Big Data Technologies

Chapter 05

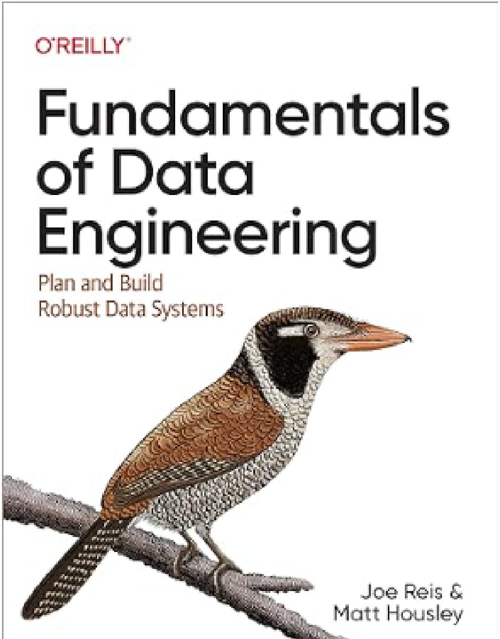
Fundamentals of Data Engineering

Data Generation in Source Systems

The Data Engineering Lifecycle in Depth

- Part II
    - Includes Chapters 5-9
    - Data Generation in Source Systems
    - Storage
    - Ingestion
    - Queries, Modeling, and Transformation
    - Serving Data for Analytics, ML, and Reverse ETL

Text Book



O'REILLY

Fundamentals of Data Engineering

Plan and Build Robust Data Systems

Joe Reis & Matt Housley

Objectives

- Discuss operational source patterns and the significant types of source systems
- Explain the considerations of a data generating system
- Discuss how the undercurrents of data engineering apply to this first phase of the data engineering lifecycle
- Explain opportunities to build customer facing systems
- Discuss the integration of data engineers and source system teams

Outcomes

At the conclusion of this lecture and lab you will have discovered how source systems are vital to data engineering lifecycle. You will also have learned that better collaboration with source systems teams leads to better outcomes.

Review Chapter 1

- List the 5 stages of the Data Engineering lifecycle from Chapter 1?
- List the 4 technologies every data engineer should be familiar with?
- List the 5 business responsibilities of a data engineer
- List the 6 undercurrents to the Data Engineering lifecycle
- Which came first, Relational Database Model or SQL?

Review Chapter 2

- Define data life cycle management
- Discuss a Data Engineers relationship to business objectives

Review Chapter 3

- Defined data architecture
- Explain how data architecture sits fundamentally at the core of a business
- Explain 3 of the current data architectures

Review Chapter 4

- Discuss some of the trade-offs of using Opensource Software
- Explain the ideal timeframe for how far to look into the future when making tech decisions
- Explain the concept of TCO
- Explain the concept of TOCO
- Explain the concept of Interoperability

Data Engineering Lifecycle

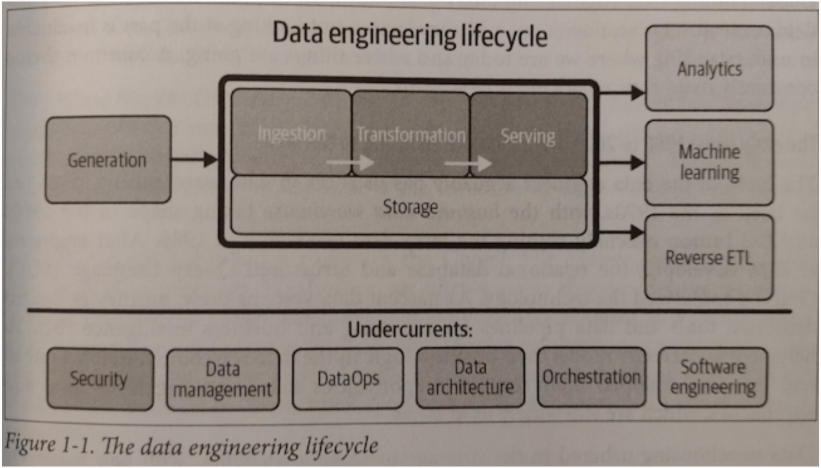


Figure 1-1. The data engineering lifecycle

Figure 2: *Figure 7.1*

Sources of Data: How is Data Created?

- Analog data
- Digital data

Source Systems: Main Ideas

- Structured and Unstructured Data
  - text files and XML
  - CSV
  - JSON
  - Parquet
  - Logs
  - Images
  - Audio files, mp3

Source Systems: Main Ideas

- APIs
  - A standard way of exchanging data
  - Available over the web
  - Weather API

Application Databases (OLTP Systems)

- Online Transaction Processing
- Databases for e-commerce are OLTP
    - Reads and writes at a high rate
    - Sometimes just called Transactional databases
    - You are transacting with the website
    - Support low latency and high concurrency
    - Places like YouTube or Twitter or Facebook use OLTPs

Application Databases

- ACID
  - Consistency means data base will return the last item written for a read
  - Isolation determines that if two writes are competing, the last sequential write will be the source of truth
  - Durability means data won't be lost even on a power failure
- These guarantee that the database will always have a consistent picture of the world
  - Developers need to know this to design applications
- Atomic actions
  - SQL queries either run completely or fail completely
- Applications don't need to have ACID based systems behind them

OLAP - Online Analytical Processing System

- OLAP
  - Built to run large analytics queries
  - Inefficient at running lookups and transactions
  - Spark wouldn't be a good OLTP

Change Data Capture

- CDC executes replication when a change event occurs
    - Update, Insert, or Delete
    - Logs are generated on the system showing what happened

Logs

- Logs capture information about events the occur in a system
  - Operating systems
  - Applications
  - Servers
  - Containers
  - Networking equipment
  - IoT and AI/ML
- Logs track Who, What happened, and When

Log Resolution

- Keep everything had been the mantra
  - Sometimes legal requires this
- With compression and binary formats
  - We can reduce the amount of storage
  - Also we need to analyze the frequency of logs
  - As time goes on we don't need the very fine grained collection
  - We can drop logs over time
- Severity levels of logging collection can be configured

Database Logs

- CRUD
  - Create, Retrieve, Update, and Delete
  - This sequence generally is how data comes into existence

Messaging and Streams

- Related to Event-driven architecture
    - Message queues
        - A message is raw communicated data between two systems
        - From publisher to consumer
        - Gaurenteed delivery and deleted from queue once done
        - Amazon Simple Queue Service
    - Streaming platforms
        - An append-only log of event records
    - Apache Kafka
        - Append only means events are stored over a long window of time
        - Can be analyzed

Types of Time

- Event time
    - When something happens
    - Generally stored in UTC time
- Ingestion Time
    - When event data is taken into storage of some kind
- Process Time
    - Occurs after ingestion time
    - How long to process the data
- What would latency be in this case?

Source System Practical Detail

- Major Considerations for DB Technology
  - Database management system
  - Lookups
  - Query Optimizers
  - Scaling
  - CRUD
  - Consistency

Relational Databases

- Oracle, MySQL, MSSQL, and PostgreSQL
    - Data is stored in a table of relations
        - Rows
    - Each relation contains multiple fields
        - Columns
    - Each relation has a `schema`
    - Tables index by a `primary key`
    - Tables can have `foreign keys`
    - Tables are ACID

Non-Relational Databases - NoSQL

- Abandon the relational paradigm
    - Where did the idea come from (turn of century and turn of decade)?
    - Was the problem SQL or the RDBMS software
- Key Value storage
    - Retrieves records using a single key that identifies unique records
    - Key can also be used to partition data and store relevant items together
- Document stores
    - Collections are conceptually like a table
    - Documents are similar to records
    - Individual *fields* are called items and the values attributes
    - Stored as a JSON object
    - No relations so much duplication of data

Graph Databases

- Store data in a mathematical graph structure
    - A set of nodes and edges
    - Good for analyzing connectivity between elements
    - Facebook or Microsoft make use of the `graph`
        - To figure out who you are related to or what items you might want to view

Search Database

- Used just for searching for text strings
    - Email
    - Document search

Time Series Databases

- All data events stored by timestamp
  - Timestamp and event data
  - Used for optimizing retrieval of data for graphing
  - For statistical processing of time-series data

APIs

- REST APIs
- GraphQL
  - Facebook created
- WebHooks
  - Using post data to trigger an event elsewhere
- RPC and gRPC
- Third-Party Data Sources

Whom you'll work with

- Data Sources might have a contract or you need to provide a contract
- The 6 undercurrents impact the system
  - Security
  - Data Management
    - Data governance
    - Data quality
    - Schema
    - Master Data Management
    - Privacy and ethics
    - Regulatory

Whom you'll work with

- DataOps
    - Automation
    - Observability
    - Incident Response
- Data Architecture
    - Understand the upstream architecture
    - Reliability
    - Durability
    - Availability
    - People

Whom you'll work with

- Orchestration
- Software Engineering

Conclusion

- Source systems are vital in the data engineering lifecycle
- Better collaboration with source systems produces better results
- Discussed the event-driven architecture paradigm
- Talk to internal and external users about their data needs

Additional Resources

- See page 192 for additional readings

Homework

- Read FDOE chapter 6

Questions?

- Any questions?
    - Discord always open