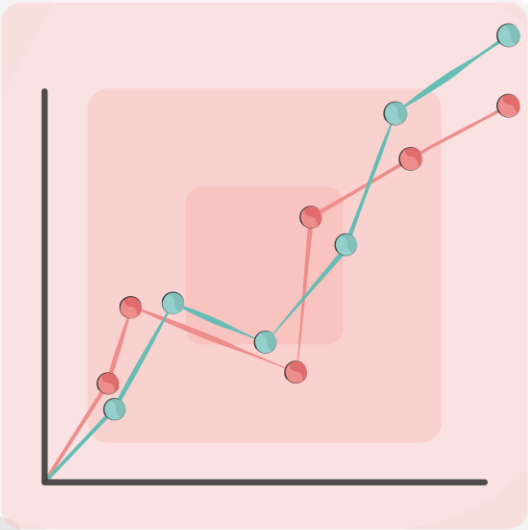
Data engineering

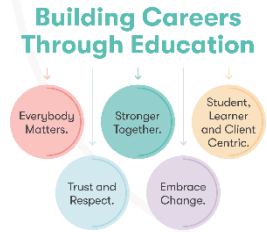
Module: Data pipelines

Topic: Data integration techniques



Welcome to today's webinar.

Ice breaker



Discussion…

- How are you feeling today? Motivated, happy etc?
- What is your key takeaway from the e-learning topic?
- What is one key skill or insight you hope to gain from today's session?

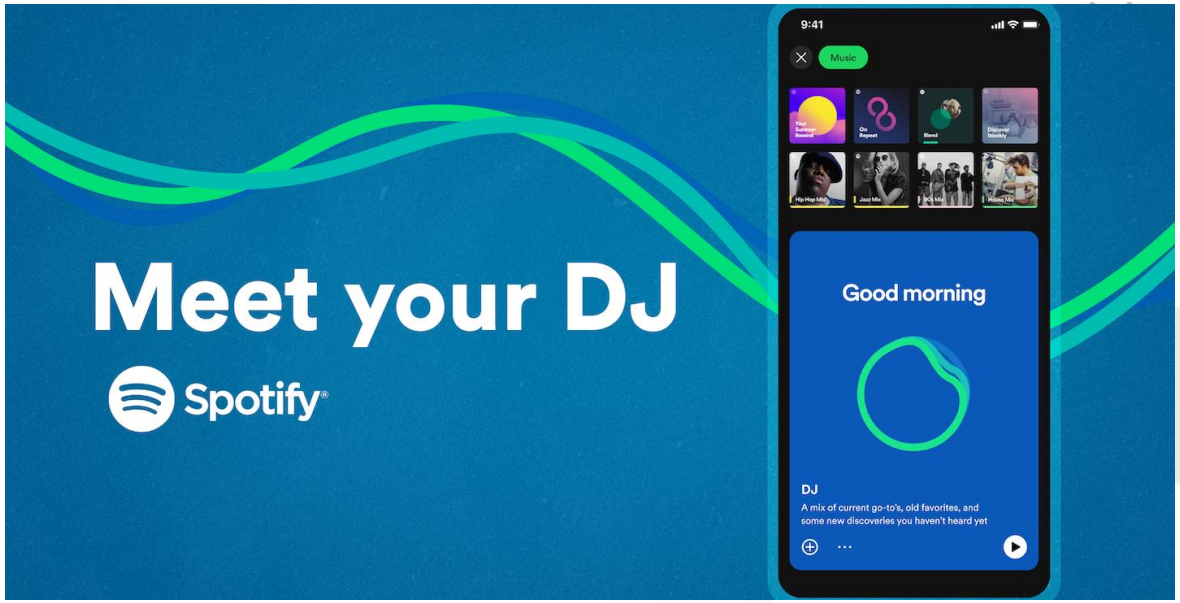
Submit your responses to the chat or turn on your microphone

BPP

Spotify’s Scalable Data Platform

Streaming, Cloud-Native, and Insight-Driven Integration…

- Migrated 1,200+ services and 20,000+ daily jobs to Google Cloud
- Uses Pub/Sub, Dataflow, and BigQuery for real-time analytics
- Supports billions of daily plays with low-latency data pipelines
- Enables personalised recommendations and artist insights
- Focuses on developer velocity and scalable, hybrid architecture



The Spotify platform, ***Image source:*** *Spotify*

e-learning recap

Reflecting on your learning…

The e-learning for this topic, covered the following areas:

- **ETL** – Too slow for modern needs
- **Methods** – Flexible access via ELT, federation, etc.
- **Streaming** – Real-time insights
- **Architecture** – Simple, centralised, or scalable
- **Tools** – NiFi, Kafka, dbt, Talend, Airflow
- **Governance** – Validate and secure data
- **Hybrid** – Mix methods for resilience

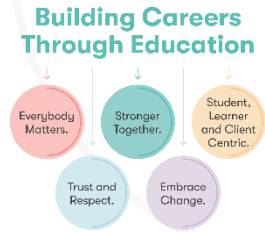
- Do you have any questions about any of these areas?
- Did everything in the e-learning make sense?

Webinar Agenda

Today, we will cover the following:

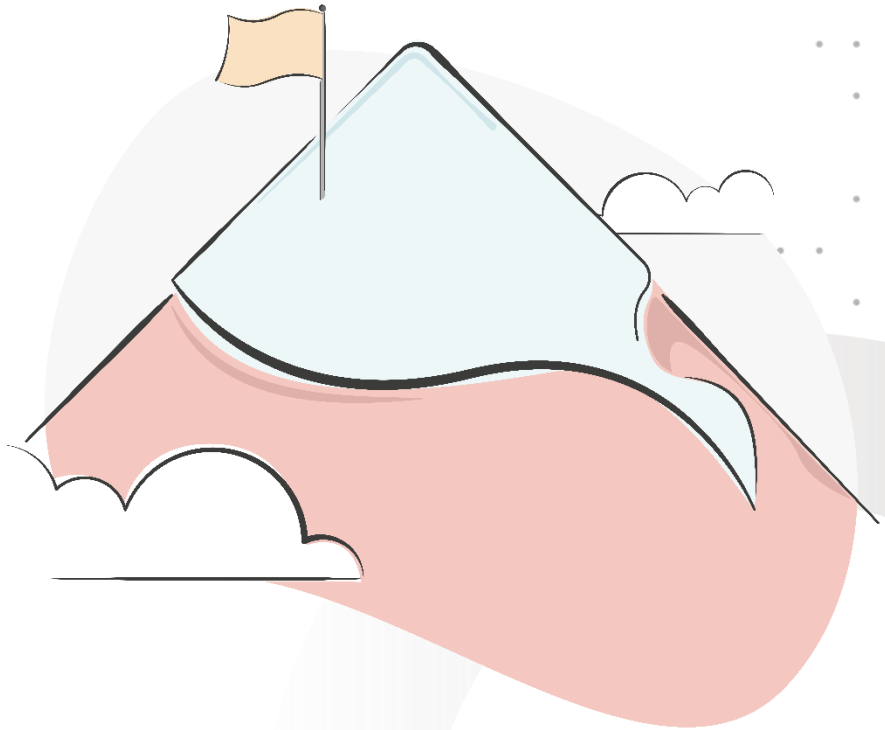
1. Welcome and Intro
2. Recap of core concepts
3. Practical lab
4. Summary
5. Q&A

Session aim and objectives



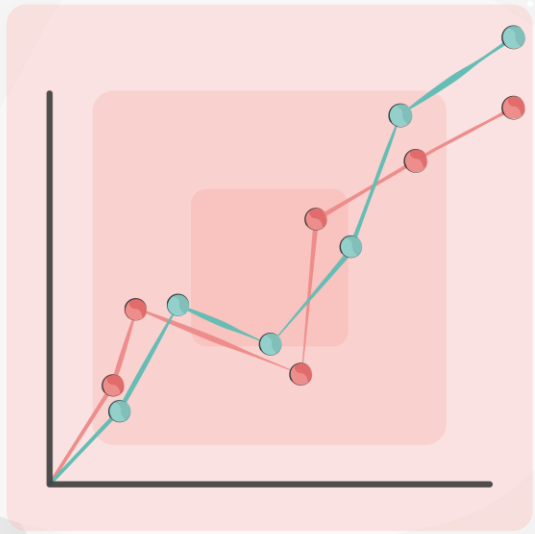
By the end of this session, you should be able to:

- Understand advanced integration methods beyond traditional ETL.
- Explore real - time and federated data integration approaches.
- Evaluate modern tools used in advanced integration scenarios.



BPP

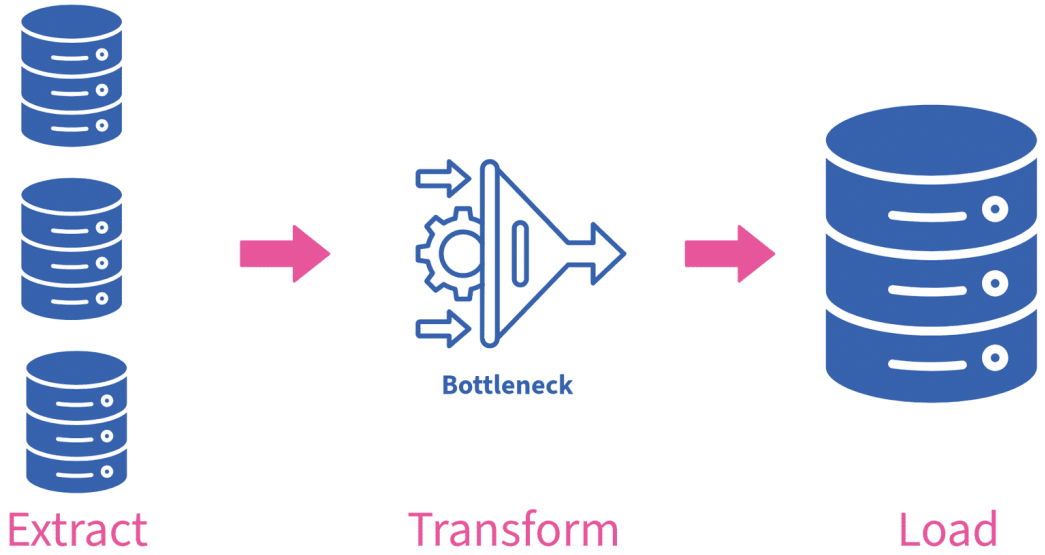
Recap of core concepts



Why traditional ETL falls short

The limits of batch-based thinking

- ETL is slow, rigid, and batch-oriented
- Poor fit for real-time or streaming data
- Struggles with schema changes and unstructured formats
- Centralised transformations create bottlenecks



From Rigid ETL to Modular Data Pipelines:
Embracing Flexibility and Orchestration

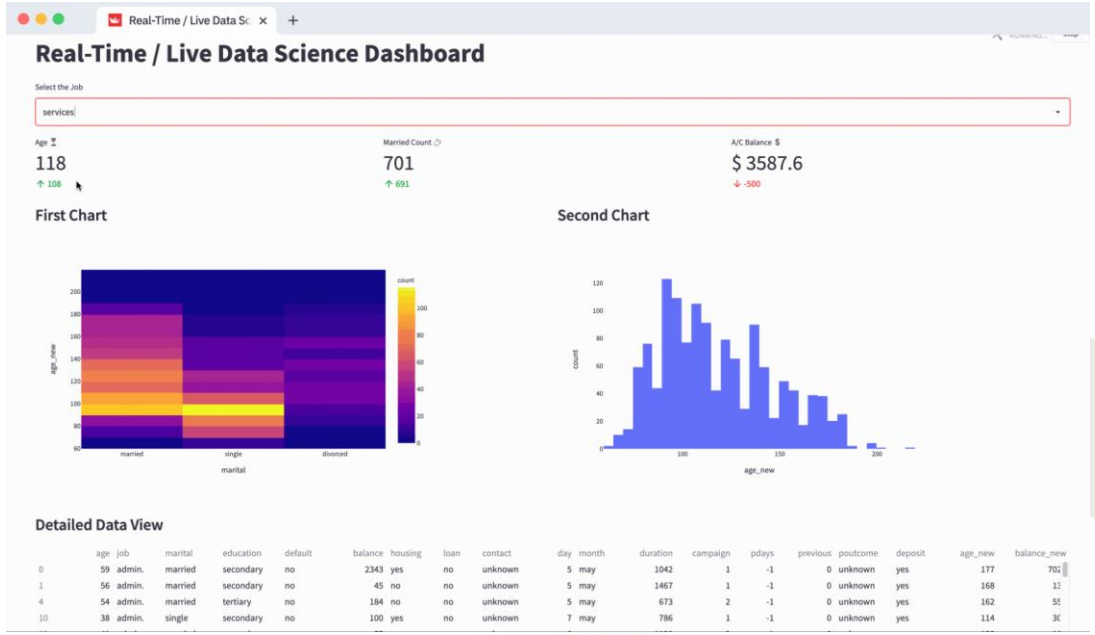
Modern integration methods

Building Careers Through Education

Everybody Matters. Stronger Together. Student, Learner and Client Centric. Trust and Respect. Embrace Change.

Flexible approaches for diverse needs

- **Federation:** Query data without moving it
- **Virtualisation:** Standardise views across sources
- **Blending:** Combine data quickly for specific use cases
- **ELT:** Load first, transform later for speed and scale

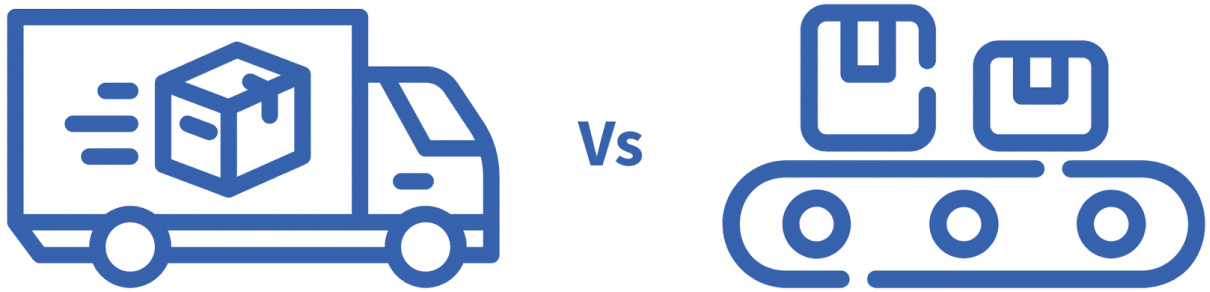


Data federation example: A dashboard pulling live data from multiple APIs, image source: Streamlit

Real-time streaming pipelines

From snapshots to continuous flow

- Streaming enables low-latency, event-driven processing
- Ideal for dashboards, alerts, and automation
- Supports real-time sources like IoT, logs, and APIs
- Requires careful design: ordering, state, fault tolerance



Batch pipeline

Like a postal truck doing pickups and deliveries once a day.

Streaming pipeline

Like a high-speed conveyor belt: always on, always moving.

An analogy for the difference between batch and streaming pipelines

BPP

Integration architecture options

Designing for scale, resilience, and control

- **Point-to-Point**: Simple but hard to scale
- **Hub-and-Spoke**: Centralised control and governance
- **Distributed**: Modular, fault-tolerant, cloud-native
- Architecture impacts flexibility and maintainability

Conider your own organisation’s needs—compliance, scale, agility—how these influence architectural decisions?

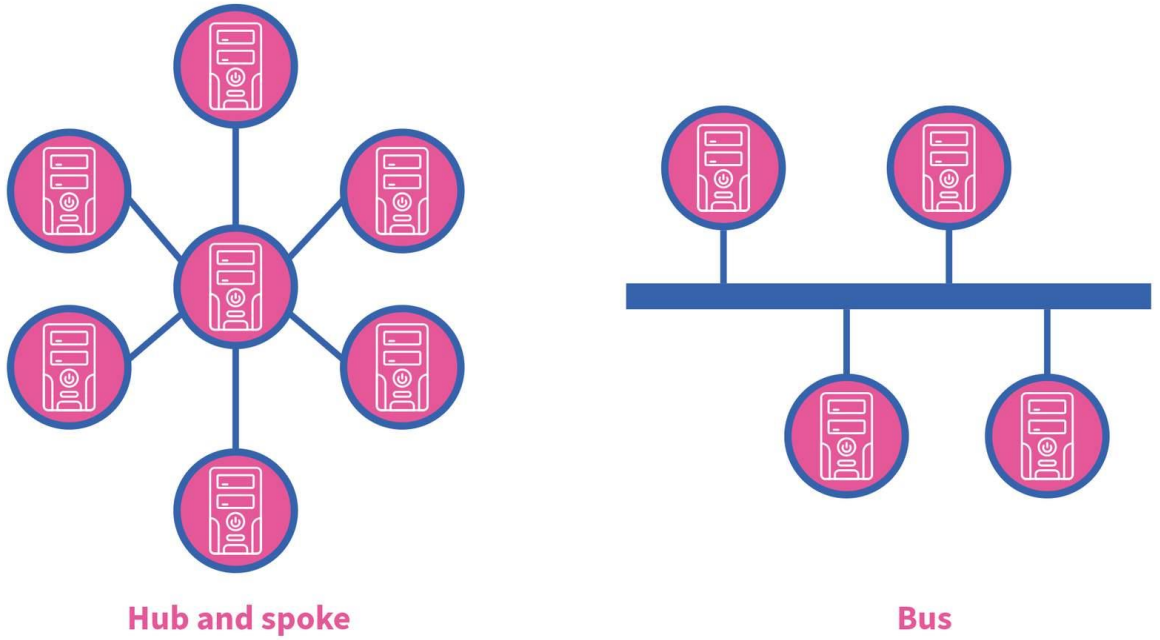


Figure: *Hub-and-Spoke versus Bus Architecture*

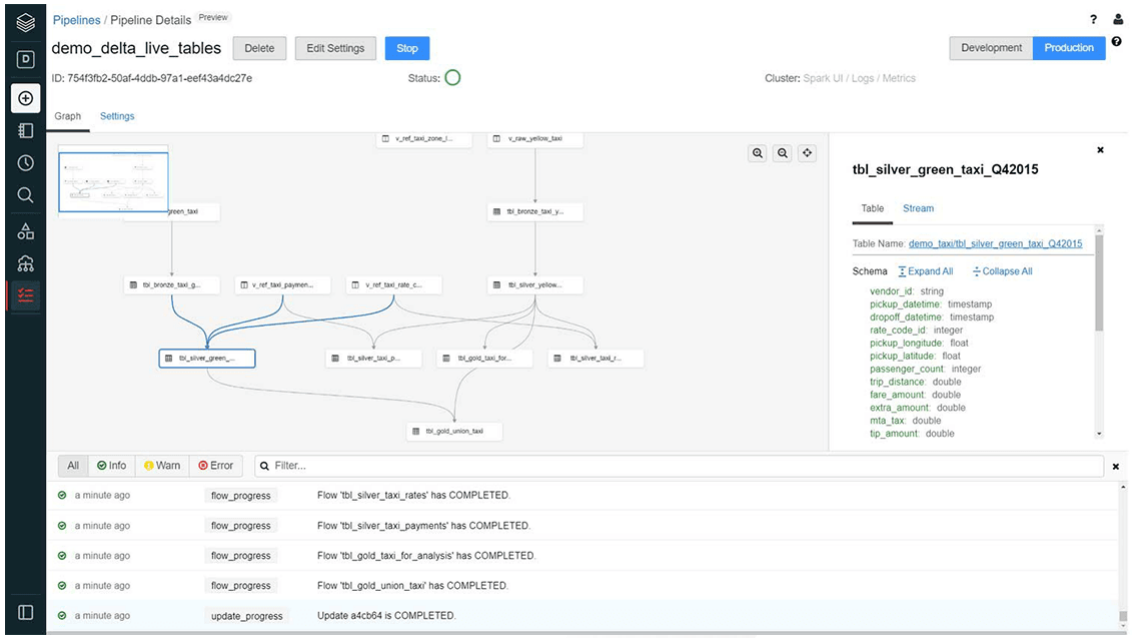
Tooling across the pipeline

Building Careers Through Education

Everybody Matters. Stronger Together. Student, Learner and Client Centric. Trust and Respect. Embrace Change.

The right tools for the right stage

- **NiFi:** Ingestion and routing
- **Kafka:** High-throughput streaming backbone
- **dbt/Talend:** Transformation and data quality
- **Airflow/Prefect:** Orchestration and scheduling



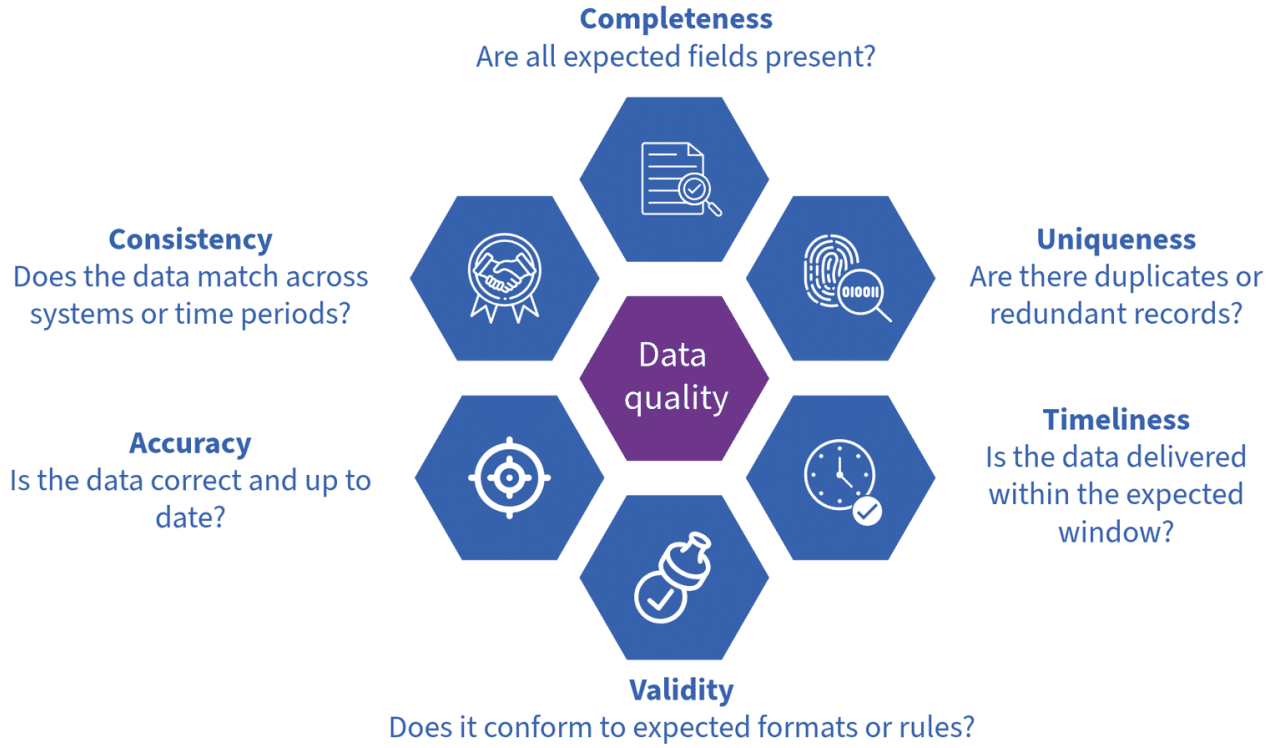
Modern Data Pipeline: *Right Tools at Every Stage*

BPP

Data quality is non-negotiable

Trustworthy data, reliable decisions

- Poor data = poor decisions and wasted effort
- Key dimensions: accuracy, completeness, consistency
- Pipelines must validate and monitor data continuously
- Quality issues can scale quickly if unchecked



Key Attributes of Data Quality in the Pipeline

Built-in governance

Visibility, Control, and Compliance

- Track data lineage and transformations
- Enforce schema validation and access control
- Log data movement and changes
- Governance is embedded, not optional

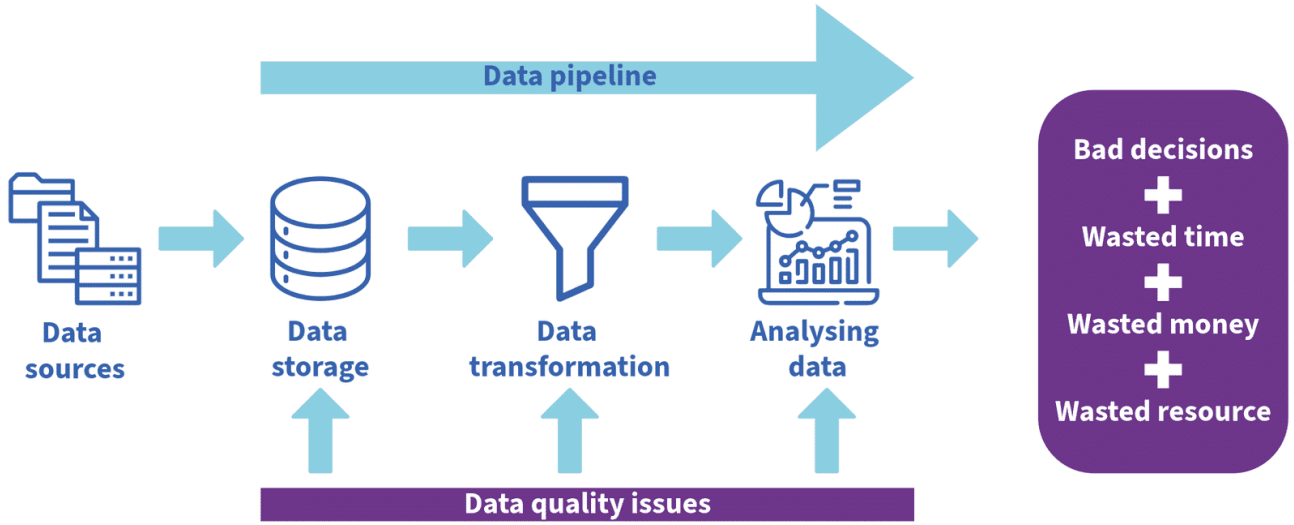
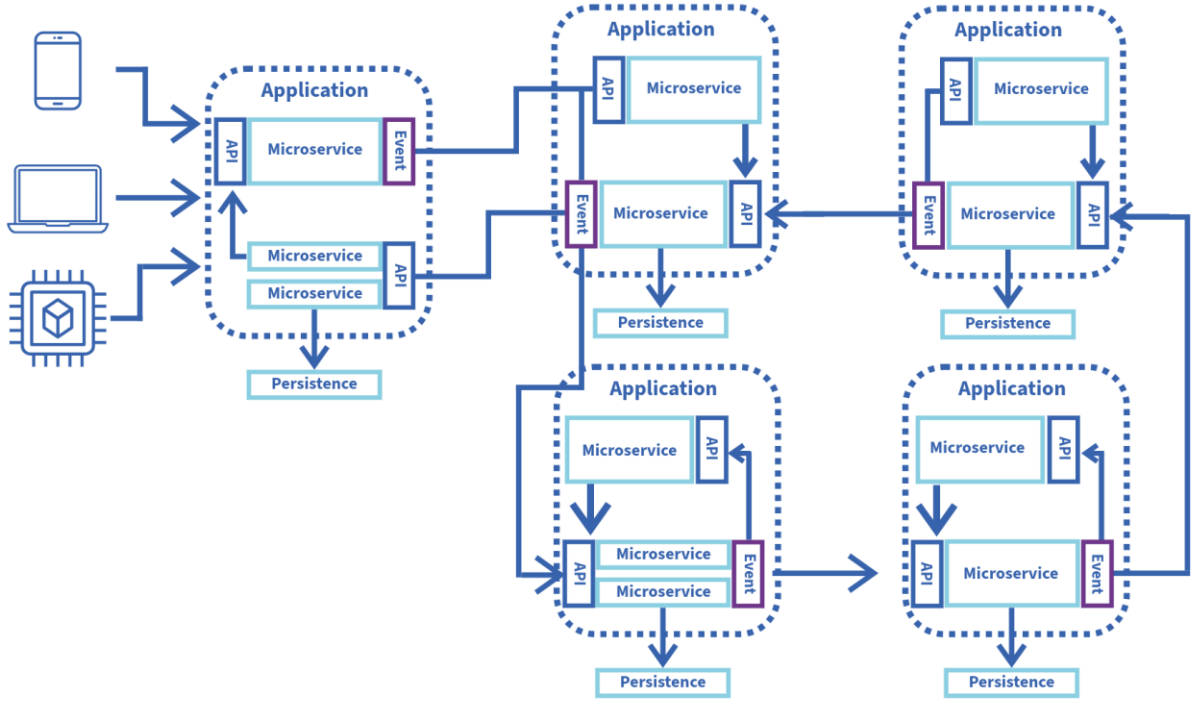


Figure: Data quality issues in pipelines

Hybrid, modular design thinking

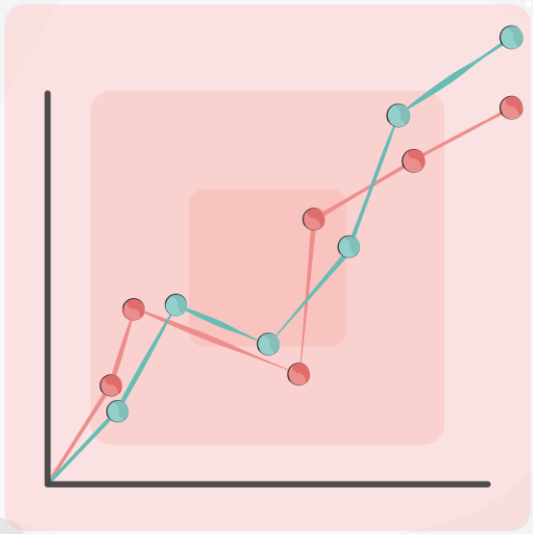
No one-size-fits-all pipeline

- Track data lineage and transformations
- Enforce schema validation and access control
- Log data movement and changes
- Governance is embedded, not optional



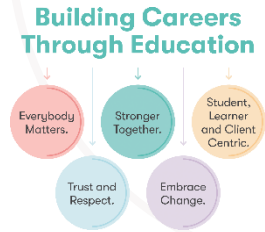
Hybrid, Modular Design for Scalable Data Governance

Practical lab



Exercise part 1

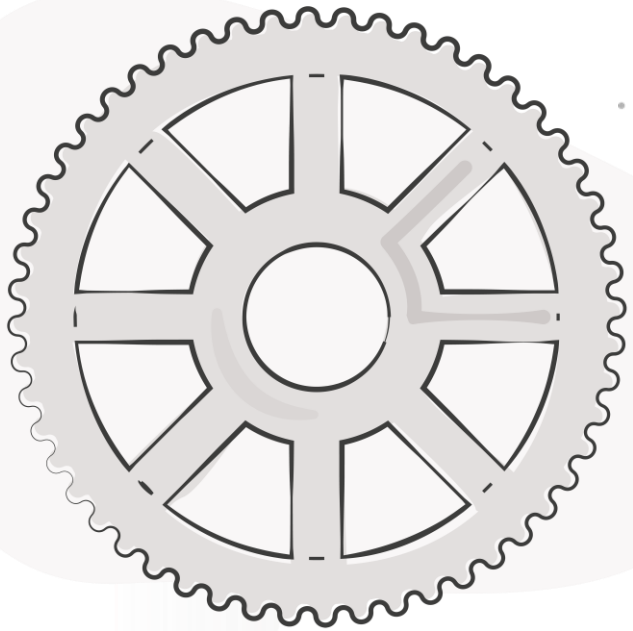
Initial Design Based on UK Data



1. **Design Database Schema:** Create a schema to receive user data from an existing software application.
2. **Implement Data Pipeline:** Develop a pipeline to clean, validate, and load data into the new schema.
3. **Use Test Data**: Utilise provided test data for 10 UK users and their login timestamps.
4. **Instructor Support:** Your instructor will act as the stakeholder SME for any data-related questions.
5. **Future Phases:** Prepare to integrate data from additional countries with varying formats.
6. **Documentation:** Establish comprehensive documentation for the schema and pipeline for future implementation by other teams.

Files Provided in the Hub:

- UK User Data.csv: Contains 10 sample records.
- UK-User-LoginTS.csv: Contains login timestamps for January 2025.



Practical challenge

BPP

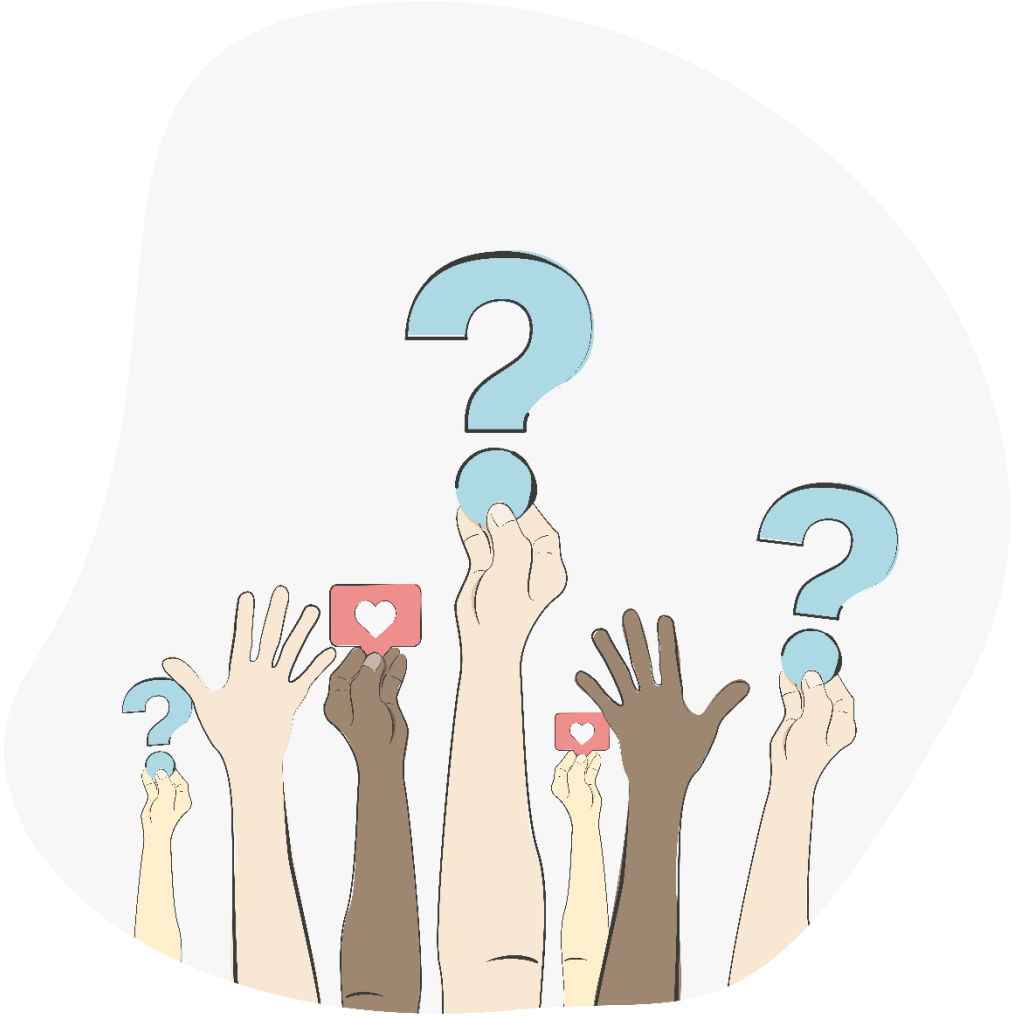
Key Learning Summary

Here are some key points to takeaway from this session:

- **Traditional ETL is limited** by latency, rigidity, and poor support for real-time, unstructured, or large-scale data needs.
- **Modern integration paradigms** include data federation, virtualisation, blending, and ELT—each suited to different pipeline goals and data types.
- **Streaming data integration** enables real-time insights using tools like Kafka and CDC, supporting event-driven architectures.
- **Architectural models** such as point-to-point, hub-and-spoke, and distributed integration impact scalability, fault tolerance, and governance.
- **Tooling ecosystems** (e.g., Apache NiFi, Talend, Kafka, dbt, Airflow) support ingestion, transformation, orchestration, and monitoring across the pipeline lifecycle.
- **Data quality and governance** are essential for reliable pipelines, using mechanisms like schema validation, lineage tracking, and automated quality checks.
- **Hybrid integration strategies** are often necessary—combining multiple techniques to balance speed, flexibility, and control.

BPP

Any questions or feedback?



Building Careers Through Education

Everybody Matters.
Stronger Together.
Student, Learner and Client Centric.
Trust and Respect.
Embrace Change.

BPP

Thank you