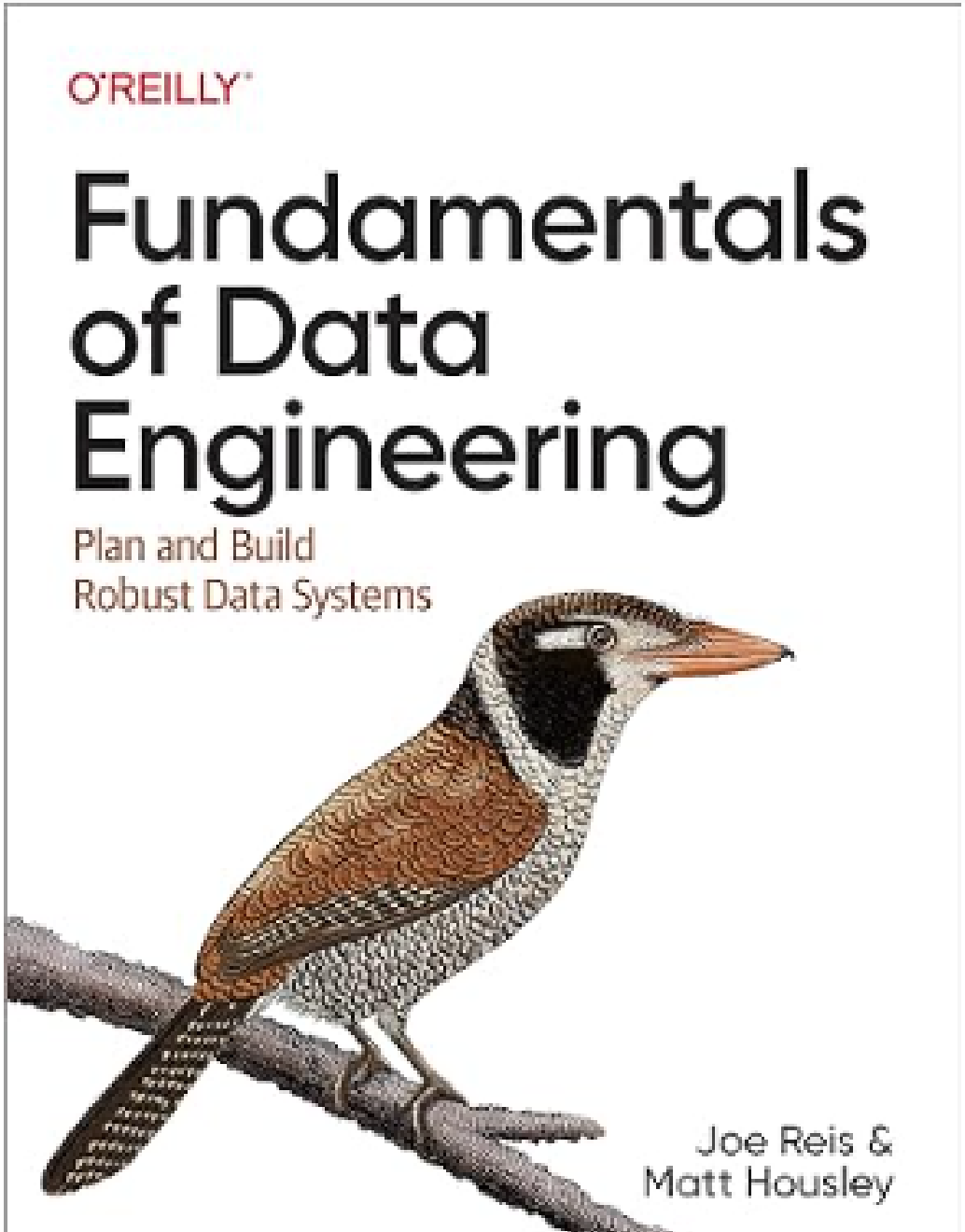
Big Data Technologies

Chapter 02

Fundamentals of Data Engineering

Data Engineering Lifecycle

Text Book


O'REILLY
Fundamentals of Data Engineering
Plan and Build Robust Data Systems
Joe Reis &
Matt Housley

Objectives

- Discuss and define data life cycle management
- Discuss and Engineers relationship to the technology stack and its abstractions
- Describe how to organize work as a Data Engineer
- Describe the Data Engineering Life Cycle and its 5 stages
- Define the 6 undercurrents of the Data Engineering Life Cycle
- Discuss a Data Engineers realtionship to business objectives

Outcomes

At the conclusion of this lecture and lab you have reviews the Data engineering lifecycle, examined its undercurrents and examined the how a data engineer and business objects relate to each other.

Review

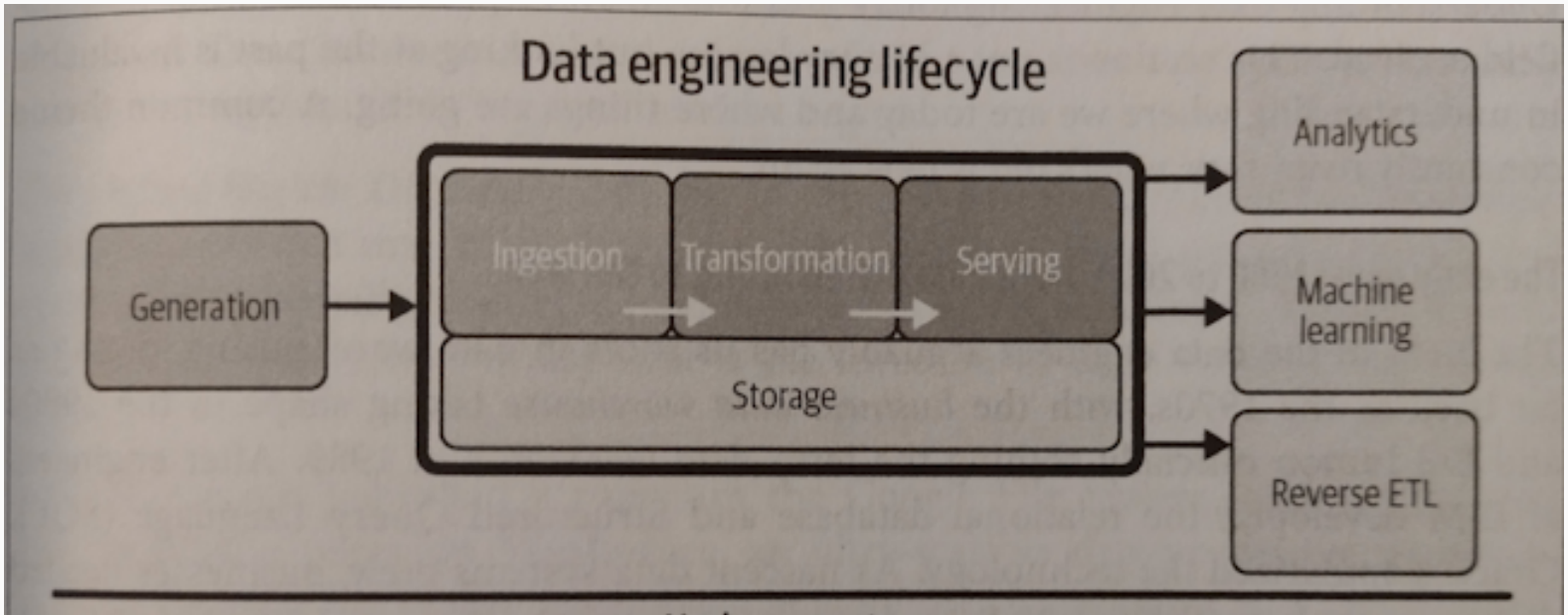
- List the 5 stages of the Data Engineering lifecycle from Chapter 1?
- List the 4 technologies every data engineer should be familiar with?
- List the 5 business responsibilites of a data engineer
- List the 6 undercurrents to the Data Engineering lifecycle
- Which came first, Relational Database Model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

Thinking in terms of data

- We want to encourage yuo to move beyond viewing data engineering as a collection of specific technologies
  - Landscape changes too much and too often
  - Hyperscalars (Public Cloud)
  - Startups and VC funded companies
- Move you from a Data Engineer to a Data Lifecycle Engineer
  - Lifecycle turns raw data into a useful end product
    - Key that it is ready to be consumed by Analysts, ML, and so forth

Data Engineering Lifecycle

- Generation
- Storage
- Ingestion
- Transformation
- Serving Data



Data Lifecycle vs Data Engineering Lifecycle

- A subtle distinction
- Data Engineering Lifecycle is a small subset of the Data lifecycle
- Image form fig 2.2 goes here

Generation: Source Systems

- Where data is *generated*
  - IoT device
  - Car
  - Point of Sale system
  - Data Warehouse
  - Database
  - RDBMS and NoSQL clusters
  - You viewing a webpage (analytics)

Know thy data

- Wnat to know the characteristics of the data
  - Is it reliable?
  - What kind of errors can happen/do happen?
  - Will it be sequential or out of order?
  - Are there upstream dependencies?
  - Any checks to make sure bad data doesn't come to you?
  - Does the data have a schema?
  - Schema is column data type enforcement
  - Table relations
  - Schema on read data?

Storage

- You need to store data
  - Cloud Native - Amazon S3
    - Object storage
  - Hard Disk
- How to control access to this data?
- Just storing or does the storage have any analysis features?
- Schema agnostic?
  - Object storage has no schema
  - Text file has no schema
- Regulatory compliance?

Data Frequency

- Hot data or cold data?
- What will be the retrieval pattern

Ingestion

- Can be most significant bottle neck
  - What if you have petabytes of data to transfer?
  - All work can stop here waiting on this step
- What will be the output destination after ingestion?
  - Is the data from a streaming format?
  - Do I need to convert things?
- Do I need a permanent conversion phase?

Batch vs Streaming

- Streaming allows us to provide data to downstream systems
  - In a continuous real-time fashion
  - Latency to consider
- Batch is ingested at a pre-determined interval
  - Threshold of size or time usually
  - Machine Learning uses batch usually
- Consider flow rate of streaming – can downstream handle it?
- How fast do I need to generate data? Millisecond?
- What is my benefit of streaming setup vs batch?

Push vs Pull

- How will you transfer data
  - Push it to the next phase?
  - Have that next phase pull it from you?
    - ETL (Extract)
    - Genearally as a batch

Transformation

- After we have the data ingested we might want to transform it...
  - What is ROI?
  - Is it self-contained?
  - Who provides the Business Rules to tell me what to transform and how

Serving Data

- Data only has value when it is presented for a practical purpose
  - Reports
  - Triggers for other actions
  - Endpoints for other scientists to consume

Serving Data Areas

- Areas such as. . .
  - Analytics
    - Operational or Inventory
    - Business Intelligence
    - Embedded or Customer facing reports and insights
  - Machine Learning (ML)
  - Reverse ETL
    - Feeds *cleaned* data back into the source input system
    - Push metrics to a third party sales forcasting system
    - Reinjest the results

Major Undercurrents

- figure 2.7

Security

- Principle of least privilege
  - People and policies are biggest security threat
  - Data Security via timing
    - Employees leave or are transferred
    - What happens to their access?

Data Management

- Sounds very corporat...
  - Important body of knowledge
  - Best practices that has been around for decades
  - Data is vital to the company's ability to operate
- Elements such as...
  - Data governanace (accountability and discovery)
  - Data modeling and design
  - Storage
  - Integration
  - Privacy (GDPR)

Data Governance

- *"Data governance is, first and foremost, a data management function to ensure the quality, integrity, security, and usability of the data collected by an organization"*
- The organization creates rules and policies for managing data
- Discoverabilty
- Metadata
  - Data about data
  - Business metadata
  - Technical metadata
    - Schema
  - Operation metadata
  - Reference metadata
    - Regional codes

Data Accountability

- Who is responsible for maintaining the data?
  - *"We all are!"*
  - This means that no one is responsible

Data Quality

- *"Can I trust the data?"*
  - Quote from everyone in the business
  - Is it accurate?
  - Is it complete?
  - Is it timely?

Data Modeling and Design

- Process of converting data into a usable form
  - Need to understand Data Lineage
    - Which system is impacted as data moves?
    - Helps with error tracking and accoutnability

Data Integration and Interoperability

- Making sure your data works with your tools
  - There is a reason that every system on the planet can export to CSV file
- Increasingly interaction with systems happen via API over HTTP
  - Less manual moving and copying things
  - But still exists out there

Data Lifecycle Management

- Put it all in the cloud!
  - Save money!
- Not so fast, now you have data sovereignty laws
  - GDPR and other protection laws
  - Cloud costs (not cheap!)
- Is there ever a time you can/should destroy data?
  - Outside of commiting a crime
  - Is it even legal to do so?

How to Orchestrate all of this

- Perhaps you inherit a data pipeline
  - Perhaps it is well orchestrated
  - What if you have to create your own?
- This is the concept of orchestration
  - Key comptetive advantage if your orchestration pipeline
    - Smooth
    - Fast
    - Reliable
    - Secure

Additonal Resources

- See page 71 and 72 of chapter two for additional readings

Homework

- Read FDOE chapter 3 and 4

Conclusion

- We defined data engineering and the data lifecycle
- We described our data maturity model
- We described Type A & B engineers
- We described the DEs place in the corporate structure

Questions?

- Any questions?
  - Discord always open