Data Lake

- Ein Data Lake ist ein zentrales Repository, das strukturierte, semi-strukturierte und unstrukturierte Rohdaten in großem Umfang speichert
- Eignet sich besonders für Big-Data-Analysen und Machine Learning, erfordert aber zusätzliche Verarbeitung zur Datennutzung

Data Warehouse

- Ein Data Warehouse ist ein zentrales, strukturiertes System zur langfristigen Speicherung und Analyse großer Datenmengen aus verschiedenen Quellen
- Optimiert für komplexe Abfragen und Business-Intelligence-Anwendungen

Cloud Data Warehouse

- Ein Cloud Data Warehouse ist ein Data Warehouse, das in der Cloud gehostet wird und dadurch flexibel skalierbar und wartungsarm ist. Beispiele sind Snowflake, Google BigQuery oder Amazon Redshift

Data Lakehouse

- Datenarchitektur, die die Flexibilität und Skalierbarkeit eines Data Lakes mit den strukturierten Datenmanagement- und Analysefunktionen eines Data Warehouses kombiniert
- Ermöglicht sowohl explorative Datenanalyse als auch strukturierte BI-Abfragen auf einer gemeinsamen Plattform
- ACID Transaktionen:
  - Atomicity: Transaktion wird entweder vollständig ausgeführt oder gar nicht – es gibt keine halbfertigen Zustände
  - Consistency: Transaktion überführt die Datenbank von einem konsistenten Zustand in einen anderen, wobei alle definierten Regeln (z.B. Integritätsbedingungen) eingehalten werden
  - Isolation: Gleichzeitige Transaktionen beeinflussen sich nicht gegenseitig – jede Transaktion läuft so, als wäre sie allein im System
  - Durability: Sobald eine Transaktion abgeschlossen ist, bleiben ihre Änderungen dauerhaft gespeichert, selbst bei Systemausfällen

Data Mart

- Spezialisierte Teilmenge eines Data Warehouses, die auf die Anforderungen eines bestimmten Fachbereichs zugeschnitten ist. Es verbessert die Performance und Übersichtlichkeit für gezielte Analysen

Data Mesh

- Dezentraler Ansatz zur Datenarchitektur, bei dem einzelne Teams für ihre eigenen Datenprodukte verantwortlich sind ("Data as a Product")
- Fördert Skalierbarkeit und Eigenverantwortung durch domänenorientierte Datenverwaltung

Datenprodukt

- Wiederverwendbares, klar definiertes Datenartefakt (Datensatz), das für einen konkreten Anwendungsfall erstellt wird und von anderen leicht konsumiert werden kann

ETL

- Prozess, bei dem Daten aus verschiedenen Quellen extrahiert, in ein geeignetes Format transformiert und schließlich in ein Zielsystem wie ein Data Warehouse geladen werden
- Dient dazu, Rohdaten für Analysen nutzbar zu machen, indem sie bereinigt, vereinheitlicht und angereichert werden
- Moderne ETL-Prozesse können auch in Echtzeit (Streaming-ETL) oder als ELT-Variante ablaufen, bei der die Transformation nach dem Laden erfolgt

ETL Pipeline

- Extraktion → ADF (Extrahieren von Daten aus verschiedenen Quellen)
- Transform → Databricks und Spark (Verarbeiten, bereinigen und anreichern der Daten in skalierbaren Umgebungen)
- Load → Snowflake, BigQuery (Laden der transformierten Daten in ein Cloud DWH)
- Orchestrierung → Apache Airflow (Steuerung und Automatisierung des Ablaufs der gesamten ETL-Prozesse)
- Montoring → Grafana, Azure Monitor (Überwachung von Performance, Ausfällen und Fehlern der Pipeline)
- Data Quality & Testing → Great Expectations (Validieren der Datenqualität durch automatisierte Tests und Regeln)

Pipeline Montoring

- Überwachung von Datenpipelines, Systemen und Prozessen, um sicherzustellen, dass Daten zuverlässig, korrekt und zielgerecht verarbeitet werden
- Fehlererkennung:
  - Erkennen von fehlgeschlagenen ETL-Jobs
  - Identifikation von Datenanomalien
  - Monitoring von Datenlatenzen
- Performance Überwachung:
  - Laufzeit von Pipelines
  - Ressourcenverbrauch
  - Datenvolumen und Verarbeitungsraten
- Sicherstellen der Datenqualität:
  - Validierung von Daten gegen Regeln
  - Schema Überwachung
- Transparenz und Reporting:
  - Dashboarding von Metriken (Grafana, Power BI)
  - Alerting bei Schwellenwertüberschreitung
- Beispiele:
  - Apache Airflow Monitoring: Überwacht den Status von DAGs, ob Tasks fehlschlagen oder hängenbleiben

  - Snowflake oder BigQuery Monitoring: Überwachung von Query-Ausführungszeiten und -Kosten
  - Streaming Monitoring: Sicherstellen, dass Messages rechtzeitig und vollständig verarbeitet werden

Datenmanagement

- Übergeordnete Prozess der Erfassung, Speicherung, Organisation, Pflege und Nutzung von Daten in einem Unternehmen
- Umfasst verschiedene Disziplinen wie Data Governance, Datenintegration, Datenqualität, Metadatenmanagement und Archivierung, um den wertschöpfenden Einsatz von Daten sicherzustellen
- Datenintegrität: Zusammenführen und Vereinheitlichung von Daten aus verschiedenen Quellen

Data Governance

- Bezeichnet den Rahmen aus Richtlinien, Prozessen und Verantwortlichkeiten, der sicherstellt, dass Daten im Unternehmen korrekt, sicher, einheitlich und regelkonform verwaltet werden
- Umfasst Themen wie Datenqualität, Datenschutz, Zugriffsrechte, Compliance und Datenverantwortung, um Vertrauen und Kontrolle über Daten zu gewährleisten

Datenqualität

- Vollständigkeit → Sind alle erforderlichen Datenfelder vorhanden und ausgefüllt?
- Korrektheit → Entsprechen die Daten den realen, erwarteten Werten (z.B. stimmen Postleitzahlen mit Städten überein)?
- Konsistenz → Sind die Daten in verschiedenen Systemen oder Tabellen widerspruchsfrei (z.B. gleicher Kundennamen in allen Datensätzen)?
- Aktualität → Sind die Daten aktuell bzw. zeitgerecht verarbeitet (z.B. keine veralteten Transaktionen im Reporting)?
- Eindeutigkeit → Gibt es doppelte Datensätze, wo es keine geben sollte (z.B. doppelte Kunden-IDs)?
- Validität → Entsprechen die Daten den erwarteten Formaten oder Regeln (z.B. E-Mail-Adressen im gültigen Format, Zahlen in numerischen Feldern)?

CI/CD

- CI: Automatisches Testen und Bauen bei jeder Codeänderung
- CDelivery: Automatisches Ausliefern in eine Staging-Umgebung
- CDeployment: Automatisches Ausliefern bis in die Produktion (ohne manuelle Eingriffe)

Data Analytics Konzepte

- EDA
- Deskriptive, diagnostische, prädiktive und präskriptive Analysen
- KPI-Tracking: Überwachung von Leistungskennzahlen

Dokumentation

- Datenquellen:
  - Herkunft (Systeme, APIs, Dateien)
  - Zugriffsmethoden und -rechte
  - Aktualisierungsfrequenz
- ETL-/ELT-Prozesse:
  - Datenflüsse und Pipeline-Übersichten
  - Verwendete Transformationslogik (SQL, dbt, Spark etc.)
  - Abhängigkeiten zwischen Schritten
- Datenmodelle und Schemata:
  - Tabellen, Views, Spaltenbeschreibungen
  - Beziehungen zwischen Entitäten (z.B. ER-Diagramme)
  - Versionierung von Modellen
- Datenqualitätsregeln:
  - Validierungschecks
  - Testdefinitionen (z.B. dbt tests, Great Expectations)
- Zugriffs- und Sicherheitsrichtlinien:
  - Rollen und Berechtigungen
  - Datenklassifizierung (z.B. PII, öffentlich, vertraulich)
- Monitoring & Alerting:
  - Überwachte Metriken
  - Fehlerbehandlung und Wiederanläufe
  - SLAs und SLOs
- Code- und Tool-Dokumentation:
  - Repos, Technologien, Abhängigkeiten
  - Setup-Anleitungen und Deployment-Prozesse