-structure  Text, images, video, social, IOT, real-time events, date suppliers, web services, API’s | Reliability, security and data integrity |
| SOURCE ->Repository  REPO TYPES  1/ Transactional or OLTP repos - relation or NoSQL  2/ OLAP- Relationsional, NoSQL, Data warehouse, marts, lakes | Availability, flexibility, accessibility and securities |
| REPOSITORY -> Users  Stakeholders  API’s  Programmers  BI Analysts  Data Scientists | Provide quality data in the right format and shape  Interfaces  API’s  Applications |
| **DATA ENGINEER**  COLLECT: Extract, integrate and organise data  PROCESS: Clean, transform and prepare  STORE: Design and managed pipelines, source to destination systems, store and manage repos  INFRARSTRUCTURE - setup and manage infrastructure for ingestion processing and storage of data platform, store, distribute systems and data repos.  AVAILABLE: Make available to users securely | **SKILLS**  **MAIN - SQL, Modelling, ETL, Python.**   * Programming, architecture and SQL, NoSQL * Distributed computing, * DevOps, * Implementing ML models, * Operating Systems, * Infrastructure components(VM, Networking, App Services, Cloud-based services), * Databases and Datawarehouses (RDBMS- DB2, MySQl), (NoSQL- redi, Mongo, Cassamdra), (DW- DB2, Redshift, Synapse) * Pipelines with Beam, AirFlow and DataFlow. * ETL Tools - infosphere, AWS glue * Query, Programming and Shell * Hadoop, Hive and Spark   **TASKS**  1/ Data acquisition from multiple sources  2/ Architecture for storing  3/ Distributed systems for processing  4/ Pipelines for ETL  5/ Safeguard quality, privacy and security  6/ Performance optimisation  7/ Adhere to compliance guidelines  8/ Scalability for storage  9/ Scalability, privacy, security, compliance, monitoring backup and recovery  10/ API’s, services and programs for retriving date for end-users  11/ User access through interfaces and dashboards  12/ Data security checks. |
|  | **What is the best way to achieve what the stakeholders need (Accurate, reliable, compliant and accessible data) which is the right solution for long term use and stored in a secure reliable and highly available way.** |
|  | **FUNCTIONAL SKILLS**  -Convert business requirements into tech specs  -Work with complete software development lifecycle  -Understand the value of data  - Understand the risks of poor data management (quality, privacy, security, compliance)  - communicate with business users, data scientists, analysts and tech teams. |

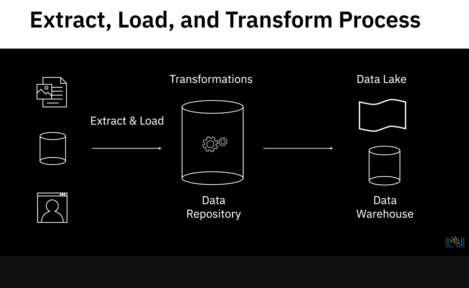


|  |  |
| --- | --- |
| **STRUCTURE**  SQL  OLTP  Spreadsheets  Forms  RFID GPS  Logs  **SEMI-STRUCTURED** (XML and JSON)  Some organisational properties  Emails,  XML  Zip files  UNSTRUCTURED  Web pages, images, video, audio  logs | **FILE FORMATS**  CSV - common seprated  TSV - tab seperated  Delimiter can be comma, tab, colon, vertical bar or space  XLSX  XML  PDF  JSON |
| **SOURCES OF DATA**  Internal Databases  Flat file and spreadsheets  XML  API and Webservices (web or network requests)  Scraping - (tools - BeautifulSoup, Scrapy, Pandas, Selenium)  Data Streams and Feeds - (GPS, Trading, Retail, Social Media) | **REPOS**  1/Relational  2/NoSQL - schema-less, speed, flexibility, scale. Used for Big data processing.  Schema-less, unstructured and semi-structured.   1. Key-Value - redis, dynamo 2. Document - Mongo, CouchDB 3. Column - Casandra, HBASE 4. Graph - Neo4J, CosmosDB   3/ Data Warehouse - central repo  ETL process. Can be SQL or NoSQL  4/ Big Data Stores - distributed infrastructure, store, scale and process large datasets. |
| **VIEWPOINT**  CSV, JSON, XML into SQL, NoSQL, Big Data Repo.  Streaming data and data at rest. |  |

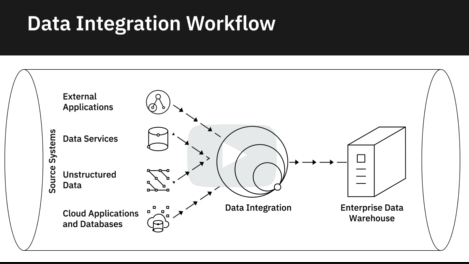


|  |  |
| --- | --- |
| **DATAWAREHOUSE, MARTS AND LAKES**  Purpose is reporting, analysis and insight  **DW**  **3 tier architecture**  Client Front-End layer(query, reporting, analyzing)  OLAP server (process data from db server)  Database server  **Data Marts**  Independent of DW (data from outside)  Dependent on DW  Hybrid  **LAKE**  Structured, Unstructured, Semi-Structure.  Not sure of use case yet. | DW’s - IBM Db2, Redshift, Synapse, Google BigQuery, SnowFlake.  Improve response time, accelerate business processes, faster data-driven decision making. |
|  | **HOW TO SELECT REPO TYPE**  Structured, Semi or Unstruct  Schema?  Performance requirements  Rest or streaming data?  Encryption needs  Volume of data-> Do you need a big data system.  Update Frequency  Scalability  Purpose  Transactional, Analytical, Archival, Datawarehousing. |
| **PIPELINES**  ELT  Extract  Batch - Stich and Blendo  Streaming - Kafka, Storm  Transform  Standardisse dates, units  Remove dupes  Filter out date not required  Enrich data  Establish table relationships - Modelling.  Business rules and validation  Loading  Initial, Incremental, full refresh. Load verfication missing or null values, server performance, load failures. | ELT - best for Big Data  Good for unstructured  Ingest raw data immediatly  Greater flexibility to analysts for EDA  Transform just the data you are going to use. |



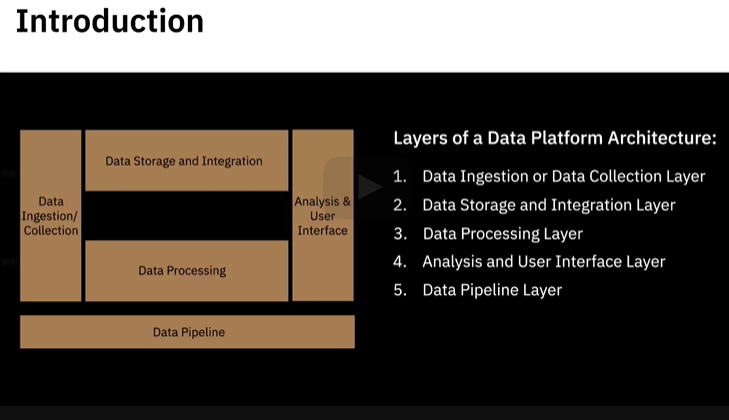


|  |  |
| --- | --- |
| **DATA PIPELINE**  Entire journey of moving data including ETL process  Batch and streaming  Typically loads into a lake. |  |
| **DATA INTEGRATION**  Usage scenarios  Data consistency across applications  Master data management  Data sharing between enterprises  Data migration and consolidation |  |



|  |  |
| --- | --- |
| **BIG DATA**  Spark and hadoop to extract load analyse  Hadoop - storage and processing across clusters  Scalable nodes, all data tyes.  HDFS - storage -> Disk Space.  Hive - Datawarehouse on Hadoop.  Spark - real time analysis ad processing and ETL. |  |

**DATA ENGINEERING LIFECYCLE**



|  |  |
| --- | --- |
| **INGEST**  Connect to data sources,  Transfer data from data courses to the data platform in stream and batch  Metadata  **STORE**  Transform ad merge  Make data available | Kafka and Kinesis  Storage : NoSQL and SQL Databases  Integration Tools - talend data fabric, ibm cloud pak for integration. |
| **DATA PROCESSING**  Structuring - schema  Normalisation - cleaning, inconsistency  Denormalisation - combine date from multiple tables  Data Cleaning |  |

**DATA REPO DESIGN**

|  |  |
| --- | --- |
| 1/ SQL  2/ NoSQL   * Documnet - not best for complex queries * Graph - not good for large volume analytic queries   3/ Data Lake - large volumes in native format  4/ Big Data Repo - high vol, high velocity, need distribute processing for fast analytics  Split large files for parralle access. | **Intended use?**  Number or Transactions  Frequency of Updates  Type of Operations  Response Time  Backup  Transactional or Analytical?  Scalability requirement  Performance (Throughput/latency)  Availbility  Integrity/ Security  Recoverability  Regulations GDPR, CCPA, HIPAA |
| **GATHERING AND IMPORTING DATA**  SQL query  NoSQL queries  API’s  Web Scraping  RSS  Data Streams  Data Exchange Platforms |  |
| **DATA WRANGLING/MUNGING**  Structuring  Merging relationsional with web api.  Normalising/denormalising  Cleaning  missing values ->delete on input  Duplicate data  Irrelevant data  Standardisation  Syntax Errors.  Outliers. | Tools for wrangling  Excel Power Query  Open refine  Google Data Prep  Watson Studio refinery  Trifacta Wrangler  Python (Numpy/ Pandas)  R |

DATA OPS

Gartner defines DataOps as a collaborative data management practice focused on improving the communication, integration, and automation of data flows between data managers and consumers across an organization. DataOps aims to create predictable delivery and change management of data, data models, and related artifacts. DataOps uses technology to automate data delivery with the appropriate levels of security, quality, and metadata to improve the use and value of data in a dynamic environment.”

(Source: [https://blogs.gartner.com/nick-heudecker/hyping-dataops/](https://blogs.gartner.com/nick-heudecker/hyping-dataops/" \o "Gartner blog Hyping DataOps" \t "https://courses.edx.org/xblock/_blank))

A small team working on a simpler or limited number of use cases can meet business requirements efficiently. As data pipelines and data infrastructures get more complex, and data teams and consumers grow in size, you need development processes and efficient collaboration between teams to govern the data and analytics lifecycle. From data ingestion and data processing to analytics and reporting, you need to reduce data defects, ensure shorter cycle times, and ensure 360-degree access to quality data for all stakeholders.

DataOps helps you achieve this through metadata management, workflow, and test automation, code repositories, collaboration tools, and orchestration to help manage complex tasks and workflows. Using the DataOps methodology ensures all activities occur in the right order the right security permissions. It helps set in a continual process that allows you to cut wastages, streamline steps, automate processes, increase throughput, and improve continually. Several DataOps Platforms are available in the market, some of the popular ones being IBM DataOps, Nexla, Switchboard, Streamsets, and Infoworks.

## DataOps Methodology:

The purpose of the DataOps Methodology is to enable an organization to utilize a repeatable process to build and deploy analytics and data pipelines. Successful implementation of this methodology allows an organization to know, trust, and use data to drive value.

It ensures that the data used in problem-solving and decision-making is relevant, reliable, and traceable and improves the probability of achieving desired business outcomes. And it does so by tackling the challenges associated with inefficiencies in accessing, preparing, integrating, and making data available.

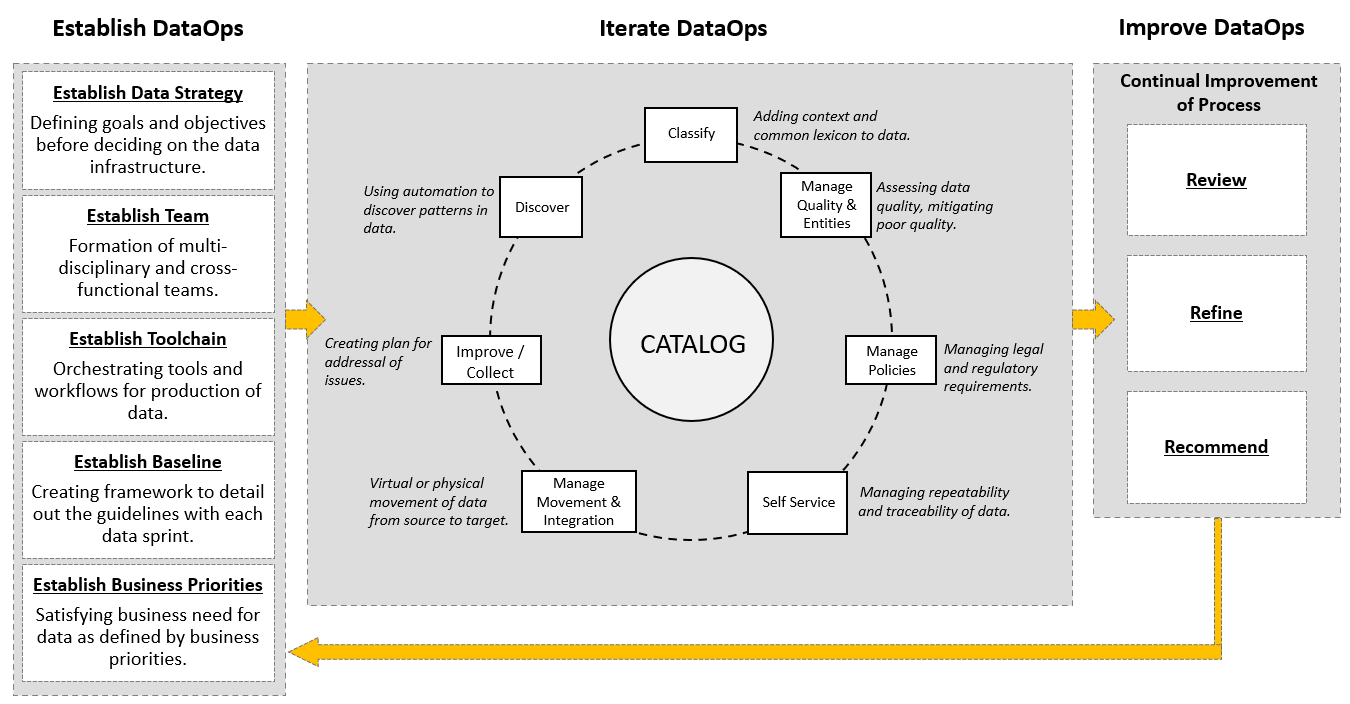
In a nutshell, the DataOps Methodology consists of three main phases:

The ****Establish DataOps Phase**** provides guidance on how to set up the organization for success in managing data.

The ****Iterate DataOps Phase**** delivers the data for one defined sprint.

The ****Improve DataOps Phase**** ensures learnings from each sprint is channeled back to continually improve the DataOps process.

The figure below presents a high-level overview of these phases and the key activities within each of these phases



### **Benefits of using the DataOps methodology:**

Adopting the DataOps methodology helps organizations to organize their data and make it more trusted and secure. Using the DataOps methodology, organizations can:

Automate metadata management and catalog data assets, making them easy to access.

Trace data lineage to establish its credibility and for compliance and audit purposes.

Automate workflows and jobs in the data lifecycle to ensure data integrity, relevancy, and security.

Streamline the workflow and processes to ensure data access and delivery needs can be met at optimal speed.

Ensure a business-ready data pipeline that is always available for all data consumers and business stakeholders.

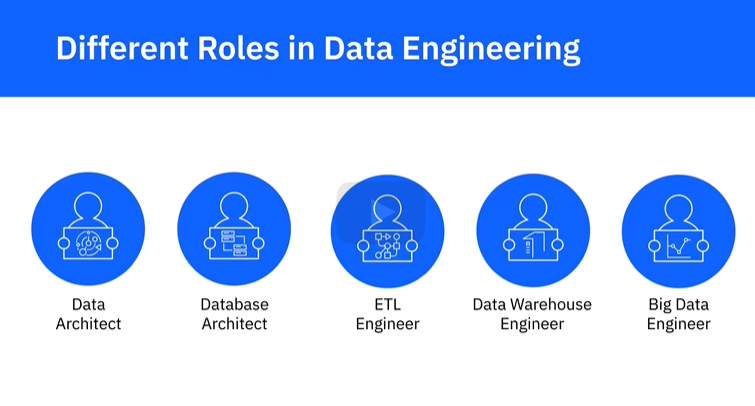
Build a data-driven culture in the organization through automation, data quality, and governance.

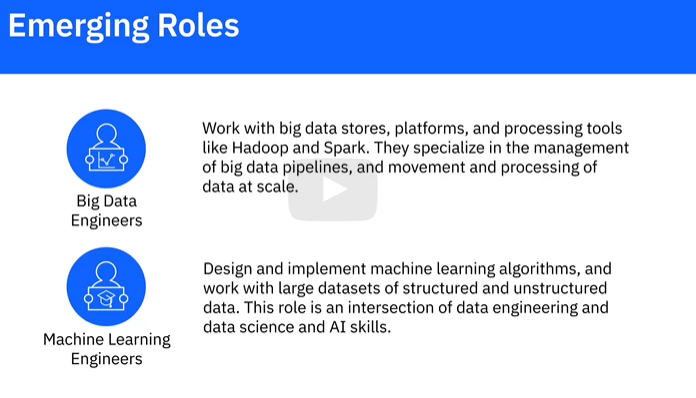
As a data practitioner, using the methodology can help you reduce development time, cut wastages and duplication of effort, increase your productivity and throughput, and ensure that your actions produce the best possible quality of data.

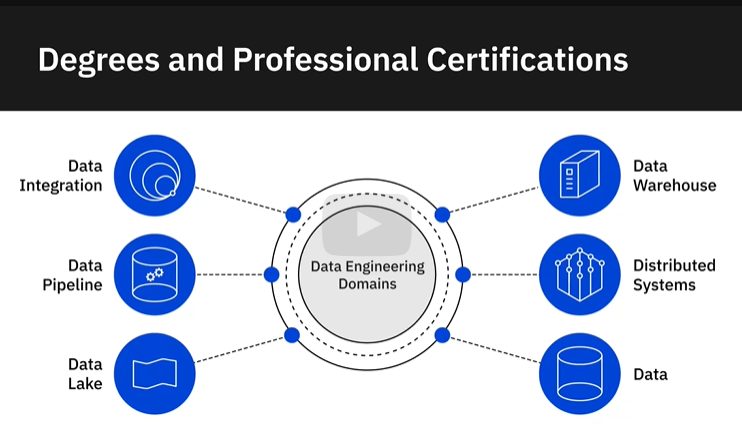
With DataOps, data professionals, consumers, and stakeholders can collaborate more effectively towards the shared goal of creating valuable insights for business. While implementing the methodology will require systemic change, time, and resources, but in the end, it makes data and analytics more efficient and reliable.

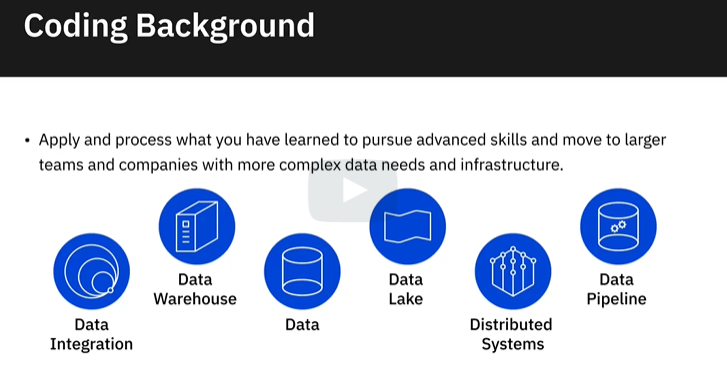
Interestingly, it also opens up additional career opportunities for you as a data engineer. ****DataOps Engineers**** are technical professionals that focus on the development and deployment lifecycle rather than the product itself. And as you grow in experience, you can move into more specialist roles within DataOps, contributing to defining the data strategy, developing and deploying business processes, establishing performance metrics, and measuring performance.

**MODULE 3**









SQL

DATA MODELLING

ETL METHODOLOGIES

PYTHON