Overview of the data analyst Ecosystems

* Data
  + Structured
    - Data that follows a rigid format and can be organized into rows and columns
  + Semi-Structured
    - Mix of data that has consistent characteristics and data that does not conform to rigid structure.
  + Unstructured
    - Data that is complex and mostly qualitative information that cannot be structured into rows and columns.
* File formats
  + Relational Database
  + Non-Relational Database
  + APIs
  + Webservices
  + Data steams
  + Social Platforms
  + Sensor Devices
* Data Repositories
  + Databases, data warehouses, data marts, Data Lakes, Big data Stores
* Languages
  + Query Languages (SQL)
  + Programming Languages (Python)
  + Shell and Scripting Languages

Types of Data

* Structured Data
  + Has well defined structure
  + Stored in well-defined schemas
  + Represented in tabular manner with rows and columns.
  + Source
    - SQL databases
    - Online Transaction processing
    - Spreadsheets
    - Online Forms
    - Sensors GPS and RFID
    - Network and Web server logs
* Semi-Structured Data
  + Some organization properties but lack fix or rigid schema
  + Cannot be stored in form of rows and columns
  + Contain tags and elements, or metadata which is use for grouping and organization in a hierarchy.
  + Sources (XML, JSON)
    - E-Mails
    - XML and other markup languages
    - Binary Executables
    - TCP/IP packets
    - Zipped files
* Unstructured Data
  + Does not easily have identifiable structure
  + Cannot have organized in a mainstream relational database in rows and columns
  + Does not follow any formation sequence semantics or rules.
  + Sources
    - Web pages
    - Social media feeds
    - Images in varied file formats
    - Video and Audio files
    - Documents and PDF files
    - Power Point presentation
    - Media logs.

Understanding Different types of file formats

* Standard file formats
  + Delimited test file formats (CSV)
    - Files use to store data as text
    - Each value is separated by a delimited
    - Delimited – A sequence of one or more characters for specifying the boundary between independent entities or values
    - Comma, Tab, Colon, Vertical Bar, Space
  + Microsoft Excel (XML Spreadsheet OR XLSX)
    - Excel Open XML file format that falls under spreadsheet.
    - Open file format, accessible to most other application
    - Can use to save all functions available in excel
    - Secure files format which means it cannot be used to save malicious code.
  + Extensible Markup Language (XML)
    - Use to set rules for encoding data.
    - Readable by humans and machines
    - Self-descriptive language (Similar to HTML)
    - Doesn’t use predefined tags like .HTML
    - Platform independent
    - Programming language independent.
    - Simplify to share data between systems.
  + Portable Document Format (PDF)
    - Develop by adobe
    - Can be viewed the same way on any device
    - Frequent in legal and financial docs
    - Can be used to fill in data to forms.
  + JavaScript Object Notation (JSON)
    - Most commonly used by APIs and Web Services to return data.
    - Language independent data format
    - Can be read in any programming language
    - Easy to use
    - Compatible with a wide range of browsers
    - Considered as one of the best tolls for sharing data.

Sources of Data

* Relational Databases
  + SQL Server, Oracle, MySQL, IBM DB2
  + Store structured data that can be degrade for analysis.
* Flat File
  + Stored as plain text format
  + Each line, or row is one record
  + Each value is separated by a delimiter
  + All the data in a flat file map to a single table
  + Most seen as a CSV File.
  + Spreadsheet Files
    - Special type of flat files
    - Organize data in tabular format
    - Can contain multiple worksheets
    - XLS or SLSX are common spreadsheet formats.
* XML Datasets
  + Contain data values that are identified or marked up using tags
  + Support complex data structures
  + Common uses include online surveys, bank statements, and other unstructured datasets.
* APIs
  + Application Program Interfaces
  + Web requests
  + Network Requests
  + Returns XML, Plain text, HTML, JSON
  + Popular Examples
    - Twitter and FB APIS
      * For customer sentiment analysis
    - Stock Market APIs
      * For Trading and analysis
    - Data lookup and validation APIs
      * For cleaning and co-relating data.
* Web Services
  + Web Scrapping
    - Extract relevant data from unstructured sources.
    - AKA Screen Scarping, Web Harvesting, and Web data extraction
    - Downloads specific data based on defined parameters.
    - Extracts: Text, Contact info, images, video, product items.
  + Popular Uses
    - Providing price comparison by collecting product details from retailers, manufactures and ecommerce sits
    - Generating Sales through public data sources
    - Extracting data from post and authors on various forums and communities
    - Collecting Training and testing datasets for machine learning.
  + Tools (BeautifulSoup, Scrapy, Pandas, Selenium)
* Data Streams and Feeds
  + Streams of data flowing from instruments, IOT and APPS, GPS, PROGRAMS, Websites, Social Media Posts.
    - Stock and Market Tickets
    - Retain Transactions
    - Surveillance and video feeds.
    - Social Media Feeds
    - Sensor Data feeds
    - Web click feeds.
    - Real time field events.
  + Applications
    - Kafka, Apache Spark, Apache Storm
* RSS (Really Simple Syndication)
  + Online Forums
  + News Sites

Languages for Data Professionals

* Query Languages
  + Accessing and Manipulating data in SQL Server
  + Insert, Update, and delete records in databases
  + Create new databases, tables, and views
  + Write Stored Procedures.
    - Advantages
      * Portable, Platform independent
      * Used to querying data in a wide variety of database and data repositories
      * Simple syntax, Fewer lines of code
      * Can retrieve large amounts of data quickly
      * Runs on an interpreter system.
* Programming Languages
  + Designed for developing applications and controlling application behaviour (Python, R, Java)
  + Open-Source general- purpose high level programming languages.
    - Syntax allows programmers to express their concepts in fewer lines of code
    - Ideal tool for beginners
    - Has in-built functions for frequently used concepts
    - Support multiple programming paradigms, oo, imperative.
      * Advantage (Python)
        + Easy to learn
        + Imperative
        + Functional
        + Procedural
        + Open Source
        + Windows and Linux
        + Widespread community support
        + Open-source libraires for data analysis.
        + Vast array of libraries and functionalities

Pandas for data cleaning and analysis

Numpy and Scipy for statistical analysis.

Beautifulscoup and Scrapy for web scraping

Matplolib and Seaborn to visually represent data in the form of bar graphs, histograms and pie-charts.

* + - * Advantage (R)
        + Developing statistical software
        + Performing data analytics.
        + Creating compelling visualizations.
        + Open Sources
        + Platform-Independent
        + Paris with many programming languages
        + Highly extensible

Libraries

Ggplot 2 and plotly (Graphical plots tools)

Allows data and scripts to be embedded in reports

Creating of interactive web apps

Can be use for developing statistical tools.

* + - * Advantages (JAVA)
        + Top ranked programming languages used today.
        + Used in several data analytics processes
        + Development in big data frameworks and tools – Hadoop, Hive, Spark.
* Shell Scripting
  + Ideal for repetitive and time-consuming operational Task (Unix/Linux Shell, PowerShell)
    - Unix Shell
      * Program written for UNIX shell. Series of UNIX commands written in a plain text file to accomplish a specific task.
      * File manipulation
      * Program Execution
      * System admin task such as back ups
      * Installation scripts for complex programs
      * Executing routine backups
      * Running batches
    - Power Shell
      * Cross-Platform automation tool
      * Configured framework by Microsoft.
      * Optimize for structured data formats
        + JSON, CSV, XML and REST APIs, website, and office apps.
      * Consist of command-line shell
      * Is object-based and can be used to filter sort, measure, group and compare objects.
      * Used for data mining, building GUIS, Creating charts, dashboards, and interactive reports.

**Understanding Data Repositories and Big Data Platforms**

* Repository – Refer to data that has been collected, organized, and isolated.
* Types of repositories
  + Databases
    - Collection of data for input, storage, search, retrieval, and modification of data.
    - DBMS – Data base management system.
      * Using Querying functions to retrieve data.
    - Types of Data base
      * Relational
        + RDBMs

Data organized in tabular format with row and columns

Well defined Structure and schema.

Optimized for data operation and querying.

Examples

Open-Source with internal support

Open source with commercial support

Commercial closed- Source

IBM DB2, SQL SERVER, MYSQL, ORACLE, POSTGRESQL

* + - * + Cloud-Based Relational Databases or (Database as a service)

Amazon RDS, Google SQL, IBM DB2, oracle cloud and azure SQL.

* + - * + Advantages

Create meaningful information by joining tables

Flexibility make change while the database is active.

Reduce redundancy allow data to be defined between multiple different tables

Easy of backup and recovery. (Continues mirroring)

ACID (Atomicity, Consistency, Isolation, Durability)

* + - * + Use cases

Online Transaction processing application (OLTP)

Accommodate large number of uses

Manage small amount of data.

Support frequent queries.

Data Warehouses

Optimized for online analytical processing (OLAP)

Iot Solutions

* + - * + Limitation

Does not work well with semi-structured and unstructured data

Migration between two RDBMS’s is possible only when the source and destination tables have identical schemas and types.

Entering a value greater than the defined length of a data field results in a loss of information.

* + - * Non-Relational
        + NoSQL

Emerged to the volume, diversity, and speed at which data is being generated.

Built for speed and flexibility and scale

Data stored in schema-less form

Widely used for big data.

Ability to handle large volumes of data.

Distributed system scaled across multiple data centre.

Efficient and cost effective

Ease of use.

Types of NoSQL Databases

Key values store

Stored as a collection of key values pairs.

Key is a unique identifier. (Integers or complex strings)

Great for storing user session data, preferences, real time recommendations, targeted advertising, in-memory data caching.

Limitation (Not a good fit if)

query data on specific data values

Need relationships between data values

Need multiple unique keys

Key- Value store

Redis, Memcached, DynamoDB

Document Based

Document-Based

Stored each record and its associated data within a single document.

Used primarily for ecommerce, med record, CRM, and analytics platforms.

Limitations (Not a good fit if)

Run complex search queries

Perform multi-operation transactions.

Types

MongoDB, DocumentDB, CouchDB, Cloudant

Column Based

Stored in Cell groups as columns or data.

A logical grouping of columns is referred to as a column family.

Continues disk entry allows for making access and search easier and faster.

Great for heavy write request, storing time-series data, weather data, and IoT data.

Limitation

Run complex queries

Change Querying patterns frequently.

Types

Cassandra, Apache HBASE

Graph Based

Graphical model to represent data.

Use for visualizing, analysing, and finding connections between different pieces of data.

Good for working with connected data.

Used in Social networks, Product recommendations, Network diagrams, fraud detections.

Limitation

Process high volume of transactions.

Types

Neo4j, CosmosDB

Text

Description automatically generated with low confidence

* + Data Warehouses
    - Works as a central repository that merged info from dispel sources.
    - Works like a multipurpose storage for difference use cases.
    - Analysis READY
    - ETL Process – Consolidates data through the extract, transform, and load process, also known as the ETL process in to one comprehensive database for analytics and business intelligence.
      * Extract data from various sources
      * Transform data to clean etc
      * Load data into a repo.
    - Data Mart
      * Build as a sub section of a Data Warehouse
      * Dedicated to only provide data from the data warehouse that is most related to the client when they need it.
      * Provide analytical capabilities in a restricted area
      * Isolated Security and Isolated performance.
      * Businesses specific reporting and analytics.
    - Data Lakes
      * Pool of raw data where each data element is given a unique identifier and is tagged with metatags for further use.
      * Selected and organized based on the use cases.
    - Extract, Transform and Load Process (ETL)
      * Automated process
      * Gathering raw data
      * Extracting info for needed for reporting and analysis.
        + Batch Processing – Larger chunks of data moved to destination at scheduled intervals.

Tools – Stitch and Blend

* + - * + Stream Processing – real time data from source, transformed in transit and loaded into data repository.

Tools – Samza, Storm, Kafka.

* + - * Cleaning, standardizing, and transforming data into usable format.
        + Standardizing

Data formats

Units of Measure

* + - * + Removing Duplicate
        + Filtering out data
        + Enriching data.

Spiting full name to first and last.

* + - * + Establishing key relationship across tables
        + Applying business rules and data validations.
      * Loading data into a data repository.
        + Transportation of process data into repo.
        + Initial loading – Populating all the data into the repository.
        + Incremental loading – Applying updates and modification periodically.
        + Full refresh- Erasing data table and reloading.
        + Load Verification checks

Missing or null values

Server performance

Load failures.

* + - * Data Pipeline
        + Moving data from one system to another
        + Use for bath and streaming data.
        + Support both long running batch queries and smaller interactive queries.
        + Loads data into a data lake but also load data into a variety of target destinations such as apps and visualization tools.
  + Big Data Stores
    - Distributed computational and storage infrastructure to store, scale, and process very large data sets.
* Factors that govern which databases to use
  + Data type
  + Data Structure
  + Querying Mechanisms
  + Latency Requirements
  + Transaction Speeds
  + Intended use of data.
* Big Data
  + VELOCITY,
    - Speed in which data accumulates.
  + VOLUME,
    - Scale of the data.
      * Increase in data sources,
      * High resolution sensors
      * Scalable infrastructure.
  + VARIETY,
    - Diversity of the data.
    - Comes from different sources. Machines people etc
    - Mobile tech
    - Social media
    - Wearables
    - Geo Tech
    - Video
  + VERACITY,
    - Quality and origin of data.
    - Attributes
      * Consistency
      * Completeness
      * Integrity
      * Ambiguity
    - Cost and need for traceability
  + VALUE
    - Ability and need to turn data into value.
      * Medical or social benefit.
  + Tools
    - Hadoop (Java Base)
      * Distributed storage
      * Processing Big Data
      * Reliable, scalable, and cost-effective solution for storing data with no format req.
      * Better real time data driven decision
      * Improved data access and analysis
      * Data offload and consolidation.
      * Hadoop Distributed File System (HDFS)
        + Storage system running on multiple hardware connected through a network.
        + Scalable and reliable big data storage by partitioning files over multiple node.
        + Splits large files across multiple system allowing for parallel access.
        + Replicate file block to prevent loss.
        + Fast recovery from hardware failures
        + Access to streaming data
        + Accommodation of large data set
        + Portability.
    - Hive
      * Data warehouse for data query and analysis
      * Store either in HDFS or other data storage systems.
      * Queries have high latency (not ideal for fast response time)
      * Not suitable for transaction processing
      * Used for ETL, Reporting and DA.
      * Easy access to data via SQL
    - Spark
      * Analytics framework design for complex real time data analytics.
      * General purpose data processing engine
      * Interactive analytics
      * Steams processing
      * Machine learning
      * Data integration
      * In-memory processing (Increase speed)
      * Interfaces for major programming languages
      * Standalone clustering tech
      * Run on other infrastructures such as Hadoop.
      * Access data in large variety of data sources

Summary

A Data Repository is a general term that refers to data that has been collected, organized, and isolated so that it can be used for reporting, analytics, and also for archival purposes.

The different types of Data Repositories include:

* Databases, which can be relational or non-relational, each following a set of organizational principles, the types of data they can store, and the tools that can be used to query, organize, and retrieve data.
* Data Warehouses, that consolidate incoming data into one comprehensive storehouse.
* Data Marts, that are essentially sub-sections of a data warehouse, built to isolate data for a particular business function or use case.
* Data Lakes, that serve as storage repositories for large amounts of structured, semi-structured, and unstructured data in their native format.
* Big Data Stores, that provide distributed computational and storage infrastructure to store, scale, and process very large data sets.

ETL, or Extract Transform and Load, Process is an automated process that converts raw data into analysis-ready data by:

* Extracting data from source locations.
* Transforming raw data by cleaning, enriching, standardizing, and validating it.
* Loading the processed data into a destination system or data repository.

Data Pipeline, sometimes used interchangeably with ETL, encompasses the entire journey of moving data from the source to a destination data lake or application, using the ETL process.

Big Data refers to the vast amounts of data that is being produced each moment of every day, by people, tools, and machines. The sheer velocity, volume, and variety of data challenge the tools and systems used for conventional data. These challenges led to the emergence of processing tools and platforms designed specifically for Big Data, such as Apache Hadoop, Apache Hive, and Apache Spark.