Curriculum für

Certified Professional for Software Architecture (CPSA)®
Advanced Level

Modul
DATA

Entwerfen, Erstellen und Warten datenzentrierter Softwarearchitekturen

2024.1-RC3-DE-20240705

Inhaltsverzeichnis

Die Abkürzung "e. V." ist Teil des offiziellen Namens des iSAQB und steht für "eingetragener Verein", der seinen Status als juristische Person nach deutschem Recht beschreibt. Der Einfachheit halber wird iSAQB e. V. im Folgenden ohne die Verwendung dieser Abkürzung als iSAQB bezeichnet.

Verzeichnis der Lernziele

- LZ 6-2 - Query Programmiermodelle
- LZ 6-3 - Query Verarbeitung & Optimierung
- LZ 7-1 - Begriffsbestimmung (Daten-)transformation
- LZ 7-2 - Anwendungsfälle
- LZ 7-3 - Typische Transformationen
- LZ 7-4 - Staging Area
- LZ 7-5 - Robuste Transformationen
- LZ 7-6 - Qualitätsstufen
- LZ 7-7 - Batch Verarbeitung
- LZ 7-8 - Stream Verarbeitung
- LZ 8-1 - Anwendungsfälle
- LZ 8-2 - Repräsentation von Massendaten
- LZ 8-3 - Modularisierung
- LZ 8-4 - Data Analytics und Business Intelligence
- LZ 8-5 - Machine Learning
- LZ 8-6 - Reverse ETL
- LZ 9-1 - Begriffsbestimmung Data Pipelines
- LZ 9-2 - Anwendungsgebiete von Data Pipelines
- LZ 9-3 - Arten von Data Pipelines
- LZ 9-4 - Qualitätskriterien für Data Pipelines
- LZ 9-5 - Building Blocks von Data Pipelines
- LZ 9-6 - Technologien und Plattformen für Data Pipelines
- LZ 9-7 - Betrieb von Data Pipelines
- LZ 10-1 - Nachteile zentraler Datenarchitekturen
- LZ 10-2 - Begriffsbestimmung Data Mesh
- LZ 10-3 - Domain Ownership
- LZ 10-4 - Data as a Product
- LZ 10-5 - Self-serve Data Platform
- LZ 10-6 - Federated Computational Governance
- LZ 10-7 - Top Down vs. Bottom up Realisierung
- LZ 11-1 - Übergreifende Themen zum Datenlebenszyklus
- LZ 11-2 - Data Governance
- LZ 11-3 - Data Stewardship und Ownership
- LZ 11-4 - Datenzugriff und -berechtigungen
- LZ 11-5 - Datenschutz, Compliance, Datensicherheit

- LZ 11-6 - Qualität von Daten
- LZ 11-7 - Data Contracts
- LZ 11-8 - Metadaten
- LZ 11-9 - Betriebsaspekte

Einführung: Allgemeines zum iSAQB Advanced Level

Was vermittelt ein Advanced Level Modul?

Das Modul kann unabhängig von einer CPSA-F-Zertifizierung besucht werden.

- Der iSAQB Advanced Level bietet eine modulare Ausbildung in drei Kompetenzbereichen mit flexibel gestaltbaren Ausbildungswegen. Er berücksichtigt individuelle Neigungen und Schwerpunkte.
- Die Zertifizierung erfolgt als Hausarbeit. Die Bewertung und mündliche Prüfung wird durch vom iSAQB benannte Expert:innen vorgenommen.

Was können Absolventen des Advanced Level (CPSA-A)?

CPSA-A-Absolventen können:

- eigenständig und methodisch fundiert mittlere bis große IT-Systeme entwerfen
- in IT-Systemen mittlerer bis hoher Kritikalität technische und inhaltliche Verantwortung übernehmen
- Maßnahmen zur Erreichung von Qualitätsanforderungen konzeptionieren, entwerfen und dokumentieren sowie Entwicklungsteams bei der Umsetzung dieser Maßnahmen begleiten
- architekturrelevante Kommunikation in mittleren bis großen Entwicklungsteams steuern und durchführen

Voraussetzungen zur CPSA-A-Zertifizierung

- erfolgreiche Ausbildung und Zertifizierung zum Certified Professional for Software Architecture, Foundation Level® (CPSA-F)
- mindestens drei Jahre Vollzeit-Berufserfahrung in der IT-Branche; dabei Mitarbeit an Entwurf und Entwicklung von mindestens zwei unterschiedlichen IT-Systemen
    - Ausnahmen sind auf Antrag zulässig (etwa: Mitarbeit in Open-Source-Projekten)
- Aus- und Weiterbildung im Rahmen von iSAQB-Advanced-Level-Schulungen im Umfang von mindestens 70 Credit Points aus mindestens drei unterschiedlichen Kompetenzbereichen
- erfolgreiche Bearbeitung der CPSA-A-Zertifizierungsprüfung

Grundlegendes

Was vermittelt das Modul „DATA“?

Daten werden in modernen Anwendungssystemen zunehmend zu “First Class Citizens”. Gründe dafür sind etwa der deutlich gestiegene und weiter steigende Einsatz mobiler Endgeräte, die massenweise Integration von Geräten über IoT, die universelle Verfügbarkeit von Anwendungen in der Cloud, der zunehmende Einsatz KI-basierter Lösungen und natürlich die Verfügbarkeit günstiger digitaler Speichertechnologie. Dies zeigt sich auch in den Trends “Big Data” oder “Data Driven”. Entsprechend sind in den letzten Jahren die Anforderungen an Software- und Systemarchitekten gestiegen, sich mit geeigneten Architekturen zur Verarbeitung großer Datenmengen zu befassen. Der hier vorliegende iSAQB Lehrplan soll einen möglichst umfassenden Überblick über alle Aspekte bieten, die im Zusammenhang mit der Verarbeitung großer Datenmengen aus der Sicht von Software- und Systemarchitekten beachtet werden müssen.

Was vermittelt das Modul „DATA“ nicht?

Insbesondere die folgenden Themengebiete sind im Lehrplan zwar enthalten, werden aber nicht vertieft behandelt:

- Data Science (Statistik, Machine Learning)
- Business Intelligence (OLAP, Multidimensional Modeling)
- Datenschutz

In vielen Spezialgebieten spielt die Verarbeitung großer Datenmengen ebenfalls eine maßgebliche Rolle. Sie werden in diesem Lehrplan nicht explizit behandelt. Beispiele dafür sind IoT, Suchmaschinen oder Wissenschaftliche Anwendungen (Klimaforschung, Kernphysik, Proteinfaltung, etc).

Struktur des Lehrplans und empfohlene zeitliche Aufteilung

Inhalt	Empfohlene Mindestdauer (min)
1. Motivation und Übersicht	90
2. Referenzarchitekturen für analytische Anwendungssysteme	210
3. Datenquellen	60
4. Ingestion	90
5. Datenspeicher	90
6. Queries	60
7. Transformation	90
8. Nutzung und Analyse	60
9. Data Pipelines	90
10. Data Mesh	120
11. Data Governance	120
Summe	1080 (18h)

Dauer, Didaktik und weitere Details

Die unten genannten Zeiten sind Empfehlungen. Die Dauer einer Schulung zum Modul DATA sollte mindestens 3 Tage betragen, kann aber länger sein. Anbieter können sich durch Dauer, Didaktik, Art und Aufbau der Übungen sowie der detaillierten Kursgliederung voneinander unterscheiden. Insbesondere die Art der Beispiele und Übungen lässt der Lehrplan komplett offen.

Lizenzierte Schulungen zu DATA tragen zur Zulassung zur abschließenden Advanced-Level-Zertifizierungsprüfung folgende Credit Points) bei:

Methodische Kompetenz:	20 Punkte
Technische Kompetenz:	10 Punkte
Kommunikative Kompetenz:	0 Punkte

Voraussetzungen

Teilnehmerinnen und Teilnehmer **sollten** folgende Kenntnisse und/oder Erfahrung mitbringen:

- Grundlagen der Beschreibung von Architekturen mithilfe verschiedener Sichten, übergreifender Konzepte, Entwurfsentscheidungen, Randbedingungen etc., wie es im CPSA-F (Foundation Level) vermittelt wird.
- Erfahrung mit der Implementierung und Architektur in agilen Projekten.
- Erfahrungen aus der Entwicklung und Architektur datenzentrierter Anwendungen mit den typischen Herausforderungen.

Hilfreich für das Verständnis einiger Konzepte sind darüber hinaus:

- Kenntnisse über SQL und über Probleme und Herausforderungen bei der Nutzung großer Datenbanken
- Kentnisse über Probleme und Herausforderungen bei der Implementierung verteilter Systeme
- Kenntnisse über fachliche Modularisierung

Gliederung des Lehrplans

Die einzelnen Abschnitte des Lehrplans sind gemäß folgender Gliederung beschrieben:

- **Begriffe/Konzepte**: Wesentliche Kernbegriffe dieses Themas.
- **Unterrichts-/Übungszeit**: Legt die Unterrichts- und Übungszeit fest, die für dieses Thema bzw. dessen Übung in einer akkreditierten Schulung mindestens aufgewendet werden muss.
- **Lernziele**: Beschreibt die zu vermittelnden Inhalte inklusive ihrer Kernbegriffe und -konzepte.

Dieser Abschnitt skizziert damit auch die zu erwerbenden Kenntnisse in entsprechenden Schulungen.

Ergänzende Informationen, Begriffe, Übersetzungen

Soweit für das Verständnis des Lehrplans erforderlich, haben wir Fachbegriffe ins iSAQB-Glossar aufgenommen, definiert und bei Bedarf durch die Übersetzungen der Originalliteratur ergänzt.

1. Motivation und Übersicht

Dauer: 90 Min.	Übungszeit: 15 Min.

1.1. Begriffe und Konzepte

operative Daten, analytische Daten, OLTP, OLAP, Data Engineer, Data Architect, Data Scientist, Lebenszyklus, Datenanalyse, Datenarchitekturen

1.2. Lernziele

LZ 1-1 - Data Engineering

Die Teilnehmer:innen können erläutern, was Data Engineering ist und kennen übliche Definitionen. Die Teilnehmer:innen können erklären, wie sich die Data Engineering entwickelt hat, um den Anforderungen von Data Science und Data Analytics hinsichtlich Datensammlung, -speicherung, -verarbeitung und -analyse zu entsprechen.

LZ 1-2 - Unterscheidung operativer und analytischer Daten

Die Teilnehmer:innen können die Unterschiede hinsichtlich Verwendung, Struktur und Kontext zwischen operativen und analytischen Daten erläutern. Sie können die Begriffe OLTP und OLAP unterscheiden. Ihnen ist klar, warum operative und analytische Daten üblicherweise separat verarbeitet werden und sie kennen typische Beispiele für operative und analytische Anwendungen. Sie können die folgenden Eigenschaften analytischer Daten erläutern:

- subjektorientiert
- beständig
- integrierbar
- historisiert

Die Teilnehmer:innen kennen Beispiele, wie analytische Daten aus operativen Anwendungen entstehen. Sie kennen ebenfalls Beispiele, wie Ergebnisse von Analysen analytischer Daten in operativen Anwendungen genutzt werden können.

LZ 1-3 - Kategorien der Datenanalyse

Die Teilnehmer:innen kennen die typischen Kategorien der Datenanalyse, können diese unterscheiden und jeweils Beispiele dazu nennen:

- Deskriptive Analyse z.B. Berichtswesen
- Explorative Analyse z.B. Ad-Hoc Reporting
- Inferentielle Analyse: Ableitung von Aussagen aus Stichproben, z.B. Wirksamkeitsanalyse von Medikamenten
- Prädiktive Analyse: zukünftige Vorhersagen basierend auf historische Daten, z.B. Kreditrisikobewertung
- Kausale Analyse: Erkennung von Ursache-Wirkung-Beziehungen
- Mechanistische Analyse: Integration spezifischen Fachwissens, z.B. physikalischer, biologischer oder chemischer Modelle

LZ 1-4 - Herausforderungen analytischer Anwendungen

Die Teilnehmer:innen können übliche Herausforderungen bei Konzeption, Umsetzung und Betrieb analytischer Anwendungssysteme speziell im Hinblick auf die folgenden Voraussetzungen bennenen:

- Umfang und Komplexität der zu verarbeitenen Daten
- Komplexität und Anzahl der Analysen, Anfragen / Queries
- Verfügbarkeit
- Datensicherheit

LZ 1-5 - Rollen im Data Engineering

Die Teilnehmer:innen können Rollen wie Data Architect, Data Analyst, Machine Learning Engineer und Data Scientist beschreiben und voneinander abgrenzen.

Den Teilnehmer:innen ist bewusst, dass die Ausprägung der jeweiligen Rollen von der Effizienz und Professionalität der jeweiligen Organisationen im Hinblick auf die Arbeit mit Daten abhängen.

LZ 1-6 - Monolithische und verteilte Datenarchitekturen

Die Teilnehmer:innen verstehen die Vor- und Nachteile monolithischer und verteilter Softwarearchitekturen für analytische Anwendungen.

Die Teilnehmer:innen wissen, dass analytische Anwendungen bisher häufig eine monolithische Architektur haben und dass mit der Diskussion um Data Mesh (siehe LZ 10-1 - Nachteile zentraler Datenarchitekturen) aktuell ein Paradigmenwandel stattfindet.

LZ 1-7 - Lebenszyklus analytischer Daten

Die Teilnehmer:innen können die folgenden Phasen der Datenverarbeitung zur Vereinheitlichung analytischer Daten unterscheiden:

- Sources - Das Erzeugen der Daten (siehe LZ 3-1 - Arten von Datenquellen und Quellsystemen)
- Ingestion und Transport - Das Extrahieren der Daten und deren Transport zu den Speichersystemen (siehe LZ 4-1 - Begriffsbestimmung Data Ingestion)
- Storage - Das Abspeichern der Daten (siehe LZ 5-1 - Speichersysteme)
- Query und Processing - Code zur Verarbeitung und Abfragen gegen die gespeicherten Daten (siehe LZ 6-1 - Analytische Queries)
- Transformation - Transformation der Daten in eine für die Analyse geeignete Form (siehe LZ 7-1 - Begriffsbestimmung (Daten-)transformation)
- Analysis und Output - Präsentation der Daten für die Analyse und deren Ergebnisse sowie Integration der Analysemöglichkeiten in Anwendungen (siehe LZ 8-1 - Anwendungsfälle)

Den Teilnehmer:innen ist bewußt, dass begleitend zu diesen Phasen auch die folgenden Querschnittsthemen zu beachten sind:

- Data Discovery - Information bereitstellen, welche Daten es gibt, wie sie aufbereitet und wo sie zu finden sind
- Data Governance - Erarbeiten und Überwachen allgemein gültiger Regeln für die Verarbeitung der Daten (siehe LZ 11-1 - Übergreifende Themen zum Datenlebenszyklus)

- Data Security - Gewährleisten eines angemessenen Grads an Datenschutz (siehe LZ 11-2 - Data Governance)
- Data Quality - Gewährleisten eines angemessenen Grads an Datenqualität (siehe LZ 11-3 - Data Stewardship und Ownership)

1.3. Referenzen

[E. F. Codd 1990], [W. H. Inmon 2005], [R. Kimball 2011]

2. Referenzarchitekturen für analytische Anwendungssysteme

Dauer: 210 Min.	Übungszeit: 30 Min.

2.1. Begriffe und Konzepte

Referenzarchitekturen, Data Warehouse, Data Lake, Lambda Architecture, Cloud, on-Premise, monolithisches System, förderiertes System, Stream & Batch Processing, Skalierbarkeit

2.2. Lernziele

LZ 2-1 - Überblick zu Referenzarchitekturen

Den Teilnehmer:innen ist der generelle Nutzen von Referenzarchitekturen bewusst:

- eine Vorlage für eine funktionale Zerlegung in wesentliche Bestandteile zur Verfügung zu stellen
- ein einheitliches Vorgehen für die Aufnahme, Verarbeitung und Verwendung von Daten vorzugeben
- dadurch eine Vorlage für die Ausprägung konkreter Software-Architekturen zu bieten

Die Teilnehmer:innen verstehen, dass Referenzarchitekturen für analytische Anwendungssysteme generell unterschieden werden, je nachdem, ob eine Basis für

- die Vereinheitlichung und Integration analytischer Daten
- die Anwendung von Verfahren der KI und des ML auf analytischen Daten (nicht Bestandteil dieses Lehrplans)
- beides zugleich

geschaffen werden soll.

Die Teilnehmer:innen kennen Ansätze, ein einheitliches Framework für Referenzarchitekturen zu analytischen Anwendungssystemen zu beschreiben.

Die Teilnehmer:innen verstehen, dass Referenzarchitekturen sowohl in monolithischen Ansätzen für die Architektur des Gesamtsystem herangezogen werden können, als auch in verteilten Ansätzen wie Data Mesh (siehe LZ 10-1 - Nachteile zentraler Datenarchitekturen) für die Architektur jedes Teilsystems eingesetzt werden können.

Die Teilnehmer:innen kennen Beispiele von Referenzarchitekturen für analytische Anwendungssysteme wie etwa

- Data Warehouse
- Data Lake
- Lambda Architecture

LZ 2-2 - Referenzarchitekturen zur Vereinheitlichung analytischer Daten

Die Teilnehmer:innen kennen die Phasen der Datenverarbeitung und wissen, welche Toolkategorien in den jeweiligen Phasen typischerweise zum Einsatz kommen (siehe [M. Bornstein 2020]).

Die Teilnehmer:innen kennen Komponenten die zur Abdeckung querschnittlicher Themen geeignet sind.

LZ 2-3 - Architekturentscheidungen anhand von Referenzarchitekturen

Die Teilnehmer:innen können Architekturentscheidungen etwa zu den folgenden Fragestellungen anhand von Referenzarchitekturen diskutieren:

- ob ein zentrales/monolithisches oder dezentrales/förderiertes System zu wählen ist
- ob ein Deployment in der Cloud (ggfs. Cloud-Ready), hybrid oder on-Premise vorzuziehen ist
- ob eine eigenständige Entwicklung von Komponenten erforderlich ist oder das Kaufen und Konfigurieren von Standardkomponenten ausreicht
- ob die Integration von Komponenten selbst vorgenommen oder auf die Verwendung bereits integrierter Komponenten vertraut wird
- ob Stream-, Batch-Processing (ggfs Micro-Batch) oder beides unterstützt werden muss
- welche Form der Skalierbarkeit konkret benötigt wird, z.B. Datenvolumen, Anzahl der Nutzer
- welche Rollen und Verantwortlichkeiten vorgesehen werden müssen

2.3. Referenzen

[J. Reis 2022], [M. Bornstein 2020], [W. H. Inmon 2005], [R. Kimball 2011], [D. Linstedt 2015], [C. Giebler et al. 2021], [P. Pääkkönen 2015]

3. Data Sources

Dauer: 60 Min.	Übungszeit: 15 Min.

3.1. Begriffe und Konzepte

Datenquellen, Strukturiertheit, Formate, Performance, Datenbanksysteme, API, Dateisysteme, Object Stores, Message Queue, Event Streaming

3.2. Lernziele

LZ 3-1 - Arten von Datenquellen und Quellsystemen

Die Teilnehmer:innen kennen Beispiele für inhaltliche und technische Arten von Datenquellen und Quellsystemen, von denen analytische Daten typischerweise übernommen werden, wie etwa:

inhaltliche Quellen

- Sensordaten
- Standard Anwendungen
- Logging Systeme
- (Web-) Analytics
- Metrics Systeme

technische Quellen

- APIs
- Datenbanken
- Event-Feed
- Messaging Systeme
- (Cloud) Storage Systeme

LZ 3-2 - Eigenschaften von Datenquellen und Quellsystemen

Die Teilnehmer:innen können unterscheiden, ob Datenquellen strukturierte, semi-strukturierte oder unstukturierte Daten liefern. Ihnen sind übliche Formate vertraut, in denen Datenquellen ihre Daten zur Verfügung stellen, wie etwa

- Dokumente
- Liste von Datensätzen
- Key-value Paare

Den Teilnehmer:innen ist bewusst, dass Quellsysteme z.T. sehr große Datenmengen bereitstellen, und dass geeignete Maßnahmen getroffen werden müssen, um diese Datenmengen transportieren zu können.

Die Teilnehmer:innen wissen, dass das zur Verfügung stellen analytischer Daten zusätzliche Last für das jeweilige Quellsystem erzeugt werden kann und dass dies i.d.R. den eigentlichen Zweck eines Quellsystems nicht beeinträchtigen darf. Den Teilnehmer:innen ist bewusst, dass es durch spezifische

technologische oder organisatorische Rahmenbedingungen schwierig sein kann auf die Daten zuzugreifen.

Die Teilnehmer:innen wissen, dass Quellsysteme Daten unter Umständen redundant bereitstellen und können entsprechend mit ihnen umgehen.

Den Teilnehmer:innen ist bewusst, welche Garantien, wie Konsistenz, Integrität, Aktualität, Korrektheit eine Datenquelle zur Verfügung stellen kann.

Die Teilnehmer:innen wissen, dass die Datenhoheit bei den Quellsystemen liegt und Datenmodelle durch diese geändert werden können.

Den Teilnehmer:innen ist bewusst, dass Quellsysteme Daten in einem internen Format verwalten, deren Bereitstellung nach aussen problematisch sein kann, weil es dabei zu einer engen Kopplung der Datenmodelle kommen kann.

LZ 3-3 - Bereitstellung von Daten durch anwendungspezifische APIs

Den Teilnehmer:innen ist bewusst, das APIs dazu dienen das interne Datenmodell zu abstrahieren und zu definieren was extern bereitgestellt werden soll. Sie kennen zentrale Aspekte beim Entwurf von APIs wie Versionierung und Schemata.

Den Teilnehmer:innen ist bewusst, das APIs üblicherweise vom Quellsystem bereitgestellt werden, aber auch vom Zielsystem angeboten werden können (reverse APIs). Sie kennen Arten von APIs die zur Bereitstellung von Daten dienen wie beispielsweise REST und RPC, Ansätze die Daten direkt transformieren wie GraphQL, sowie Ansätze für reverse APIs wie beispielsweise Webhooks.

LZ 3-4 - Bereitstellung von Daten durch Datenbanksysteme

Die Teilnehmer:innen wissen, dass Datenbanken durch ein Datenbankmanagementsystem verwaltet werden, welches beispielsweise Schemaverwaltung, Datenzugriff, Query-Optimizer, Datenverteilung bereitstellt. Ihnen ist bewusst, das Datenbanken vorrangig für die Verwaltung strukturierter Daten verwendet werden.

Die Teilnehmer:innen kennen Open-Table-Formate, welche einen datenbankähnlichen Zugriff auf Dateisysteme und Object Stores erlauben, als Alternative zu traditionellen, monolithischen Datenbankmanagementsystemen.

Die Teilnehmer:innen kennen den Unterschied zwischen relationalen und nicht-relationalen Datenbanken und kennen Beispiele für nicht-reationale Datenbanken wie Zeitreihen-, Graph- oder Dokumentendatenbanken.

LZ 3-5 - Bereitstellung von Daten durch Dateisysteme

Die Teilnehmer:innen wissen, dass Dateisysteme hierarchisch oder als Blockspeicher organisiert sein können. Sie wissen, dass der Speicherplatz bei hierarchischen Dateisystemen durch Verzeichnisse strukturiert wird, während er bei Blockspeichern in Blöcke fester Größe eingeteilt ist und sind sich der Konsequenzen für die Navigation durch das jeweilige Dateisystem bewusst.

Den Teilnehmer:innen ist bewusst, dass die Datenstruktur durch das Dateiformat festgelegt ist. Sie kennen Dateiformate die für den Einsatz mit großen Datenmengen optimiert sind.

Die Teilnehmer:innen ist bewusst, dass der Datenzugriff über Systemaufrufe auf Betriebssystemebene erfolgt und die Zugriffsberechtigungen ebenfalls vom Betriebssystem verwaltet werden.

Die Teilnehmer:innen kennen Beispiele für Disk-basierte und verteilte Dateisysteme, sowie für Blockspeicher. Sie kennen die Möglichkeiten moderner Dateisysteme hinsichtlich Kompression, Snapshots, Verschlüsselung und Datenintegrität

LZ 3-6 - Bereitstellung von Daten durch Object Stores

Die Teilnehmer:innen wissen, das der Zugriff auf die Daten typischerweise durch RESTful APIs erfolgt. Sie wissen, dass Object Stores für die Speicherung großer Mengen unstrukturierter Daten konzipiert sind und sich üblicherweise horizontal sehr gut skalieren lassen.

Den Teilnehmer:innen sind die Möglichkeiten hinsichtlich Metadaten für Kategorisierung und Verwaltung, sowie einer flexiblen Verwaltung von Zugriffsberechtigungen bewusst. Sie kennen Möglichkeiten um einen Object Store als Dateisystem zu verwenden.

Die Teilnehmer:innen kennen Beispiele für Object Stores für den Cloud und on-Premise Betrieb.

LZ 3-7 - Bereitstellung von Daten durch Message Queues und Event-Streaming

Die Teilnehmer:innen wissen, dass beide Ansätze auf einer asynchronen Bereitstellung von Daten durch das Quellsystem basieren und ein synchroner Zugriff auf das Quellsystem üblicherweise nicht möglich ist.

Die Teilnehmer:innen kennen den Unterschied zwischen Messaging Systemen und Streaming Ansätzen. Die Teilnehmer:innen wissen, dass Event Streams ein geordnetes Log der bereitgestellten Daten darstellen, welches von verschiedenen Zielsystemen abonniert werden kann aber nicht muss (Publish Subscribe), wohingegen Messaging Systeme die Daten solange vorhalten, bis sie von einem Zielsystem explizit abgeholt wurden.

Die Teilnehmer:innen sind sich der Problematik der Übertragung des initialen Zustandes bewusst und kennen Ansätze wie z.B. Snapshots um das Thema zu adressieren.

Den Teilnehmer:innen ist bewusst, dass Daten mehrfach (at least once) gesendet werden können, da es aufwendig ist sicherzustellen, dass Daten genau einmal bereitgestellt wurden (exactly once). Sie kennen die Vorteile einer idempotenten Verarbeitung.

Den Teilnehmer:innen ist bewusst, dass die Verwendung von Streams zu temporären Inkonsistenzen im Datenbestand führen kann (eventual consistency).

3.3. Referenzen

[R. Castagna 2022]

4. Ingestion und Transport

Dauer: 90 Min.	Übungszeit: 15 Min.

4.1. Begriffe und Konzepte

Data Ingestion, Identifikation von Entitäten, Änderungserkennung, Change Data Capture (CDC), Outbox Pattern, Konnektoren, Skalierbarkeit, Datenreplkation, Batch vs. Stream Ingestion, Metadatenmanagemen

4.2. Lernziele

LZ 4-1 - Begriffsbestimmung Data Ingestion

Die Teilnehmer:innen wissen, dass die Data Ingestion der Übernahme der Daten aus den Source Systemen in die Speichersysteme der analytischen Systeme dient. Sie wissen auch, dass damit auch eine Verantwortungsübernahme verbunden ist. Sie verstehen, dass mittels Data Ingestion eine Entkoppelung zwischen Source Systemen und analytischer Verarbeitung ermöglicht wird.

LZ 4-2 - Identifizieren von Entitäten

Den Teilnehmer:innen sind Verfahren für das Identifizieren von Entitäten in Quellsystemen bekannt, wie etwa Primary Keys (technische vs. fachliche Schlüssel)

Das Erkennen von Duplikaten ist Bestandteil von Modul 7 (siehe LZ 7-2 - Anwendungsfälle).

LZ 4-3 - Erkennen von Änderungen

Den Teilnehmer:innen sind Verfahren für das Erkennen von Änderungen dieser Entitäten in Quellsystemen bekannt, wie etwa

- Change Data Capture (CDC)
- DB-Trigger
- Create und Update Timestamps

Die Teilnehmer:innen wissen, für welche Art von Quellsystemen welche dieser Verfahren jeweils geeignet sind.

Den Teilnehmer:innen ist bewusst, dass das Löschen von Daten üblicherweise ebenfalls als Änderung geliefert wird, da auch gelöschte Daten für die Analyse ggfs. weiter benötigt werden.

Den Teilnehmer:innen ist bewusst, dass durch den Zugriff auf interne Änderungen des Quellsystems das Prinzip des Information Hidings verletzt werden kann. Sie verstehen, wie das Outbox Pattern dazu genutzt werden kann, dieses Problem zu adressieren.

LZ 4-4 - Konnektoren

Die Teilnehmer:innen kennen Tools und Frameworks, die über vordefinierte Konnektoren einen vereinheitlichten Zugriff auf Quellsysteme zur Verfügung stellen. Sie wissen, dass diese Tools und Frameworks vorrangig für die folgenden Aufgaben eingesetzt werden:

- Datenreplikation
- Datenintegration

- Datenvirtualisierung
- Datenorchestrierung
- Metadatenmanagement

Die Teilnehmer:innen kennen Beispiele für diese Tools.

LZ 4-5 - Eigenschaften von Data Ingestion

Die Teilnehmer:innen wissen, das die Data Ingestion unabhängig von der nachfolgenden Datenverarbeitung erfolgt, z.B. hinsichtlich Frequenz des Abrufs.

Die Teilnehmer:innen wissen, das Daten in der Regel kontinuierlich erzeugt werden und für die Verarbeitung in Pakete zerlegt werden können.

Die Teilnehmer:innen kennen die Unterschiede zwischen synchronisierter und nicht-synchronisierter Ingestion und kennen die Probleme, die mit einer synchronisierten Abfolge einhergehen.

Den Teilnehmer:innen ist bewusst, dass Abläufe für die Ingestion skalierbar gestaltet werden sollte, da diese Prozesse häufig große Datenmengen betreffen und ggf. wiederholt ausgeführt werden müssen.

Die Teilnehmer:innen wissen, dass die Ingestion den aktuellen Zustand der zuletzt geänderten Daten (Snapshot) oder die Änderungen selbst (Increments/Events) betreffen kann, mit denen dieser aktuelle Zustand erzeugt wurde. Sie wissen, dass ggfs. eine initiale Bereitstellung (initial Load) der Daten erforderlich ist.

Die Teilnehmer:innen wissen, dass die Datenquellen ihre Daten selbst aktiv liefern können (Push), auf Anfrage einmalig bereitstellen (Pull), oder periodisch bereitstellen können (Poll).

LZ 4-6 - Batch vs Stream Ingestion

Den Teilnehmer:innen sind die Verfahren zur Batch- und Stream-Verarbeitung bekannt. Ihnen ist bewusst, wie Batch Verfahren durch zeitliche (Batch) oder größen-basierte (Micro Batch) Pakete aus einem kontinuierlichen Datenstrom der Quellsysteme erstellt werden können.

Die Teilnehmer:innen kennen den Unterschied zwischen dem ELT und dem ETL Pattern.

Den Teilnehmer:innen ist bei der Stream Ingestion bewusst, dass Daten zu spät ankommen können und dass downstream Systeme dies handhaben können müssen.

LZ 4-7 - Metadaten Ingestion

Die Teilnehmer:innen verstehen die Bedeutung von Metadaten für die Nutzung von Datenquellen. Sie können technische und fachliche Metadaten unterscheiden. Sie wissen, dass Quellsysteme oft umfassend Metadaten zu den von ihnen bereitgestellten Datenquellen zur Verfügung stellen.

Die Teilnehmer:innen wissen dass beispielsweise folgende Informationen in Form von Metadaten für Folgesysteme von großem Nutzen:

- Typ (Tabelle, Bild, Text, …)
- Größe (Bytes, Gigabytes, Terabytes, …)
- Schema und Datentypen

Die Teilnehmer:innen wissen, dass es Verfahren gibt (Schema Inference), um Metadaten aus Daten abzuleiten. Ihnen ist bewusst, dass sich Metadaten über die Nutzungsdauer von Daten ändern können (Schema Evolution) und dass Verfahren existieren, diese Änderungen beim Zugriff auf die Daten zu erkennen.

Die Teilnehmer:innen wissen, dass Data Catalogs zur Verwaltung der Metadaten verwendet werden können. Sie wissen, dass neben den Metadaten der Datenquelle auch die Ausführung der Ingestion in Data Catalogs festgehalten werden.

4.3. Referenzen

[J. Reis 2022]

5. Storage

Dauer: 90 Min.	Übungszeit: 20 Min.

5.1. Begriffe und Konzepte

Speichersysteme, Relationale Datenbanksysteme, NoSQL Datenbanksysteme, Concurrency Control, Versionierung von Daten, Optimierung und Skalierung, Datenmodelle, Data Warehouse, Data Lake, Datei- und Tabellenformate

5.2. Lernziele

LZ 5-1 - Speichersysteme

Die Teilnehmer:innen kennen die folgenden wesentlichen Systeme zum Speichern digitaler Daten und können die jeweiligen Besonderheiten erläutern:

- Filesystem
- Block-Storage System
- Objekt-Storage System

Die Teilnehmer:innen kennen spezielle Kriterien zur Auswahl des passenden Systems für große Datenmengen, wie etwa

- Performanz - Zugriffszeit (Latency), Durchsatz (Throughput)
- Skalierbarkeit
- Kapazität
- Kosten

Die Teilnehmer:innen wissen, dass Speichersysteme unterschiedlich an Systeme zur Verarbeitung der Daten angebunden werden können:

- Direkte Anbindung (Direct Attached Storage - DAS)
- Speichernetzwerk (Storage-Area-Network - SAN)
- Netzwerkspeicher (Network Attached Storage - NAS)

Die Teilnehmer:innen kennen die Vor- und Nachteile des jeweiligen Ansatzes.

LZ 5-2 - Datenbanksysteme

Die Teilnehmer:innen verstehen, dass die Verwaltung von Daten Aufgaben umfasst, die von Speichersystemen nicht direkt unterstützt werden, wie etwa

- Verwaltung der Metadaten zu den Daten
- Vorgabe und Anwendung von Schema-Informationen zu den Daten
- Sicherstellen der Konsistenz der Daten
- Validierung von Daten

- Redundante Datenhaltung speziell für verbesserte Verfügbarkeit und Zugriffszeiten
- Standardisierte, flexible und performante Datenzugriffe
- Feingranulare Zugriffskontrolle
- Skalierbarkeit
- Optimierung (Reduzierung) der Speichernutzung

Die Teilnehmer:innen wissen, dass Datenbanksysteme (DBS) diese Aufgaben übernehmen. Sie verstehen, dass DBS auf der Basis aller Speichersysteme umgesetzt werden können. Sie kennen die folgenden Typen von DBS und sind mit den jeweiligen Besonderheiten vertraut:

- Relationale DBS
- NoSQL DBS
- Key-Value DBS
- (Wide) Column DBS
- Document DBS
- Graph DBS

Die Teilnehmer:innen kennen entsprechende Produkte zu diesen DBS-Typen und ihnen ist bewusst, dass Produkte auch mehrere dieser Typen abbilden können.

LZ 5-3 - Datenbanksysteme für analytische Anwendungen

Die Teilnehmer:innen kennen Systeme, die für spezielle Anwendungsfälle mit großen Datenmengen optimiert sind:

- Searchengines
- Timeseries DBS
- Multidimensionale DBS / Data Marts / Cubes
- In-Memory DBS
- Event Store DBS
- Vector DBS
- Stream Storage

Die Teilnehmer:innen verstehen, dass einige o.g. Aufgaben alleine durch die Verwendung geeigneter Datei- oder Tabellenformate abgedeckt werden können. Sie kennen speziell die folgenden Formate:

- Dateiformate: Avro, ORC, Parquet
- Tabellenformate (Open Table Formats): Delta, Iceberg, Hudi

LZ 5-4 - Concurrency Control

Die Teilnehmer:innen verstehen, dass bei Multi-Threading und -Processing Systemen mit nebenläufiger Verarbeitung sowie redundanter Speicherung von Daten die Konsistenz von Daten explizit sichergestellt werden muss. Ihnen ist das Konzept von ACID Transaktionen und deren Serialisierbarkeit vertraut. Sie wissen, dass ACID Transaktionen auf unterschiedliche Weisen und mit unterschiedlichen Garantien für Konsistenz (Consistency) und Isolation umgesetzt werden können.

Die Teilnehmer:innen wissen, dass ACID Transaktionen üblicherweise von (relationalen) Datenbankmanagementsystemen zur Verfügung gestellt werden, dass sie aber nicht auf diese Systeme beschränkt sind.

Den Teilnehmer:innen ist das alternative Konzept der BASE-Transaktionen sowie speziell der eventuellen Konsistenz (Eventual Consistency) verteilter Anwendungen vertraut.

Die Teilnehmer:innen verstehen, dass es bei der Verarbeitung großer Datenmengen (speziell im Batch) sinnvoll sein kann, auf ACID Transaktionen zu verzichten oder sie zumindest mit schwächeren Garantien für Consistency und Isolation zu verwenden um die Verarbeitungsgeschwindigkeit zu verbessern. Sie verstehen, dass beim Einsatz von Streaming Systemen eine nebenläufige Verarbeitung stattfindet, bei der der Einsatz von Transaktionen sinnvoll sein kann.

LZ 5-5 - Versionierung von Daten

Den Teilnehmer:innen ist bewusst, dass bei veränderlichen Daten nicht nur der aktuelle Zustand, sondern auch die früheren Zustände (speziell für analytische Auswertungen) von Interesse sein können. Sie kennen die folgenden Verfahren für die Versionierung von Daten und verstehen, wie diese etwa in Form von Datenbanksystemen mit den o.g. Speichersystemen kombiniert werden:

- Transaktionen (über die Serialisierbarkeit)
- Versionskontrolle
- Event-Sourcing

LZ 5-6 - Optimierung und Skalierung

Die Teilnehmer:innen kennen übliche Verfahren, um Speichersysteme bei steigender Last zu optimieren und zu skalieren, wie etwa

- Sharding
- Partitionierung (vertikal/horizontal)
- Indexierung
- Reflections
- Caching
- Append Only / Read Only
- Blockierender vs. nicht-blockierender Zugriff

Die Teilnehmer:innen wissen, dass es bei größeren Datenmengen vorteilhaft und notwendig ist die Daten auf mehreren Servern zu verteilen. Sie wissen dabei, dass mehrere Server sowohl das Speichern, Abrufen und Verarbeiten der Daten schneller macht als auch eine Redundanz der Daten bereitstellt, falls ein Server ausfällt.

Die Teilnehmer:innen verstehen, dass Daten oft nicht unbefristet aufbewahrt werden können. Sie kennen Ansätze für Data Retention, wie:

- Wahl der Speichertechnologie abhängig von der Häufigkeit des Datenzugriffs ("Hot" und "Cold" Storage)
- Automatisches Löschen von Daten nach vorgebenen Kriterien, z.B. Ablauf einer Speicherfrist

LZ 5-7 - Datenmodelle für analytische Daten

Den Teilnehmer:innen ist der Umgang mit Datenmodellen generell vertraut. Sie verstehen, wie Datenmodelle für die Arbeit mit analytischen Daten sich sowohl an den Datenmodellen der Quellsysteme als auch an den Anforderungen an die Datenanalyse orientieren müssen.

Die Teilnehmer:innen kennen das Konzept der Normalisierung von Datenmodellen. Sie verstehen, dass für die Arbeit mit analytischen Daten häufig auf die für operative Daten übliche Normalisierung verzichtet wird und vorrangig denormaliserte Datenmodelle in diesem Umfeld verwendet werden.

Die Teilnehmer:innen verstehen, dass Datenmodelle sowohl im Hinblick auf die verwendeten Quellsysteme als auch, noch weit deutlicher, im Hinblick auf die analytischen Anwendungen, Änderungen unterworfen sind. Sie kennen Ansätze, wie mit entsprechenden Änderungen umgegangen werden kann, wie etwa

- Verwendung multidimensionaler Datenmodelle (Star, Snowflake, Data Vault)
- Zerlegung in Teil-Modelle (siehe LZ 10-2 - Begriffsbestimmung Data Mesh - Domain Ownership)
- Automatisierung der Modell-Änderungen (DWH Automation)

LZ 5-8 - Data Warehouse und Data Lake

Die Teilnehmer:innen wissen, dass Data Warehouses und Data Lakes Ansätze sind, um Daten aus verschiedenen Quellsystemen zusammenführen und um einen vereinheitlichten Zugriff auf diese Daten bieten zu können. Sie wissen, dass die beiden Ansätze für OLAP Anwendungen optimiert sind.

Die Teilnehmer:innen wissen, wie Speichersysteme und darauf aufbauende Datenbanksysteme als Grundlage für DWH und DL Systeme verwendet werden.

Die Teilnehmer:innen kennen wesentliche Unterschiede zwischen DWH und DL Systemen, wie etwa

- Ein definiertes Schema (Schema on Write) beim DWH (das sich im Laufe der Zeit ändern kann) gegenüber mehreren parallelen Schemata (etwa mit Schema on Read) beim DL.
- Nur vereinheitlichte Daten im DWH während im DL auch die ursprünglichen Quelldaten (Rohdaten) vorgehalten werden.
- Die bessere Eignung von Data Lakes für Künstliche Intelligenz und Machine Learning Szenarien, da dort noch die ursprünglichen Quelldaten vorhanden sind
- Optimierte Strukturen für den lesenden (analytischen) Zugriff beim DWH gegenüber vereinfachten schreibenden Zugriffen beim DL.
- Begrenzung auf strukturierte Daten beim DWH während DL auch unstrukturierte und semi-strukturierte Daten aufnehmen können.
- Hoher Aufwand für die Integration neuer Datenquellen beim DWH während neue Datenquellen in den DL direkt aufgenommen werden können.
- Hoher Aufwand für die Vereinheitlichung von Daten beim lesenden (analytischen) Zugriff im DL während dies beim DWH in deutlich geringerem Umfang erforderlich ist.

Die Teilnehmer:innen kennen Lösungsansätze, um in DL Systemen den Aufwand für die Vereinheitlichung beim lesenden Zugriff zu reduzieren:

- Aufteilen des DL in Bereiche unterschiedlicher Datenqualität – etwa Bronze, Silber und Gold –, wobei die Rohdaten im Bronze-Bereich und die vereinheitlichten, gut analysierbaren Daten im Gold-Bereich zu finden sind (Data Lakehouse).

- Technische Aspekte der Vereinheitlichung direkt bei der Eingangsverarbeitung erledigen (einheitliche Zeichensätze, Null-Values, Datumsformate, …)
- Modularisierung des DL etwa über die Verwendung von DDD.

5.3. Referenzen

[starke]

6. Query und Processing

Dauer: 60 Min.	Übungszeit: 15 Min.

6.1. Begriffe und Konzepte

Analytische Queries, Data in Rest, Data in Motion, SQL, OLAP, Stream Processing, Query-Engine, Performance, verteilte Daten, Query Optimierung

6.2. Lernziele

LZ 6-1 - Analytische Queries

Die Teilnehmer:innen verstehen, dass sich analytische Queries stets lesend auf eine größere Anzahl von Datensätzen beziehen. Sie wissen, dass diese Daten für die Analyse sowohl komplett abgespeichert (Data in Rest) als auch im Fluss (Data in Motion) sein können.

Die Teilnehmer:innen kennen Query-Ausdrücke für analytische Anwendungen, wie etwa

- Aggregationen (Sum, Count, Avg, …)
- Fenstern/Windows (Over Partition by)
- OLAP Operationen (Slice/Dice, Drill up/down/…)
- Umformungen (Pivot, Transpose, …)
- Modularisierung (Views, Common Table Expressions, Stored Procedures, User Defined Functions)

LZ 6-2 - Query Programmiermodelle

Die Teilnehmer:innen wissen, dass unterschiedliche Programmiermodelle für analytische Queries eingesetzt werden. Ihnen sind wesentliche Programmiermodelle bekannt und sie können beurteilen, für welche Aufgaben sie speziell geeignet sind:

- SQL
- OLAP
- Spark
- Map-Reduce
- Stream Processing
- Dataframes
- Numerical/Statistical

Den Teilnehmer:innen ist bewusst, dass die Art des benötigten Datenzugriffs zentral für die Wahl einer Datenarchitektur ist.

Die Teilnehmer:innen kennen für jedes dieser Programmiermodelle Beispiele zu Frameworks, Libraries und Tools für die Verarbeitung analytischer Queries.

LZ 6-3 - Query Verarbeitung & Optimierung

Die Teilnehmer:innen wissen, dass die Verarbeitung der Queries von der sog. Query-Engine übernommen

werden. Die Teilnehmer:innen wissen, dass diese Engines sehr eng an ein verwendetes Datenbanksystem gekoppelt sein können, dass es aber auch Engines gibt, die davon unabhängig arbeiten.

Den Teilnehmer:innen ist bekannt, dass die Verarbeitung von Queries in den folgenden Schritten erfolgt:

- Parsen der Query-Statements
- Erstellen und Optimieren des Ausführungsplans
- Ausführen der Query gemäß Ausführungsplan
- Rückgabe der Ergebnisse

Die Teilnehmer:innen verstehen, dass analytische Queries i.d.R. nicht komplett im Hauptspeicher verarbeitet werden können und dass analytische Queries oft auf verteilten Daten operieren müssen. Sie wissen, dass eine performante Ausführung von der Engine insbesondere durch Lastverteilung auf parallel arbeitende Worker gewährleistet werden kann.

Den Teilnehmer:innen ist bewusst, dass Unterschiede in Ausführungsplänen zu drastischen Laufzeit-Unterschieden bei der Ausführung von Queries führen können. Sie kennen Beispiele wie Full Table Scans oder Full Outer Joins (Kreuzprodukte), die auf ungünstige Ausführungspläne hinweisen können.

Die Teilnehmer:innen kennen gängige Verfahren zur Optimierung (analytischer) Queries, wie etwa die Vorgabe günstiger Join-Beziehungen im Star-Schema.

6.3. Referenzen

[starke]

7. Transformation

Dauer: 90 Min.	Übungszeit: 15 Min.

7.1. Begriffe und Konzepte

Datentransformation, Data Cleansing, Duplikaterkennung, Datenanonymisierung, Normalisierung/Denormalisierung, Data Profiling, Data Lineage, Qualitätsstufen, Batch Verarbeitung, Stream Verarbeitung

7.2. Lernziele

LZ 7-1 - Begriffsbestimmung (Daten-)transformation

Die Teilnehmer:innen verstehen, dass Daten aus operativen Systemen oftmals aufbereitet werden müssen, um für die Analyse nutzbar zu sein. Sie wissen, dass dieser Schritt als (Daten-)Transformation bezeichnet wird.

Die Teilnehmer:innen verstehen, dass Transformationen in Form von Queries beschrieben werden können, dass bei steigender Komplexität aber einzelne Queries zu unverständlich werden und Abfolgen von Queries (Data Pipeline siehe LZ 9-1 - Begriffsbestimmung Data Pipelines) besser für die Umsetzung von Transformationen geeignet sind. Sie wissen, dass Transformationen üblicherweise vor dem Speichern der Daten erfolgen, es mitunter aber sinnvoll sein kann, Transformationen erst vor der Präsentation auszuführen.

LZ 7-2 - Anwendungsfälle

Die Teilnehmer:innen kennen übliche Aufgaben, die von Transformationen übernommen werden, wie speziell

- Umwandeln der Datenrepräsentation von Quell- in Zielformate
- Bereinigen der Daten (Data Cleansing)
- Identifizieren und Entfernen von Duplikaten
- Ergänzung der Daten (Enhancement)
- Vereinheitlichung/Standardisierung von Daten mehrerer Quellsysteme, z.B. Lokalisierung
- Reduktion des Datenumfangs, Komprimierung
- Anonymisierung, Pseudonomisierung, Verschlüsselung

Die Teilnehmer:innen verstehen, wie Transformationen Verfahren des Query und Processing (siehe LZ 6-3 - Query Verarbeitung & Optimierung) nutzen, um auf die Daten in den Speichersystemen (siehe LZ 5-1 - Speichersysteme) zuzugreifen.

Die Teilnehmer:innen verstehen, wie Transformationen bei verteilten Daten parallel ausgeführt werden.

Die Teilnehmer:innen verstehen die Bedeutung von Schemas und Datenmodellen für die Umsetzung von Transformationen.

LZ 7-3 - Typische Transformationen

Die Teilnehmer:innen kennen typische Transformationen für die Aufbereitung analytischer Daten, wie etwa

- Normalisierung / Denormalisierung von Datenstrukturen - Star/Snowflake Schema, Data Vault
- Zuordnung von künstlichen IDs (surrogate keys) zu IDs der Quellsysteme
- (Multidimensionale) Aggregation (Cubes)
- Umwandlung zwischen flachen und geschachtelten Datenstrukturen
- Historisieren (Slowly Changing Dimensions)
- Ersetzen / Aussortieren unwahrscheinlicher Werte / Datensätze (Outlier)

LZ 7-4 - Staging Area

Den Teilnehmer:innen ist das Konzept der Staging Area vertraut. Sie wissen, dass

- analytische Daten aus Datenquellen in einer Staging Area gesammelt und aufbereitet werden, bevor sie für die Analyse zur Verfügung gestellt werden
- Staging Areas typischerweise für die Batch Verarbeitung von Daten zum Befüllen eines DWH Systems eingesetzt werden
- das tatsächliche Befüllen des DWH keine komplexen Operationen mehr erfordert und üblicherweise ganz oder gar nicht erfolgt (atomar)
- die Staging Area sowohl das vollständige Verarbeiten der Daten einer Datenquelle (initial load) als auch das Verarbeiten der Änderungsdaten (incremental load) unterstützen muss.
- die Staging Area immer nur genau die Änderungsdaten erhält, die seit der letzten Batchverarbeitung in den Quellsystemen angefallen sind.

Die Teilnehmer:innen wissen, welche Transformationen typischerweise in einer Staging Area angewendet werden.

LZ 7-5 - Robuste Transformationen

Die Teilnehmer:innen verstehen, dass die Umsetzung von Transformationen oft fragil ist, da sich die zu verarbeitenden Daten von Ausführung zu Ausführung im Hinblick auf Umfang, Format oder Bedeutung ändern können. Die Teilnehmer:innen kennen Lösungsansätze, um Transformationen robust zu gestalten:

- Vermeiden der Umsetzung von Geschäftslogik
- Gewährleisten der Wiederholbarkeit - Idempotente Transformationen
- Verwendung von Standards
- Einsatz von Verfahren zur Code-Generierung für eine automatisierte Anpassung von Transformationen bei Schema-Änderungen
- Auswahl geeigneter Programmiermodelle

Die Teilnehmer:innen kennen Verfahren zum Data Profiling, um die zu erwartenden Daten für die Umsetzung der Transformationen vorab zu analysieren.

Die Teilnehmer:innen kennen Verfahren für die Überwachung (Monitoring) der Ausführung von Transformationen.

Die Teilnehmer:innen kennen das Konzept der Data Lineage für das Nachvollziehen der Abhängigkeiten zwischen Ein- und Ausgangsdaten von Transformationen.

LZ 7-6 - Qualitätsstufen

Die Teilnehmer:innen verstehen, dass über eine schrittweise Transformation eingehender Daten Qualitätsstufen (Quality-Gates) definiert werden können. Sie verstehen, dass damit je nach Art der Analyse Daten einer höheren (Clean Data) oder einer niedrigeren (Raw Data) herangezogen werden können.

Die Teilnehmer:innen verstehen, dass Staging Areas (siehe LZ 7-4 - Staging Area) ein Quality Gate darstellen, mit dem sichergestellt wird, dass nur qualitätsgesicherte Daten zur Analyse in einem Data Warehouse bereitgestellt werden.

Die Teilnehmer:innen verstehen, dass in einem Data Lake (siehe LZ 5-8 - Data Warehouse und Data Lake) Daten unterschiedlicher Qualitätsstufen gemeinsam aufbewahrt werden und dass es daher sinnvoll sein kann, Zonen unterschiedlicher Datenqualität (etwa Bronze, Silber, Gold) innerhalb von Data Lakes festzulegen (Medallion Architecture).

LZ 7-7 - Batch Verarbeitung

Die Teilnehmer:innen verstehen das Verfahren der Batch Verarbeitung (Batch Processing) für die Transformation von Daten. Sie wissen, dass die Transformation ein Teil der übergreifenden ETL (Extract-Transform-Load) oder ELT (Extract-Load-Transform) Prozesse ist. Ihnen ist bewusst, dass

- die Batch Verarbeitung üblicherweise dazu dient, eine größere Anzahl Datensätze aus mehreren Datenquellen zu verarbeiten und die Ergebnisse in eine Datensenke (z.B. Staging Area) zu schreiben
- alle (initial/complete load) oder nur die letzten Änderungen (incremental load) im Batch verarbeitet werden können
- die Batch-Verarbeitung idR zeitlich geplant (scheduled) und wenige Male je Tag ausgeführt wird
- eine Batch-Verarbeitung idR alle Datensätze verarbeitet oder (im Fehlerfalle) keine

Die Teilnehmer:innen verstehen das Verfahren der Micro-Batch Verarbeitung von Daten. Sie wissen, dass Micro-Batches im Gegensatz zu Batches auch dann angestoßen werden, wenn ausreichend Datensätze in der Datenquelle angefallen sind.

LZ 7-8 - Stream Verarbeitung

Die Teilnehmer:innen verstehen das Verfahren der Stream Verarbeitung (Stream Processing) von Daten. Sie wissen, dass Stream Verarbeitung auf der Basis des Event Streamings aufsetzt.

Die Teilnehmer:innen wissen, dass bei der Stream Verarbeitung

- die Daten mehrerer Streams miteinander zu einem weiteren Stream kombiniert werden können.
- Datensätze (etwa fehlerhafte oder unvollständige) im Stream voneinander getrennt und separat (in unterschiedlichen Streams) weiterverarbeitet werden können.

Die Teilnehmer:innen verstehen, warum das Schreiben von Daten aus einem Datenstrom meist idempotent gestaltet wird.

Die Teilnehmer:innen können zustandslose (stateless) und zustandsbehaftete (stateful) Stream Verarbeitung unterscheiden.

Die Teilnehmer:innen verstehen, dass Operationen nicht auf allen Datensätzen eines Streams erfolgen können, sondern immer nur auf einzelnen oder einer Gruppe von aufeinanderfolgenden Datensätzen. Sie kennen dazu das Konzept der Fenster (Window) Funktionen.

Die Teilnehmer:innen kennen Frameworks oder Tools für die Stream Verarbeitung, wie Kafka Streams.

7.3. Referenzen

[starke]

8. Serving Data

Dauer: 60 Min.	Übungszeit: 15 Min.

8.1. Begriffe und Konzepte

Data Analytics, Business Intelligence (BI), Reporting Tools, Machine Learning, Reverse ETL, Datenvisualisierung, Vertrauenswürdigkeit von Daten, Rohdaten, Aggregationen

8.2. Lernziele

LZ 8-1 - Anwendungsfälle

Die Teilnehmer:innen können die wesentlichen Anwendungsfälle für die Nutzung analytischer Daten unterscheiden:

- Basis für Data Analytics und Business Intelligence (BI)
- Test- und Trainingsdaten für das Erstellen und Verbessern von Maschine Learning Modellen
- Bereitstellen der Ergebnisse analytischer Auswertungen für die Verwendung in operativen Anwendungen (Reverse ETL)

LZ 8-2 - Repräsentation von Massendaten

Die Teilnehmer:innen wissen wie die Verfahren des Query-Processings wie Aggregation oder Cubes für die Darstellung von Massendaten verwendet werden können. Ihnen ist bewusst, dass bei analytischen Daten die Vertrauenswürdigkeit eine zentrale Rolle spielt. Sie wissen, dass daher neben den Daten selbst auch Information zu deren Schema, zum Umfang, zur Aktualität, zu den Datenquellen oder zur Qualität angeboten werden können.

Die Teilnehmer:innen kennen Tools zur Visualisierung von Massendaten. Sie kennen verfügbare Arten von Diagrammen (Heatmap, Ranking, Top, …) und können diese speziell im Hinblick auf Kosten/Nutzen beurteilen. Sie kennen BI-/Reporting Tools und wissen, wie diese für explorative (Slice & Dice, Drill, etc) oder descriptive Analysen eingesetzt werden können. Die Teilnehmer:innen wissen, dass auf Basis dieser Tools analytische Anwendungen wie Dashbords umgesetzt werden können.

LZ 8-3 - Modularisierung

Die Teilnehmer:innen wissen, dass analytische Daten vielfältig und unübersichtlich sein können. Sie kennen das Konzept des Datenprodukts und wissen, dass die Datenprodukte verwendet werden können, um analytische Daten fachlich zu zerlegen. Sie kennen speziell deren Verwendung im Data Mesh Ansatz (siehe LZ 10-1 - Nachteile zentraler Datenarchitekturen).

LZ 8-4 - Data Analytics und Business Intelligence

Den Teilnehmer:innen sind Data Analytics und BI Anwendungen vertraut. Sie kennen Anwendungsfälle für die Analyse operativ anfallender Daten für die Unternehmenssteuerung.

Die Teilnehmer:innen kennen Frameworks und Tools für die Umsetzung von Data Analytics und BI Anwendungen. Ihnen ist bewusst, das diese Anwendungen üblicherweise direkten Zugriff auf die zugrundeliegenden Speichersysteme benötigen.

LZ 8-5 - Machine Learning

Die Teilnehmer:innen wissen, wie Machine Learning Modelle für den Einsatz im Unternehmen auf Basis operativ anfallender Daten trainiert, verbessert und getestet werden. Sie kennen Anwendungsfälle für die Nutzung solcher Modelle in operativen Anwendungen. Ihnen ist bewusst, das ML Verfahren, anders als Data Analytics, auf einen Extract der vollständigen Rohdaten angewiesen sind und spezielle Repräsentationen (Vektoren) benötigen. Sie kennen die häufige Notwendigkeit spezialiserter Hardware (GPUs).

LZ 8-6 - Reverse ETL

Den Teilnehmer:innen ist das Konzept der Reverse ETL vertraut. Sie kennen Anwendungsfälle für die Integration von vereinheitlichten Daten oder von Ergebnissen analytischer Auswertungen in operative Anwendungen.

Die Teilnehmer:innen wissen, dass Reverse ETL Tools mit Konnektoren für gängige operative Anwendungen existieren, die die Umsetzung entsprechender Integrationslösungen deutlich erleichtern. Sie kennen Beispiele für diese Tools.

9. Data Pipelines

Dauer: 90 Min.	Übungszeit: 30 Min.

9.1. Begriffe und Konzepte

Data Pipelines, Qualitätskriterien, Monitoring, Edge-Computing

Data Pipelines, Effizienz, Batch (ETL und ELT), Microbatch, Stream Verarbeitung, Eventgetriebene Verarbeitung, Qualitätskriterien, Building Blocks, Technologien, Betrieb

9.2. Lernziele

LZ 9-1 - Begriffsbestimmung Data Pipelines

Die Teilnehmer:innen wissen, dass Data Pipelines dazu dienen analytische Daten durch die einzelnen Phasen des Data Engineerings zu bewegen. Sie wissen was typischerweise in den einzelnen Phasen geschieht (siehe LZ 1-7 - Lebenszyklus analytischer Daten). Sie kennen die wesentlichen Eigenschaften von Data Pipelines wie:

- Isolation
- Unabhängigkeit
- Einfache Einrichtung und Betreibbarkeit
- hohe Verfügbarkeit
- Erweiterbarkeit
- Skalierbarkeit

LZ 9-2 - Anwendungsgebiete von Data Pipelines

Die Teilnehmer:innen kennen typische Anwendungsgebiete von Data Pipelines:

- Data Engineering
- Analytics/ML Processing
- Delivery

LZ 9-3 - Arten von Data Pipelines

Die Teilnehmer:innen wissen um die verschiedenen Arten von Data Pipelines. Sie wissen in welchen Situationen diese zum Einsatz kommen und wann sie vorteilhaft sind.

Die Teilnehmer:innen kennen die Unterschiede zwischen Batch (ETL und ELT), Microbatch, Stream und Eventgetriebener Verarbeitung.

LZ 9-4 - Qualitätskriterien für Data Pipelines

Die Teilnehmer:innen kennen maßgebliche Qualitätskriterien, die die Güte einer Data Pipeline beschreiben, wie etwa:

- Durchsatz

- Zuverlässigkeit
- Latenz
- Stabilität

Die Teilnehmer:innen kennen Ansätze wie Idempotenz, Caching, Parallelisierung, Edge Computing um die Qualität von Data Pipelines zu verbessern.

LZ 9-5 - Building Blocks von Data Pipelines

Die Teilnehmer:innen wissen aus welchen Building Blocks eine Data Pipelines besteht und in welchen Phasen des Lebenszyklus diese verwendet werden. Sie kennen Beispiele dafür, wissen wann diese Building Blocks für einen Anwendungsfall geeignet sind und wie diese kombiniert werden können.

- Konnektoren
- Werkzeuge zur Ablaufsteuerung und Orchestrierung
- Kostengünstige Speichersysteme für hohe Datenvolumen
- Datenkataloge
- Werkzeuge zur Datentransformation
- Report-Generatoren

LZ 9-6 - Technologien und Plattformen für Data Pipelines

Die Teilnehmer:innen wissen, wie Data Pipelines auf der Basis von Technologien, wie SQL, Python Dataframes oder Spark, manuell erstellt werden können. Sie kennen zudem auch integrierte Datenplattformen zur Erstellung und dem Management von Data Pipelines, wie z.B.:

- Databricks
- Fivetran
- Skyvia

LZ 9-7 - Betrieb von Data Pipelines

Den Teilnehmer:innen ist bewusst, dass für den Betrieb von Data Pipelines speziell die folgenden Themen bedacht werden müssen:

- Monitoring
- Abhängigkeiten (Data Lineage)
- Metadaten
- Late arriving data
- Orchestrierung
- Schemaänderungen

9.3. Referenzen

[H. Varshney 2023], [E. Levy 2021], [B. Singhal 2022]

10. Data Mesh

Dauer: 120 Min.	Übungszeit: 30 Min.

10.1. Begriffe und Konzepte

Dezentralisierung, Data Mesh, Domain-driven Design, Team Topologies, Data as a Product, Product Thinking, Self-serve Data Platform, Federated Computational Governance, Top Down vs. Bottom Up Realisierung

10.2. Lernziele

LZ 10-1 - Nachteile zentraler Datenarchitekturen

Die Teilnehmer:innen kennen die Probleme zentralisierter Datenarchitekturen. Sie verstehen, aufgrund welcher Probleme zentrale Data Teams nicht gut skalieren. Sie verstehen auch, warum dezentrale Domain-Teams nicht gut zu zentralen Data Teams passen.

LZ 10-2 - Begriffsbestimmung Data Mesh

Die Teilnehmer:innen wissen, dass der Data Mesh Ansatz darauf abzielt, dezentrale Datenarchitekturen zu ermöglichen.

Die Teilnehmer:innen wissen, dass der Data Mesh Ansatz auf den folgenden vier Säulen beruht:

- Domain Ownership
- Data as a Product
- Self-serve Data Platform
- Federated Computational Governance

LZ 10-3 - Domain Ownership

Die Teilnehmer:innen kennen relevante Konzepte des Strategic Domain-Driven Design, wie Domains, Subdomains, Bounded Contexts und Context Mapping Patterns.

Die Teilnehmer:innen verstehen die Bedeutung sowie das Problem von Polysemen als geteilte Konzepte zwischen unterschiedlichen Domains.

Die Teilnehmer:innen verstehen, dass die Verantwortung für analytische Daten und deren Qualität von den zentralen Data Teams bei einem Data Mesh an die einzelnen Domain Teams übergeht. Sie wissen, dass die Domain Teams ihre eigenen analytischen Daten verwalten. Sie wissen auch, dass die Domain Teams die analytischen Daten ihrer Domain ihrem Unternehmen zur Verfügung stellen müssen, diese Daten sollten allerdings nur in Form klar definierter Schnittstellen (Datenprodukt) bereitgestellt werden.

LZ 10-4 - Data as a Product

Die Teilnehmer:innen wissen, dass Data as a Product bedeutet, die Idee der Produktentwicklung und Vermarktung auf Daten zu übertragen.

Die Teilnehmer:innen verstehen, dass Domain Teams ihre Daten anderen Teams in Form von Datenprodukten zur Verfügung stellen. Sie verstehen, dass Datenprodukte First-Class Citizens der Systemarchitektur darstellen, vergleichbar mit UI und API Komponenten.

Sie können die drei Archetypen von Datenprodukten in einem Data Mesh unterscheiden:

- source-aligned
- aggregate
- consumer-aligned

Die Teilnehmer:innen kennen die Charakteristiken von Datenprodukten:

- Discoverable
- Addressable
- Trustworthy
- Self-descriptive
- Secure
- Understandable
- Interoperable
- Natively accessible
- Valuable on their own

Die Teilnehmer:innen kennen unterschiedliche Formen von Datenprodukten wie Datasets, Reports, Machine Learning Features bzw. kompletten ML Modellen.

Die Teilnehmer:innen wissen, dass Data Contracts im Sinne einer API der Implementierung von Datenprodukten dienen.

LZ 10-5 - Self-serve Data Platform

Die Teilnehmer:innen verstehen die wesentlichen Konzepte einer Domain-agnostischen Data Platform. Sie wissen, dass diese Plattform vom Data Platform Team verantwortet wird.

Die Teilnehmer:innen wissen, dass die Data Platform den Domänen Teams ermöglicht, ihre Datenprodukte eigenständig zu erstellen.

Die Teilnehmer:innen kennen typische Komponenten einer Data Platform, wie Storage, Data Pipelines, Data Catalogs, Access Management, Monitoring, Visualisation. Ihnen sind bestehende Lösungen speziell der großen Cloud Provider bekannt.

LZ 10-6 - Federated Computational Governance

Die Teilnehmer:innen wissen, dass die sog. Governance Group (im Sinne einer Gilde) aus Vertretern der einzelnen Domain Teams und dem Data Platform Team besteht.

Die Teilnehmer:innen wissen, dass sich die Governance Group auf global gültige Regel und Vorgaben des Data Mesh Ecosystems, speziell hinsichtlich Interoperability, Security und Compliance verständigt.

Die Teilnehmer:innen verstehen, wie die Self-Serve Data Platform der Verwaltung, Automatisierung und Durchsetzung der global vereinbarten Regeln und Vorgaben dient. Sie wissen, dass die Verantwortung dafür bei den Domain Teams liegt.

LZ 10-7 - Top Down vs. Bottom up Realisierung

Die Teilnehmer:innen wissen, dass der Data Mesh Ansatz sowohl Top down wie auch Bottom up realisiert werden kann. Sie verstehen, dass sich der Top Down Ansatz vor allem für Situationen eignet, in denen bestehende monolithische Datensysteme (Data Warehouse, Data Lake) zerlegt werden sollen, während sich der bottom up Ansatz eher an bestehende Domain Teams richtet, welche ihre Daten dezentral für analytische Zwecke zur Verfügung stellen wollen und einen direkten Nutzen für die Domain Teams bieten wollen.

10.3. Referenzen

[J. Christ et al. 2018], [J. Majchrzak 2022], [Z. Dehghani 2023]

11. Querschnittsthemen

Dauer: 120 Min.	Übungszeit: 20 Min.

11.1. Begriffe und Konzepte

Data Management, Data Governance, Data Contracts, Data Ownership, Datenqualität, Datensicherheit, Anonymisierung, Pseudonymisierung, Personalisierung, Metadaten, Verantwortlichkeit, DataOps

11.2. Lernziele

LZ 11-1 - Übergreifende Themen zum Datenlebenszyklus

Die Teilnehmer:innen verstehen, dass insbesondere die folgenden Themen übergreifend über die einzelnen Phasen des Datenlebenszyklus hinweg beachtet werden müssen: - Data Governance - Datensicherheit - Datenqualität - Betriebsaspekte (DataOps)

LZ 11-2 - Data Governance

Die Teilnehmer:innen verstehen, dass Data Governance eine Vielzahl von Prinzipien und Praktiken umfasst, die sicherstellen, dass Daten im gesamten Lebenszyklus von der Erstellung bis zur Löschung sicher, effektiv und effizient verwaltet werden. Die Teilnehmer:innen kennen lokale und globale Richtlinien für alle Phasen des Datenlebenszyklus und wissen wie sie diese festlegen und dokumentieren.

LZ 11-3 - Data Stewardship und Ownership

Die Teilnehmer:innen verstehen die Notwendigkeit von dedizierten Verantwortlichen für Datenpflege und -verwaltung. Dies beinhaltet die Definition von Verantwortlichkeiten für Datenqualität, Datenschutz und andere Governance-Aspekte. Die Teilnehmer:innen können die verschiedenen Rollen und Verantwortungsbereiche in der Data Governence nennen und beschreiben.

LZ 11-4 - Datenzugriff und -berechtigungen

Die Teilnehmer:innen kennen die Notwendigkeit einer Definition, wer Zugang zu welchen Daten hat und unter welchen Bedingungen. Die Teilnehmer:innen wissen, wie Daten auditiert und gemonitored werden können und kennen die Wichtigkeit der Auditierung für die Bereiche Security, Privacy, Benutzbarkeit und Zuverlässigkeit.

LZ 11-5 - Datenschutz, Compliance, Datensicherheit

Die Teilnehmer:innen kennen Vorgehensweisen und Mechanismen zur Implementierung von Sicherheitsmaßnahmen zum Schutz sensibler Daten.

Die Teilnehmer:innen kennen gesetzliche und regulatorischer Anforderungen wie die Datenschutz-Grundverordnung (DSGVO) in der EU, das Bundesdatenschutzgesetz (BDSG) in Deutschland oder den California Consumer Privacy Act (CCPA) in den USA.

Die Teilnehmer:innen kennen Mechanismen zum Schutz von Daten vor unbefugtem Zugriff, Verlust oder Diebstahl. Dies umfasst Verschlüsselungsmaßnahmen, Firewalls und anderen Sicherheitstechnologien. Die Teilnehmer:innen verstehen, welche Daten anonymisiert oder pseudonymisiert werden müssen und kennen die jeweiligen Nutzungsszenarien. Die Teilnehmer:innen kennen gängige Praktiken zur Anonymisierung und Pseudonymisierung von Daten und wissen, in welchen Fällen diese angewendet werden.

LZ 11-6 - Qualität von Daten

Die Teilnehmer:innen verstehen, warum die Gewährleistung von Datenqualität bei analytischen Daten eine besondere Herausforderung ist. Sie verstehen dass sich die Data Governance darum kümmert die Datenqualität zu gewährleisten. Die Teilnehmer:innen kennen typische Qualitätsmerkmale von Daten und können diese erläutern.

- Auffindbarkeit
- Benutzbarkeit (Verständlichkeit, Korrektheit)
- Zuverlässigkeit, Verfügbarkeit

LZ 11-7 - Data Contracts

Die Teilnehmer:innen wissen, dass ein Data Contract die Struktur, das Format, die Semantik, die Qualität und die Nutzungsbedingungen für den Datenaustausch zwischen einem Datenanbieter und seinen Konsumenten definiert. Ihnen ist bewusst, dass es ist damit ein zentrales Werkzeug ist, damit sich Teams über Daten-Schnittstellen verständigen können und somit Stabilität, Datenqualität und Nachvollziehbarkeit in der Datenarchitektur gewährleisten. Den Teilnehmer:innen ist bewusst, dass Data Contracts daher trotz des Namens nicht im Sinne eines konkreten Vertrages zwischen einem Datenanbieter und einem Konsumenten zu verstehen sind.

Die Teilnehmer:innen kennen Standards für die Definition von Data Contracts.

LZ 11-8 - Metadaten

Die Teilnehmer:innen verstehen die Bedeutung von Metadaten im Lebenszyklus der Daten und kennen Möglichkeiten, diese zu erheben und festzuhalten. Sie kennen verschiedene Kategorien von Metadaten und können diese unterscheiden:

- Business Metadaten
- Technische Metadaten
- Betriebs-Metadaten
- Referenzielle Metadaten

LZ 11-9 - Betriebsaspekte

Die Teilnehmer:innen kennen Vor- und Nachteile von Cloud, On-Premise und Software-as-a-Service. Sie kennen Edge Computing und Content Delivery Networks. Die Teilnehmer:innen kennen die Grundkonzepte von Containern und Kubernetes. Die Teilnehmer:innen können Maßnahmen für einen kosteneffizienten Betrieb bennenen. Die Teilnehmer:innen wissen, wie sie ein effektive Monitoring- und Alerting für datenzentrierte Anwendungen sicherstellen können. Sie verstehen, welche Leistungsmetriken überwacht werden sollten und wie Anomalien erkannt werden können. Den Teilnehmer:innen ist die Notwendigkeit eines Daten-Lifecycle-Managements mit automatisierten Verfahren und Regeln zur Qualitätssicherung, zur Transformation, zur Archivierung oder zur Löschung von Daten bewusst. Die Teilnehmer:innen kennen den Begriff DataOps und verstehen, wie damit agile Methoden auf den Bereich des Data Engineerings übertragen werden.

11.3. Referenzen

[J. Reis 2022]

Referenzen

Dieser Abschnitt enthält Quellenangaben, die ganz oder teilweise im Curriculum referenziert werden.

A

- [R. Agarwal] R. Agarwal: Kafka Connectors — All you need to know to start using connectors in Kafka. https://medium.com/@the_infinity/kafka-connectors-all-you-need-to-know-to-start-using-connectors-in-kafka-d905cf8a371c, [Online; Stand: 2.06.2023].

B

- [F. Bachmann et al. 2000] F. Bachmann, L. Bass, J. Carriere, P. Clements, D. Garlan, J. Ivers, R. Nord, and R. Little: Software architecture documentation in practice: Documenting architectural layers. tech. rep., Carnegie-Mellon University Pittsburgh PA Software Engineering Inst, 2000.
- [M. Bornstein 2020] M. Bornstein, J. Li, and M. Casado: Emerging Architectures for Modern Data Infrastructure. https://a16z.com/emerging-architectures-for-modern-data-infrastructure, 2020.

C

- [E. F. Codd 1990] E. F. Codd: The relational model for database management: version 2. Addison-Wesley Longman Publishing Co., Inc., 1990.
- [R. Castagna 2022] R. Castagna: Strukturierte und unstrukturierte Daten: Die Unterschiede. https://www.computerweekly.com/de/feature/Strukturierte-und-unstrukturierte-Daten-Die-Unterschiede, November 2022, [Online; Stand: 2.06.2023].
- [B. Carnes 2020] B. Carnes: Basic SQL Commands - The List of Database Queries and Statements You Should Know. https://www.freecodecamp.org/news/basic-sql-commands/, 2020.
- [J. Christ et al. 2018] J. Christ, L. Visengeriyeva, S. Harrer: Data Mesh Architecture - Data Mesh From an Engineering Perspective. https://www.datamesh-architecture.com, 2022.

D

- [K. Dutta 2015] K. Dutta and M. Jayapal: Big Data Analytics for Real Time Systems. 02 2015.
- [Z. Dehghani 2023] Z. Dehghani, J. Christ, and S. Harrer: Data Mesh: Eine dezentrale Datenarchitektur entwerfen. O'Reilly Media, Inc., 2023.

E

- [W. Eckerson 2015] W. Eckerson: Which Data Warehouse Automation Tool is Right for You?. https://www.eckerson.com/register?content=which-data-warehouse-automation-tool-is-right-for-you, 2015.

G

- [C. Giebler et al. 2021] C. Giebler, C. Gröger, E. Hoos, R. Eichler, H. Schwarz, and B. Mitschang: The data lake architecture framework. BTW 2021, 2021.
- [M. Grellmann 2022] M. Grellmann: Die sechs Arten der Datenanalyse. https://martin-grellmann.de/die-sechs-arten-der-datenanalyse, 2022.

I

- [W. H. Inmon 2005] W. H. Inmon: Building the data warehouse. John wiley & sons, 2005.

K

- [R. Kimball 2011] R. Kimball and M. Ross: The data warehouse toolkit: the complete guide to dimensional modeling. John Wiley & Sons, 2011.
- [J. Klump et al. 2021] J. Klump, L. Wyborn, M. Wu, J. Martin, R. R. Downs, A. Asmi, et al.: Versioning data is about more than revisions: A conceptual framework and proposed principles. 2021.
- [J. Kreps et al. 2011] J. Kreps, N. Narkhede, J. Rao, et al.: Kafka: A distributed messaging system for log processing. in Proceedings of the NetDB. vol. 11, pp. 1–7, Athens, Greece, 2011.
- [J. Kutay] J. Kutay: Change Data Capture (CDC): What it is and How it Works. https://www.striim.com/blog/change-data-capture-cdc-what-it-is-and-how-it-works/, [Online; Stand: 2.06.2023].

L

- [E. Levy 2021] E. Levy: Batch vs Stream vs Microbatch Processing: A Cheat Sheet. https://www.upsolver.com/blog/batch-stream-a-cheat-sheet, 2021.
- [D. Linstedt 2015] D. Linstedt and M. Olschimke: Building a scalable data warehouse with data vault 2.0. Morgan Kaufmann, 2015.
- [S. Luber 2018] S. Luber and N. Litzel: Was ist Data Profiling?. https://www.bigdata-insider.de/was-ist-data-profiling-a-691538/, 2018.

M

- [J. Majchrzak 2022] J. Majchrzak, S. Balnojan, M. Siwiak, M. Sieraczkiewicz: Data Mesh in Action. Manning Publication, 2022.
- [P. Mhatre 2021] P. Mhatre: Data Warehouse vs Data Vault vs Data Lake vs Delta Lake vs Data Fabric vs Data Mesh. https://medium.com/@mhatrep/data-warehouse-vs-data-vault-vs-data-lake-vs-delta-lake-vs-data-fabric-vs-data-mesh-1cf4c8991961, 2021.

P

- [P. Pääkkönen 2015] P. Pääkkönen and D. Pakkala: Reference Architecture and Classification of Technologies, Products and Services for Big Data Systems. Big Data Research, vol. 2, no. 4, pp. 166–186, 2015.
- [D. L. Parnas 2002] D. L. Parnas: The Secret History of Information Hiding. pp. 398–409. Berlin, Heidelberg, Springer Berlin Heidelberg, 2002.
- [T. B. Pedersen 2009] T. B. Pedersen: Multidimensional Modeling. pp. 1777–1784. Boston, MA: Springer US, 2009.

R

- [C. Richardson 2018] C. Richardson, Microservices patterns: with examples in Java. Simon and Schuster, 2018.
- [J. Reis 2022] J. Reis and M. Housley: Fundamentals of Data Engineering. O'Reilly Media, Inc., 2022.

S

- [Y. Sharvit 2022] Y. Sharvit: Data-oriented programming unlearning objects. Manning, 2022.
- [B. Singhal 2022] B. Singhal and A. Aggarwal: Etl, elt and reverse etl: A business case study. in Second International Conference on Advanced Technologies in Intelligent Control, Environment, Computing and Communication Engineering (ICATIECE), pp. 1–4, 2022.
- [A. Silberschatz 2011] A. Silberschatz, H. F. Korth, and S. Sudarshan: Database system concepts. 2011.

V

- [H. Varshney 2023] H. Varshney: What is a Data Staging Area? Staging Data Simplified 101. https://hevodata.com/learn/data-staging-area/, 2023.