

The background features a light blue gradient with various geometric elements. In the top left, there are white and light blue downward-pointing chevrons and a series of parallel diagonal lines. A vertical column of five squares, alternating in color between light yellow and light blue, is positioned in the upper left. The bottom left corner contains a bar chart with five bars of varying heights, each composed of stacked segments in shades of purple and blue. A small white square is located above the second bar from the left. The title 'Data Warehouses' is centered on the right side of the image in a bold, dark blue font.

Data Warehouses

Agenda

01

Einführung

03

Use Case / Übung

02

Implementierung
eines DWH

04

Big Data und Data Lakes

Git-Repo für die Übung

<https://inf-git.fh-rosenheim.de/sINFchzogl/DataWarehousing>

ODER:

<https://github.com/tinzog/DataWarehousing>



01

Einführung

: Data Warehouses



Definition Data Warehouse

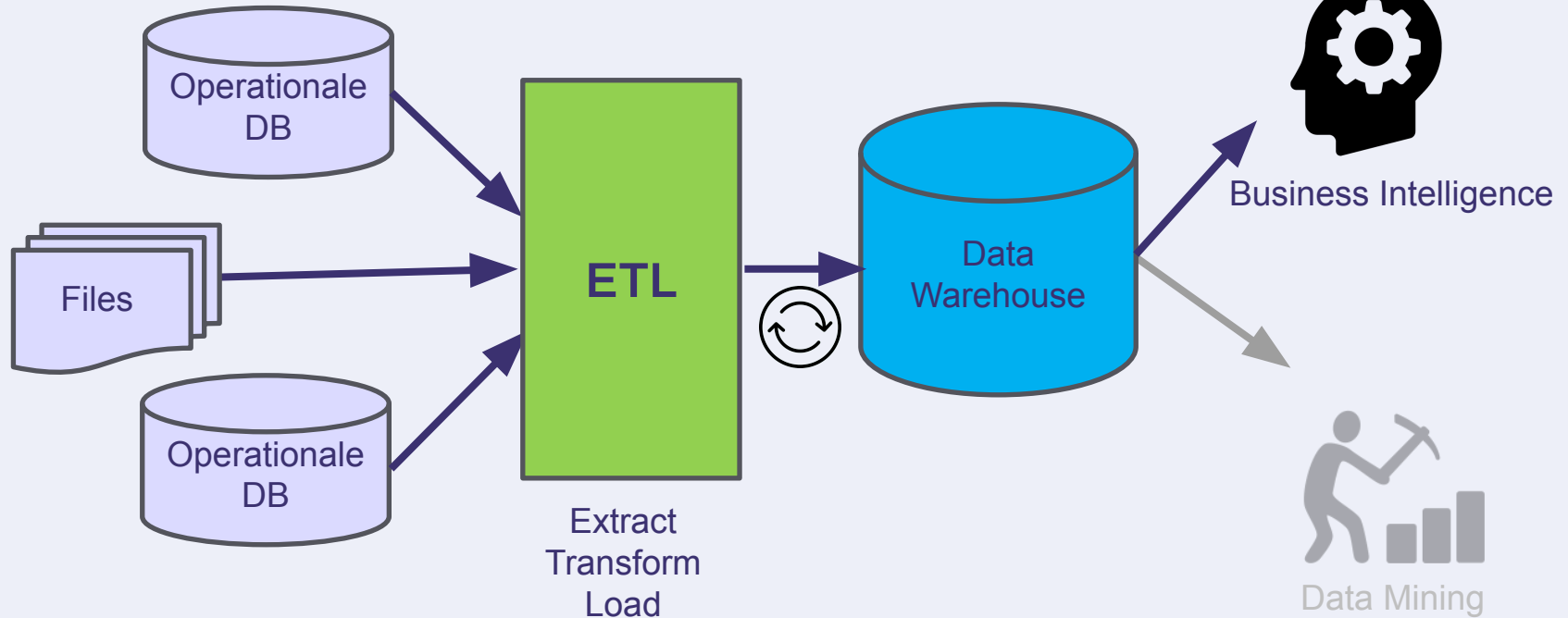
Eine für **Analysezwecke** optimierte zentrale Datenbank, die Daten aus mehreren, i.a. **heterogenen Quellen** zusammenführt und verdichtet.

OLTP

Online transactional processing

OLAP

Online analytical processing



Warum separates Data Warehouse?

1. **Unterschiedliche Nutzung und Datenstruktur:** Fokus auf Analyse
2. **Performance:** OLTP optimiert für ACID, OLAP würde Systeme ausbremsen
3. **Funktionalität:** Historische Daten verfügbar, heterogene Datenquellen zusammengeführt
4. **Sicherheit:** Operative Daten unangetastet, Zugriffssteuerung

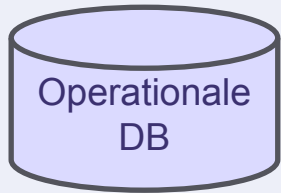
Nachteile:

1. **Datenredundanz**
2. **Hoher Einrichtungs- und Administrationsaufwand**
3. **Daten nicht zeitaktuell**
4. **Hohe Kosten**



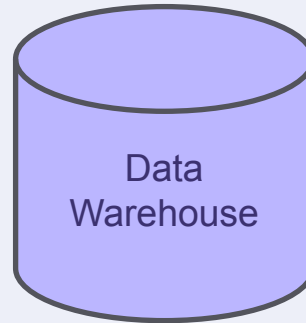
DWH Eigenschaften nach Inmon (1)

A Data Warehouse is a **subject-oriented, integrated, non-volatile**, and **time variant** collection of data in support of **management decisions** (W. H. Inmon, Building the Data Warehouse, 1996)



application-oriented

z.B.
Personalverwaltung,
Lagerverwaltung,
Buchhaltung



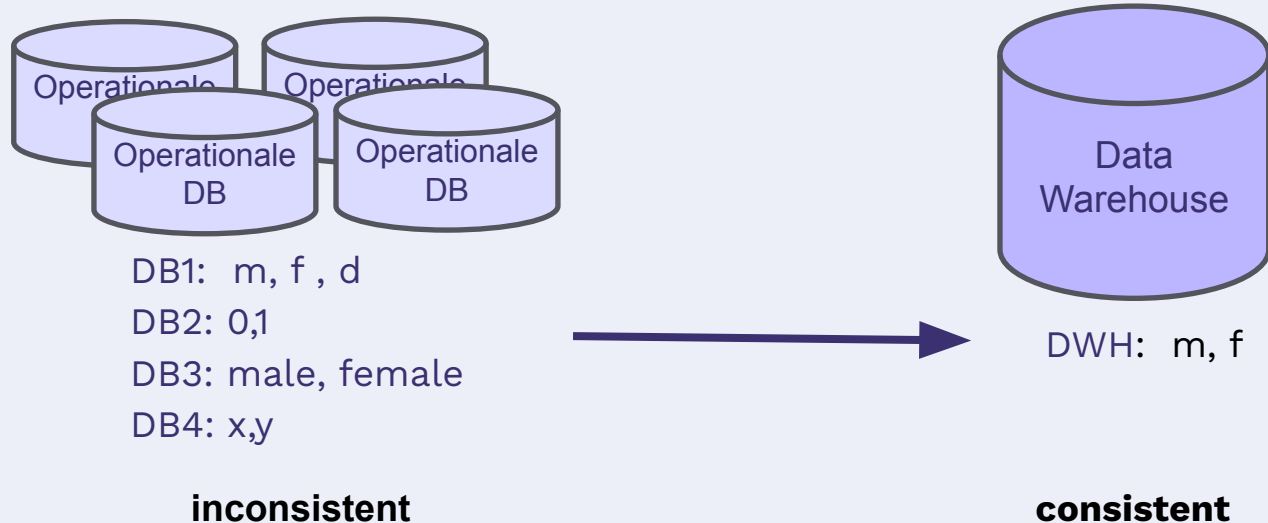
subject-oriented

z.B.
Kunde,
Produkt,
Filiale

DWH Eigenschaften nach Inmon (2)

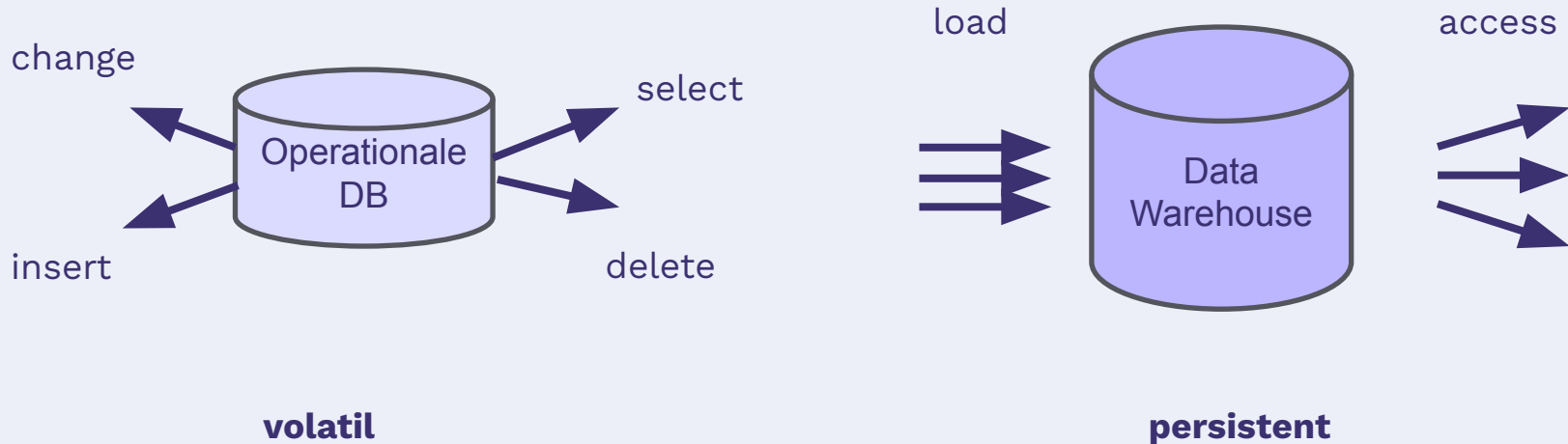
Integrierte Datenbasis (integrated):

Daten aus mehreren verschiedenen Datenquellen werden integriert



DWH Eigenschaften nach Inmon (3)

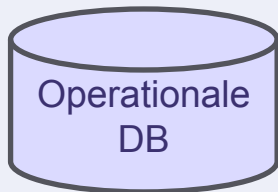
Nicht-flüchtige Datenbasis (non-volatile): Daten im DWH werden i.d.R. nicht mehr geändert: stabile persistente Datenbasis



DWH Eigenschaften nach Inmon (4)

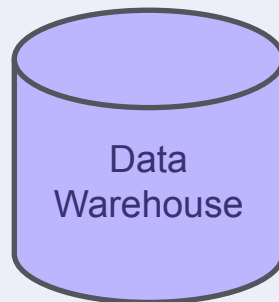
Historische Daten (time-variant):

Zeitreihenanalysen möglich, Speicherung über längeren Zeitraum



Aktuelle Daten

- Zeitbezug optional
- Zeithorizont: 60-90 Tage
- Daten änderbar



Schnappschuss

- Zeitbezug aller Objekte
- Zeithorizont: 2 – 10 Jahre
- keine Änderung nach Schnappschuss

Vergleich OLTP / OLAP

Apekt	Operationale Datenbanken OLTP	Data Warehouses OLAP
Entstehung	Aus Applikation	Controlling-Anforderungen
Bedeutung	Tagesgeschäft	Business Intelligence
Datenzugriff	Häufiger Zugriff, CRUD	Selten, zur Berichtserstellung
Nutzer	Sachbearbeiter, Online-Nutzer	Management / Data Scientist
Änderung / Aktualität	Häufig / stets aktuell	Zeitpunkt des letzten Snapshots
Datenquellen	Meist eine	Oft mehrere heterogene Quellen
Datenmerkmale	Nicht abgeleitet, autonom, zeitaktuell, dynamisch,	Konsistent, abgeleitet, nicht aktuell, statisch
Optimierungsziele	Hoher Durchsatz /Verfügbarkeit, kurze Antwortzeit	Optimiert für Abfragen

Bsp. Warenhauskette

Verfügbare Daten



- Lagerbestände in den Filialen
- Verkaufszahlen in den Filialen
- Warenkorbdaten der Kunden
- Zeitpunkte und Inhalte von Marketingkampagnen

Fragestellungen und Ziele



- Minimierung der Bestände
- Optimierung der Produktpalette
- Optimierung der Preisgestaltung (Gewinnerhöhung)
- Ermittlung von Kassenschlagern und Ladenhütern
- Erfolgskontrolle von Marketingkampagnen
-

Bsp Versicherungen

Verfügbare Daten



1. **Persönliche Informationen:** Alter, Geschlecht, Beruf, Gesundheitsinformationen, Risikofaktoren...
2. **Police-Daten:** Typ, Deckung, Prämien, Bedingungen, Konditionen,..
3. **Schadensdaten:** Datum, Typ, beantragter Betrag, Abwicklungsdaten,..
4. **Finanzdaten:** Zahlungsinformationen, Buchhaltungsdaten, Investitionen,..
5. **Interaktionsdaten:** Aufzeichnungen über Interaktionen mit Kunden über verschiedene Kanäle wie Callcenter, E-Mails und Online-Portale.
6. **Externe Daten:** Marktdaten, Wetterinformationen,..

Bsp Versicherungen



Fragestellungen und Ziele

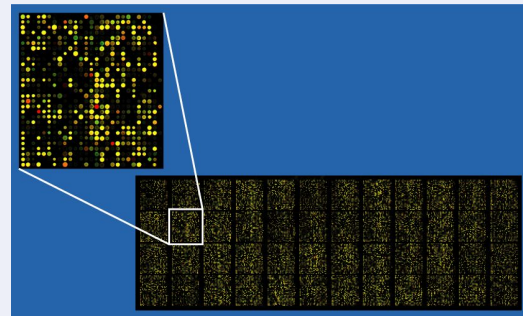
1. **Risikobewertung:** von Gruppen / Personen
2. **Muster bei Ansprüchen:** Betrugsmuster / Hochrisikomuster
3. **Kundensegmentierung:** Personalisiertes Marketing / Produktangebote
4. **Preisstrategien:** für das Produktportfolio
5. **Operationelle Effizienz:** Kosten senken, Service verbessern
6. **Regulatorische Compliance:** Datensicherheit, Revisionssicherheit
7. **Markttrends:** aktuelle Trends bewerten, Firma anpassen

Bsp Bioinformatik: Microarrays



Verfügbare Daten

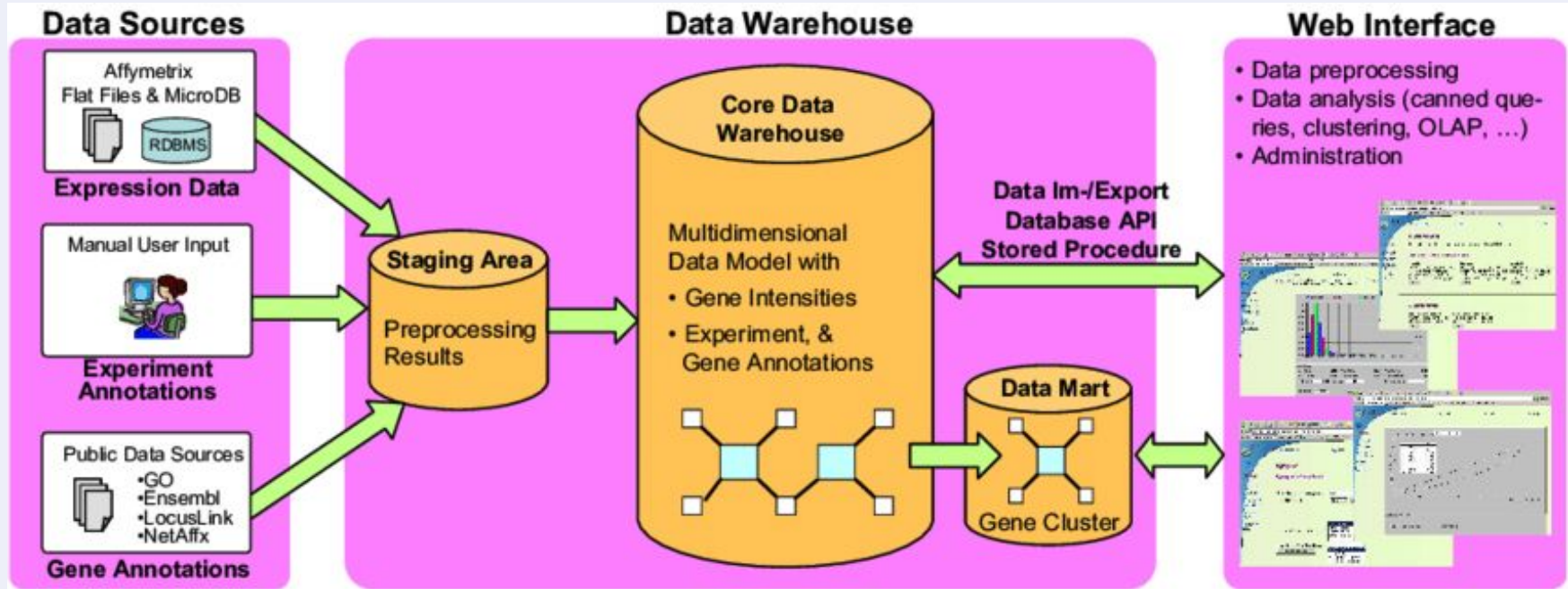
- Expressionsmuster (Microarrays)
- Information zu Proben/Patienten
- Experimentelle / klinische Daten
- Aktuelle Protein-/Gendatenbanken



Fragestellungen und Ziele

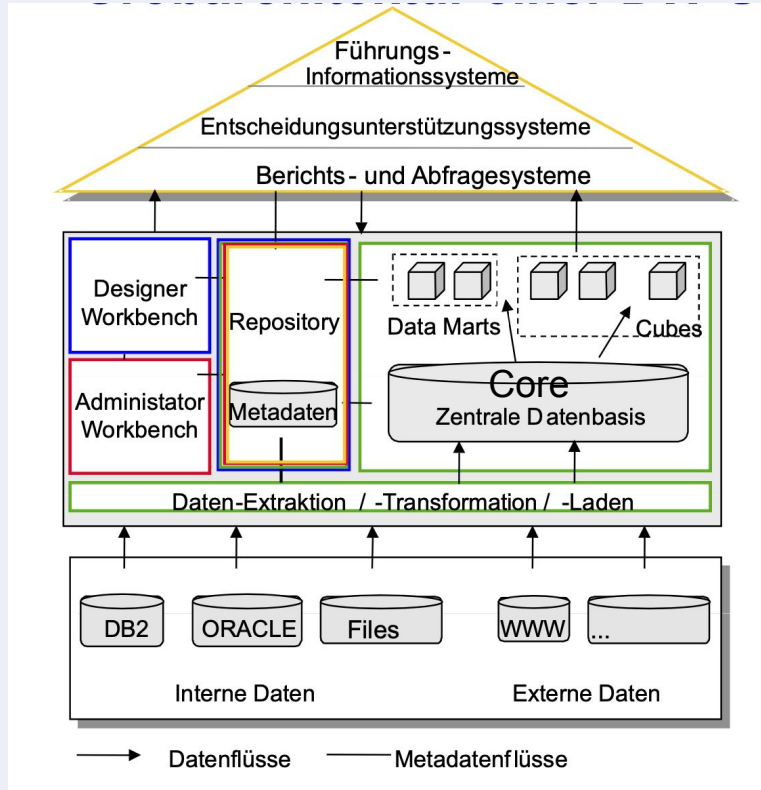
- Welche Gene werden bei Patienten / Gesunden exprimiert? In welchen Geweben?
- Analyse von Stoffwechselketten
- Gibt es Expressionsgruppen?
-

GeWare: Expression Data Warehouse



Kirsten, Toralf & Rahm, Erhard. (2007). Technical Report A Data Warehouse for Multidimensional Gene Expression Analysis.

Grobarchitektur DWH



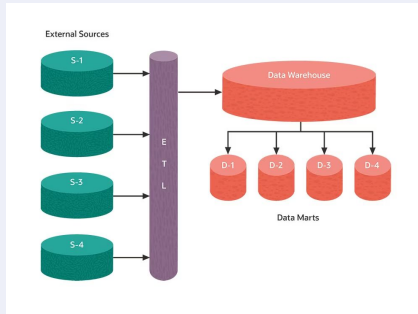
Business Intelligence

Data Warehouse

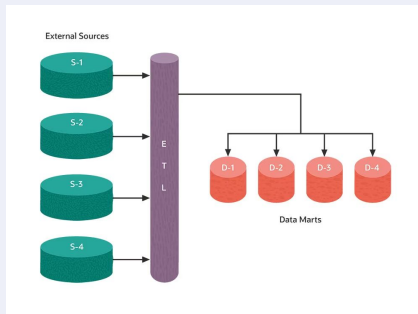
Operative Datenquellen

Data Marts

Dependent



Independent



- Subjekt-/Themenorientiert
- z.B. Unternehmensbereich
- Beantwortet bestimmte Fragestellungen

PRO

- Performance / Effizienz
- “Single Source of truth”
- Zugriffskontrolle
- Flexibilität

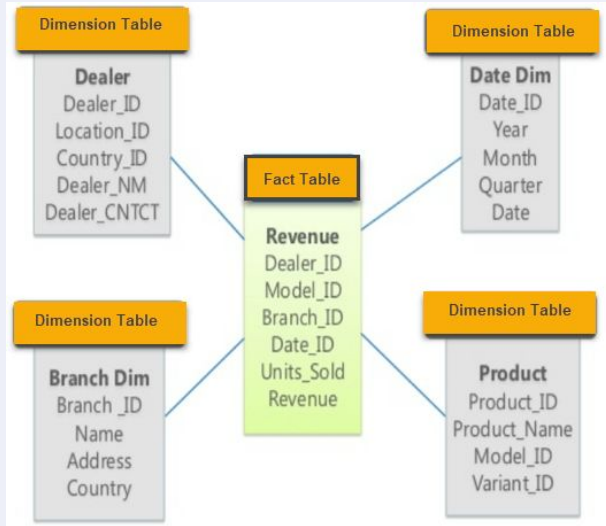
CON

- Redundanz
- Abweichungen zwischen Marts

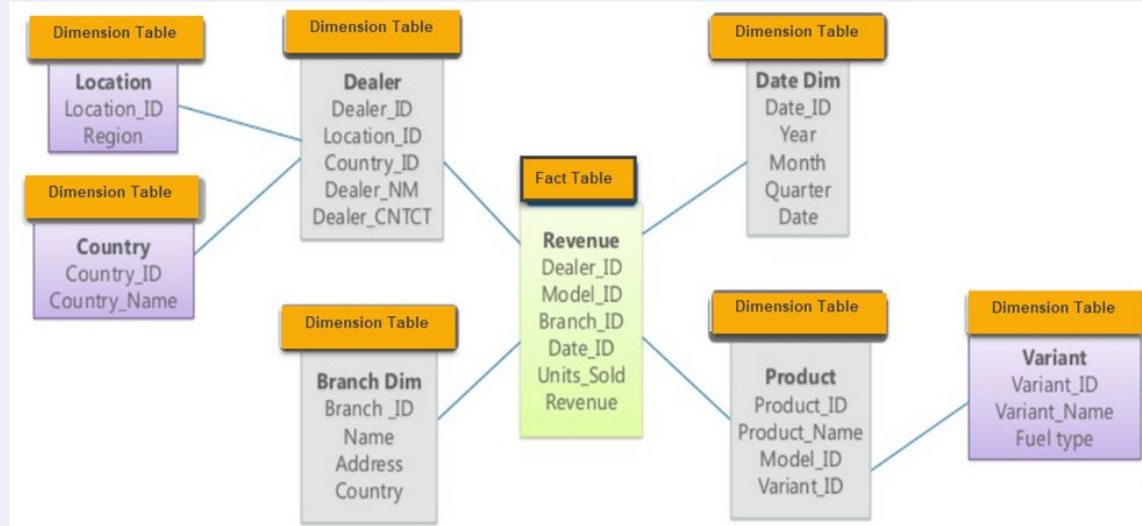
Wichtigste Core Architekturen

- **Dimensional Core Model** (Kimball)
- **Relational Core Model** (Inmon)
- **Data Vault Model**

Dimensional



Star



Snowflake

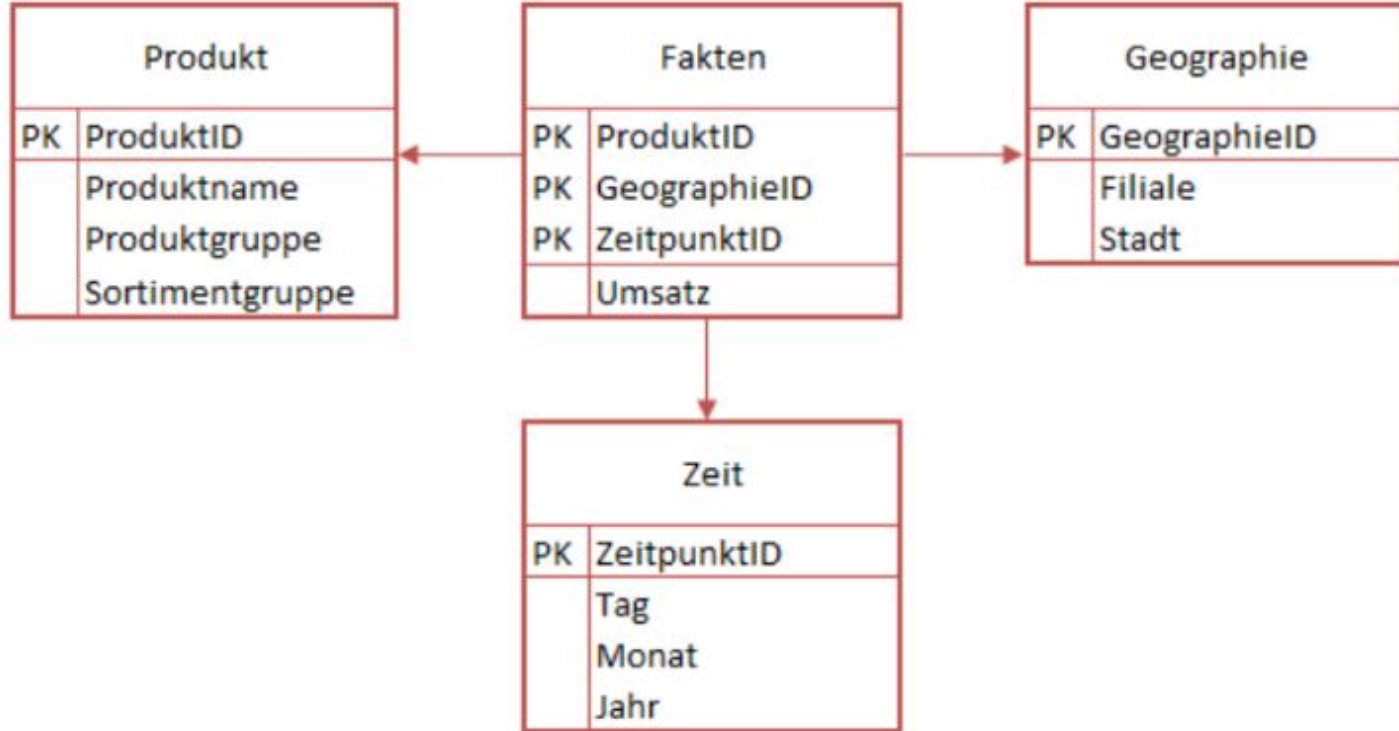
Übung

Eine Elektronikmarktkette speichert einzelne Transaktionen, in denen jeweils nur ein Gegenstand erworben wurde. In der Tabelle finden Sie einen Auszug aus diesen Transaktionen.

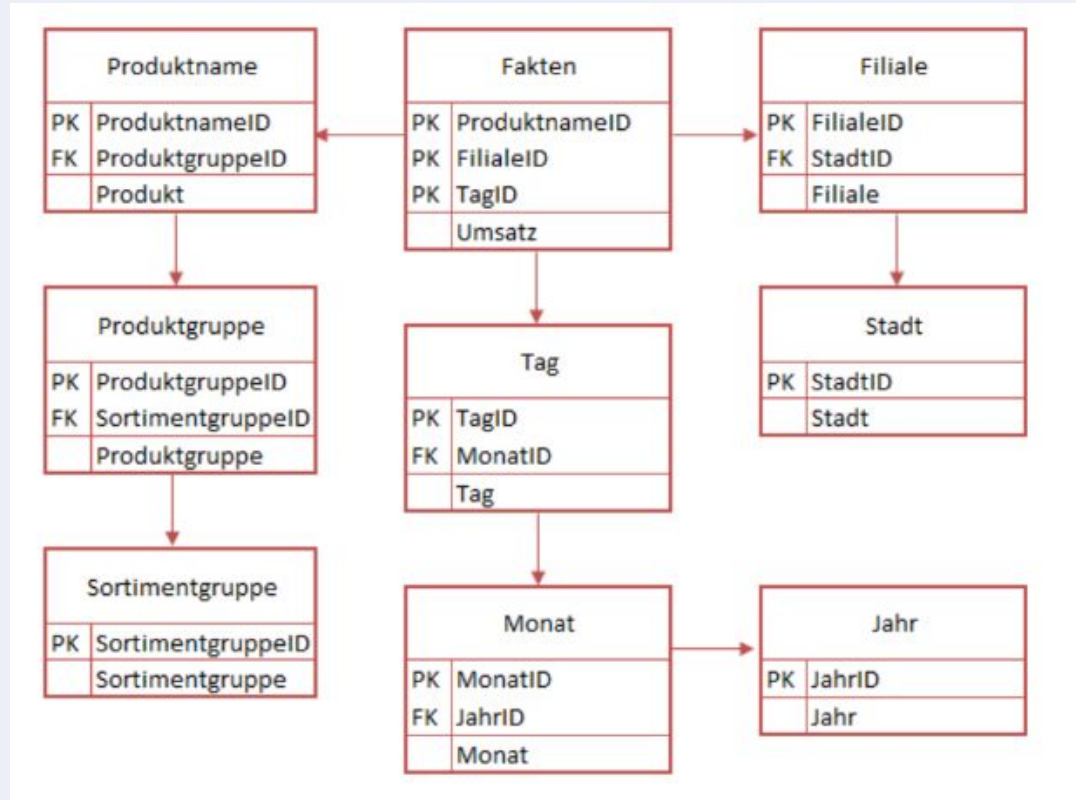
Datum	Produkt	Produktgruppe	Sortiment- gruppe	Filiale	Stadt	Umsatz (in €)
10.06.2013	„Ohne Limit“	DVD	Medien	Hauptstr.	Bielefeld	5,99
12.06.2013	Sony 5.1 Heimkino	Multimediasysteme	Multimedia	Südring	Paderborn	459,00
12.06.2013	Asus EEEPC	Notebook	Computer	Bahnhofstr.	Bielefeld	6,99
14.06.2013	Hama HDMI-Kabel	Multimedia Zubehör	Multimedia	Poststr.	Dortmund	11,59
17.06.2013	„Drive“	DVD	Medien	Südring	Paderborn	12,99
17.06.2013	Philipps Bügeleisen	Reinigung	Haushalt	Hauptstr.	Bielefeld	32,49
20.06.2013	Apple MacbookPro	Notebook	Computer	Schillerstr.	Dortmund	1099
21.06.2013	AEG Spülmaschine	Küche	Haushalt	Bahnhofsstr.	Bielefeld	366,79
23.06.2013	Lady Gaga	CD	Medien	Poststr.	Dortmund	4,99
26.06.2013	HP Drucker	Peripherie	Computer	Südring	Paderborn	65,29

Erstellen Sie auf Basis der Tabelle ein Star Schema
Normalisieren Sie das Star Schema zu einem Snowflake Schema.

Lösung 1

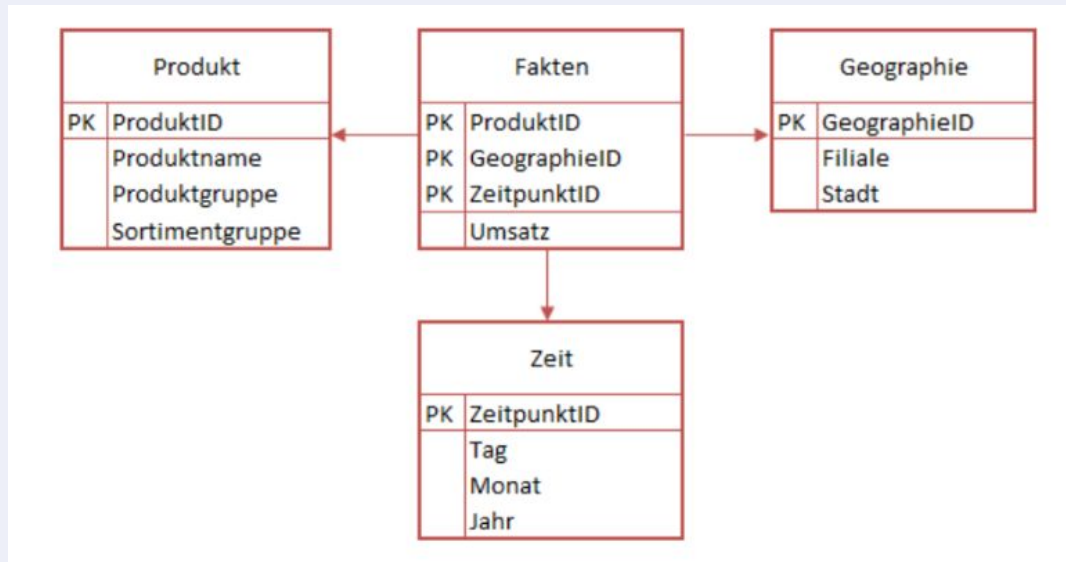


Lösung 2

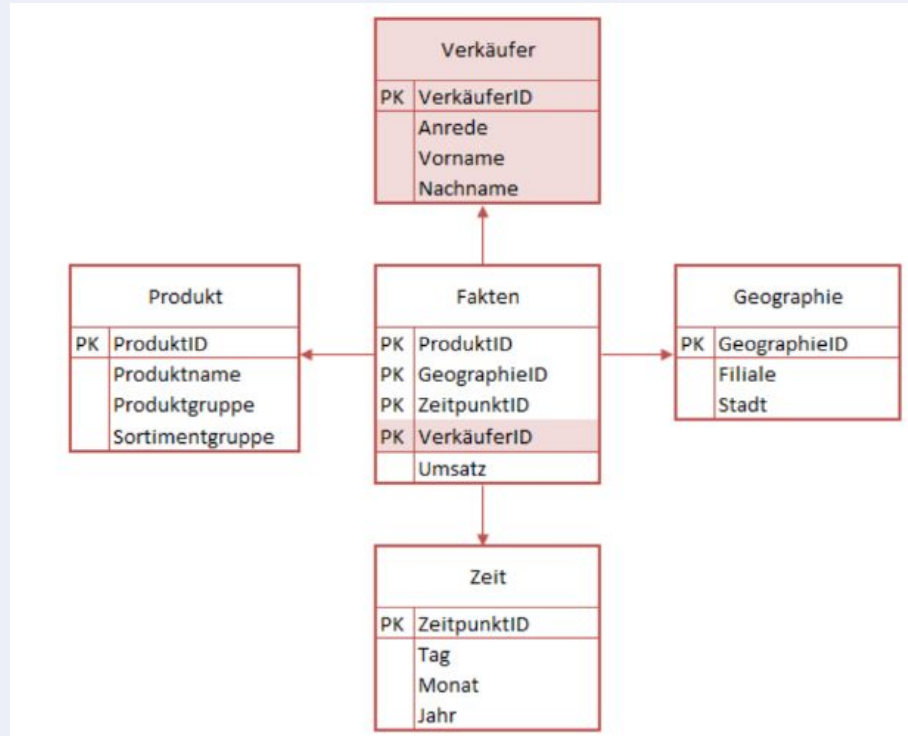


Übung-Erweiterung

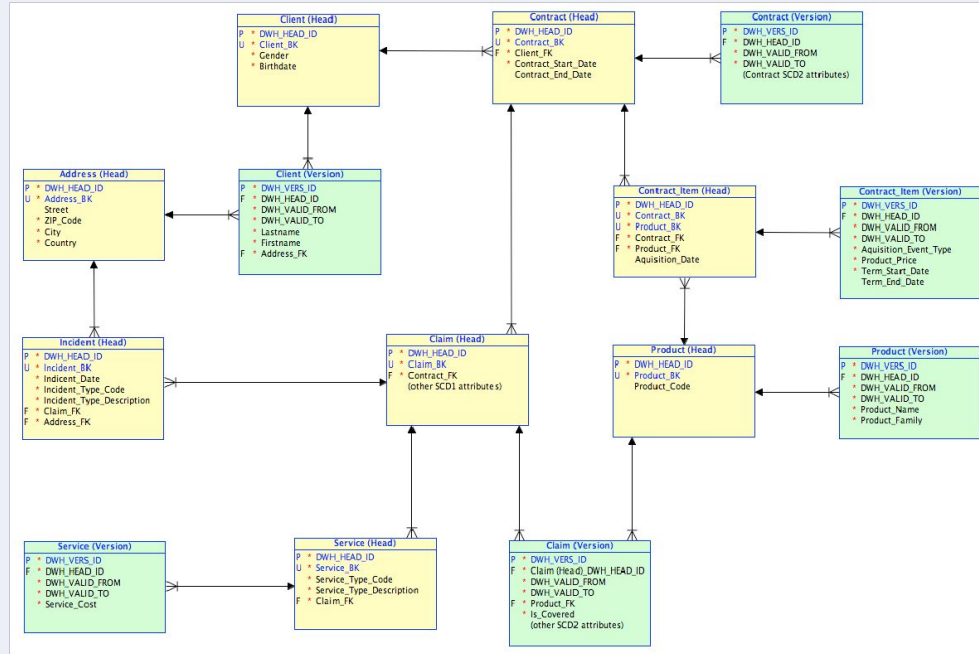
In der Transaktion soll nun auch der Verkäufer mit Anrede, Vornamen und Namen gespeichert werden. Geben Sie an, welche Änderungen jeweils pro Schema nötig sind



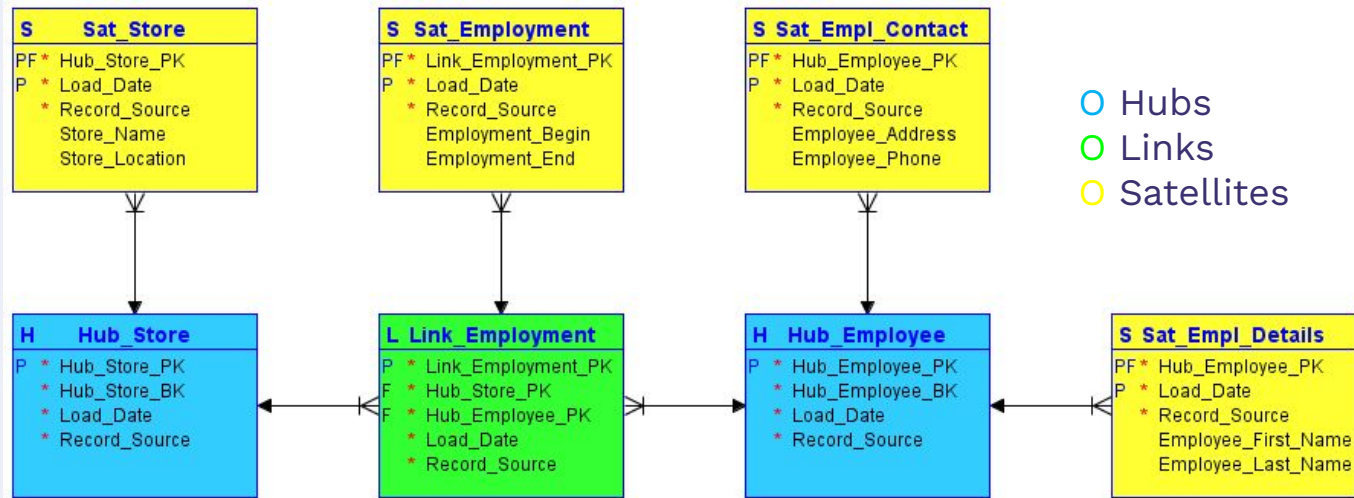
Lösung



Relational



Data Vault



Architekturentscheidung

Dimensionales Modell:

- Für Data Marts/Core-Modelle mit klaren analytischen Bedürfnissen
- Benutzerfreundlich für Geschäftsanwender, einfach, performant
- Aber: Anpassungen bei neuen Analysebedürfnissen notwendig

Relationales Modell:

- Für Core-Schichten mit unklaren/entwickelnden Anforderungen
- Flexibel, vereinfacht Core-ETL
- Aber: Komplexe Datenlieferung an Data Marts

Data Vault Modell:

- Für Umgebungen mit vielen Quellsystemen/strukturellen Änderungen, inkrementeller Entwicklung
- Erleichtert das Hinzufügen neuer Daten
- Aber: Komplexes Modell, schwierige Datenlieferung an Data Marts

Quiz

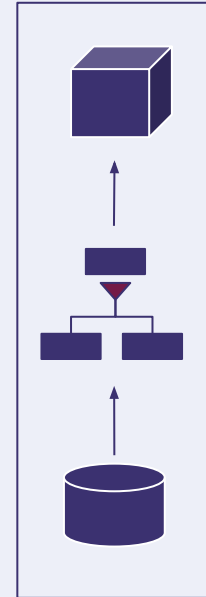
https://docs.google.com/forms/d/e/1FAIpQLSftE5iNt0GaLOcNEYIpBbGipj_scXW0ScTlCLyc66uv0JCxw/viewform?usp=sf_link



02

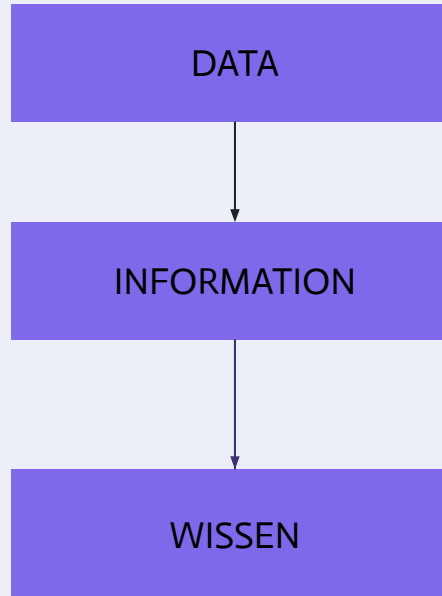
Implementierung eines Data Warehouse

Was ist...

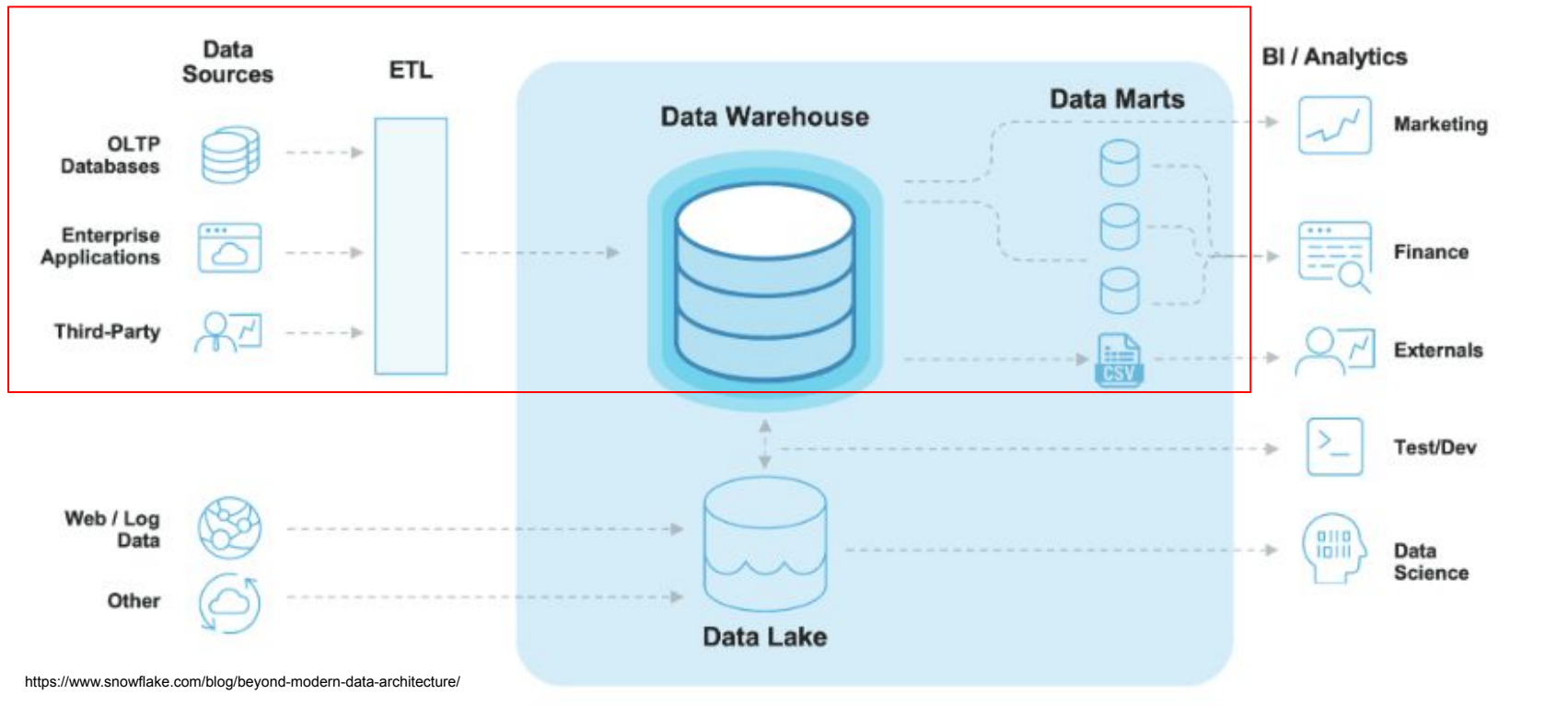


Arten von Data DWH-Anwendungen

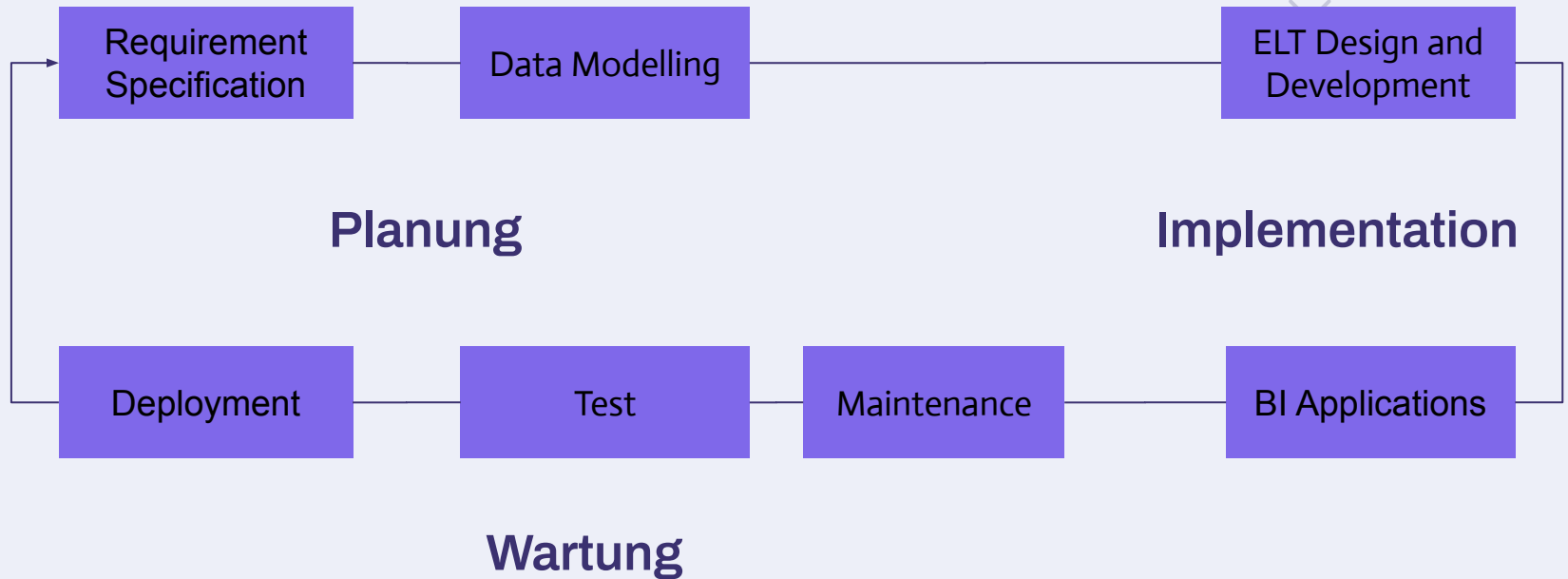
- Informationsverarbeitung
 - Abfrage- Tools
 - Berichterstellung-Tools
 - Statistik-Tools
- Analytische Verarbeitung
 - Drilldown
 - Slice-Dice
 - Pivot
- Data Mining
 - Muster Erkennung
 - Beziehungen
 - Einstufung
 - Prognose



DWH in moderne Datenpipeline



Lebenszyklus der DWH Entwicklung



Persona-Matrix

Schritt	Analysts	Data Scientist	Data Engineers	Database Administrator
Requirement Specification	v	v		
Data Modelling	v	v	v	v
ELT Design and Development			v	v
BI Applications	v	v		
Maintenance			v	
Test and Deploy	v	v	v	

Requirement Specification

- Unterstützung individueller Sichten (z.B. bzgl. Zeithorizont, Struktur)
- Erweiterbarkeit (z.B. Integration neuer Quelle)
- Automatisierung der Abläufe
- Eindeutigkeit über Datenstrukturen
- Datenqualität
- Zugriffsberechtigungen und Prozesse
- Ausrichtung am Zweck: Analyse der Daten

Datenmodellierung

Schritt 1 – Wählen Sie den Organisationsprozess aus. Stellen Sie Fragen zu einem Prozess

Schritt 2 – Deklarieren Sie die Granularität (Wie hoch ist die Detailgenauigkeit)

Schritt 3 – Identifizieren Sie die Dimensionen und Hierarchien

Schritt 4 – Identifizieren Sie die Fakten.

Datenmodellierung

Kennzahlen/Fakten

Beschreiben betriebswirtschaftliche Sachverhalte

- Additiv (Berechnung zwischen sämtlichen Dimensionen möglich)
- Semi Additive (Berechnung möglich mit Ausnahme temporaler Dimension)
- Nicht-Additiv (keine additive Berechnung möglich)

Dimensionen

Beschreibt mögliche Sicht auf die assoziierte Kennzahl

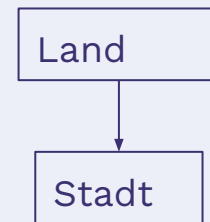
- Conformed dimension
- Junk dimension
- Degenerated dimension
- Role-playing dimension
- Role-playing dimension
- Static dimension
- Shrunk dimension
- Slowly changing dimensions

Datenmodellierung

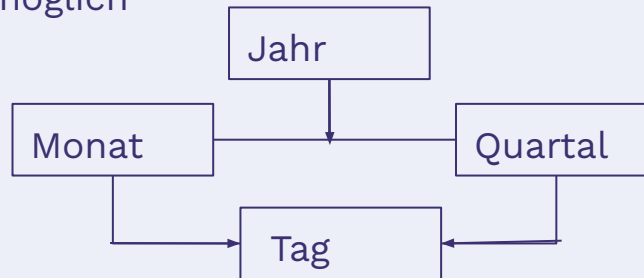
Hierarchien in Dimensionen

- Dimensionselemente sind Knoten einer Klassifikationshierarchie
- Klassifikationsstufe beschreibt Verdichtungsgrad

Einfache Hierarchien: höhere Ebene enthält die aggregierten Werte einer niedrigeren Hierarchiestufe



Parallele Hierarchien: innerhalb einer Dimension sind mehrere verschiedene Arten der Gruppierung möglich



Slowly changing dimensions (SCD)

Die SCD im Data Warehouse ist ein wichtiges Konzept, das verwendet wird, um den historischen Aspekt von Daten in einem Analysesystem zu ermöglichen.

Type 1 : Ignoriert alle Änderungen und prüft die Änderungen.

Type 2 :Überschreibt die Änderungen

Type 3 :Der Verlauf wird als neue Zeile hinzugefügt.

Type 4 :Der Verlauf wird als neue Spalte hinzugefügt.

Type 5 :Eine neue Dimension wird hinzugefügt

Slowly changing dimensions

CustomerKey	CustomerCode	MaritalStatus	Gender	Designation	AnnualIncome
11011	AW00011011	M	M	Professional	760000
11012	AW00011012	M	F	Management	500000
11015	AW00011015	S	F	Skilled Manual	NULL
11018	AW00011018	S	M	Clerical	650000
11019	AW00011019	S	M	Skilled Manual	NULL

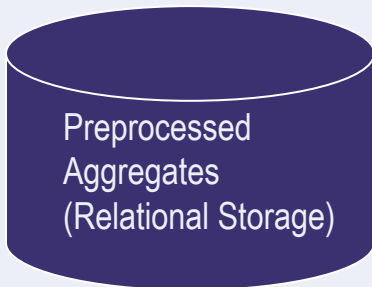
CustomerKey	MaritalStatus	Gender	Designation	FirstDesignation	JoinedDate	DateFirstPurchase
11011	M	M	Professional	Junior Executive	2010-12-28	2010-12-30
11012	M	F	Management	Professional	2013-03-01	2013-03-16
11015	S	F	Skilled Manual	Skilled Manual	2013-01-16	2013-01-18
11018	S	M	Clerical	Clerical	2011-11-15	2011-01-17
11019	S	M	Skilled Manual	UnSkilled Manual	2013-02-12	2013-02-12

CustomerKey	CustomerCode	MaritalStatus	Gender	Designation	StartDate	EndDate	IsCurrent
11011	AW00011011	M	M	Professional	2012-01-01	9999-12-31	Yes
11012	AW00011012	M	F	Management	2012-01-01	9999-12-31	Yes

Umsetzung des multidimensionalen Modells

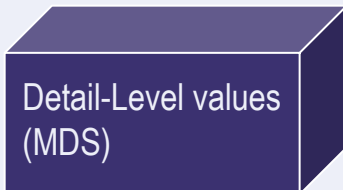
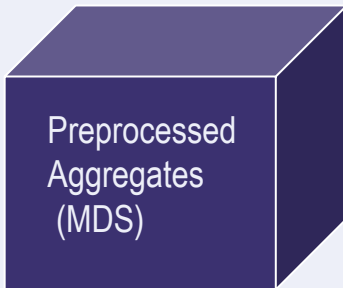
ROLAP

Cube Structure (MDS)



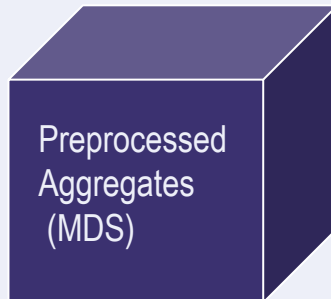
MOLAP

Cube Structure (MDS)



HOLAP

Cube Structure (MDS)



Übung :relationale Repräsentation mehrdimensionaler Daten (ROLAP)

Use Case: DEMO Company

DEMO Company ist ein fiktives **Telekommunikationsunternehmen**, das Bayern, Sachsen und NRW bedient. Derzeit beliefert das Unternehmen 14,5 Millionen Privat-, Gewerbe- und Industriekunden in diesen Gebieten, darunter auch die Stadt München. Drei **Callcenter** kümmern sich um alle Kundeninteraktionen mit dem Unternehmen, einschließlich Rechnungsanfragen, Zahlungen auf elektronischem Weg, Terminvereinbarung, Erteilung von Notfallaufträgen, Ausfallmeldungen usw.

Kunden haben Interesse an einer stärker automatisierten Abwicklung vieler dieser Transaktionen bekundet, z.b. als **Self-Service-Website**, ohne den Umweg über Callcenter.

In der Vergangenheit hat das Unternehmen neuere Technologien nur langsam eingeführt. Aufgrund des Kundenfeedbacks und des Wunsches, neue Möglichkeiten zu erkunden, die Kosteneinsparungen ermöglichen würden, hat sich die Geschäftsleitung jedoch für die Implementierung eines Kunden-Self-Service-Portals mit eigener Datenbank entschieden. Dies wird mit der Datenbank synchronisiert, die zum vorhandenen Kundenbetreuungssystem gehört.

Use Case: DEMO Company

Geschäftsprobleme

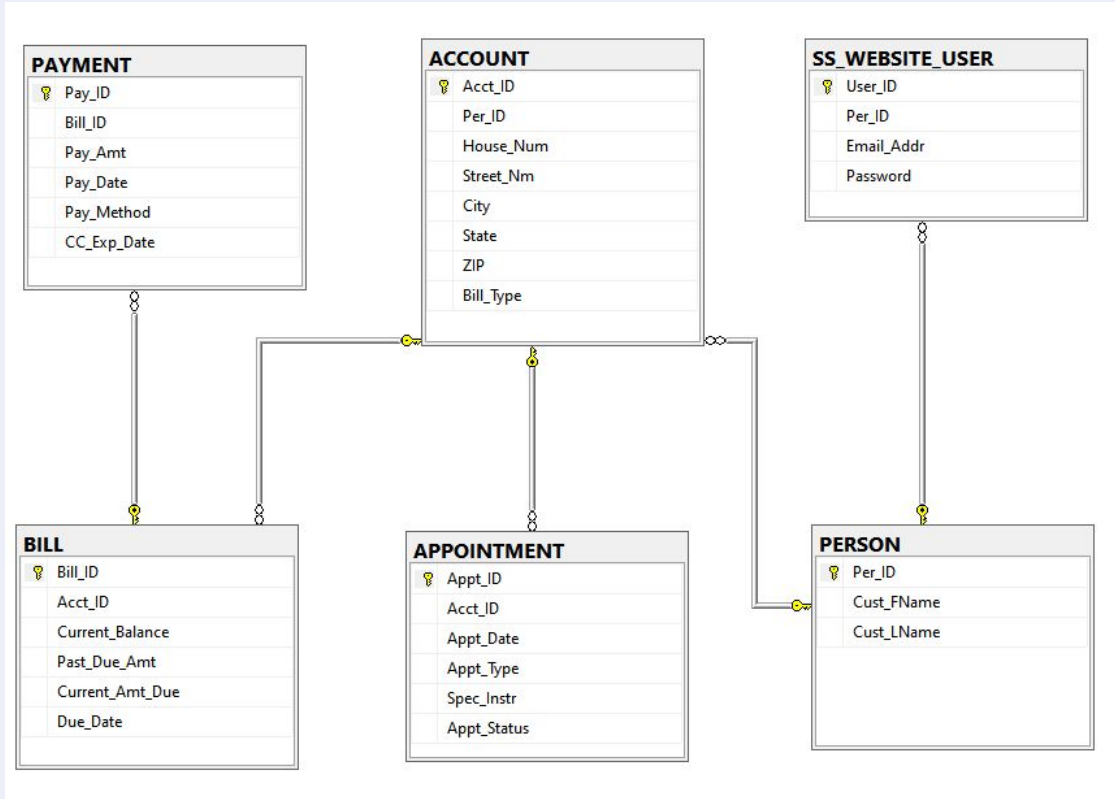
1. Die **Call-Center-Abteilung** hat in den letzten drei Jahren aufgrund der gestiegenen Anzahl von Anrufen und des ständigen Bedarfs an Überstunden und/oder zusätzlichen Mitarbeitern zur Bewältigung des Volumens ständig das Budget überschritten.
2. Der Service Level, ein Unternehmens-KPI, lag aufgrund der Zeit, die für die Bearbeitung bestimmter Anrufe benötigt wird, unter dem Zielwert, beispielsweise bei Zahlungsaufforderungen, da auch ein Dritter kontaktiert werden muss, um die Zahlung abzuwickeln.
3. Die **Rechnungsabteilung** steht aufgrund der Porto- und Papier-/Umschlagkosten für den Versand von Rechnungen an 14,5 Millionen Kunden unter erheblichem Budgetdruck.
4. Die Überweisungsabteilung wurde damit beauftragt, nach Möglichkeiten zu suchen, die Kosten für die Zahlungsabwicklung zu senken, und Untersuchungen zeigen, dass elektronische Zahlungen für das Unternehmen kostengünstiger wären als Zahlungen, die auf herkömmliche Weise per „Papierscheck“ erfolgen.
5. Die Kundenzufriedenheit mit dem Unternehmen ist in den letzten Jahren gesunken, da die Kontenverwaltung nicht online möglich ist.

Anforderungen

Data Warehouse

1. Das System muss die während eines bestimmten Kalenderzeitraums eingegangenen Zahlungen nach Methode ermitteln.
2. Das System muss die Rechnungsbeträge ermitteln, die während eines bestimmten Kalenderzeitraums fällig sind.
3. Das System muss die von einem Kunden erhaltenen Zahlungen ermitteln.
4. Das System muss die Rechnungsbeträge ermitteln, die ein Kunde geschuldet hat.
5. Das System muss überfällige Rechnungsbeträge nach Kalenderzeitraum und Kunde ermitteln.

DEMO Company: Datenmodell OLTP



	Acct_ID	Per_ID	House_Num	Street_Nm	City	State	ZIP	Bill_Type
▶	0123456789	0111111111	123	First St	Chicago	IL	58697	paper
	1122334455	5666666666	6543	Lake Ave	Joliet	IL	60939	paper
	1234567890	0111111111	456	Second St	Chicago	IL	58697	paper
	2233445566	6777777777	5432	State St	Rockford	IL	68488	paper
	2345678901	0111111111	789	Third St	Chicago	IL	58697	paper

	Appt_ID	Acct_ID	Appt_Date	Appt_Type	Spec_Instr	Appt_Status
▶	4321098765	6677889900	2015-10-28	Turn Off	Fence	Cancelled
	5432109876	2233445566	2014-12-21	Meter Read	Call Ahead	Pending
	6543210987	9012345678	2014-09-10	Turn On	NULL	Completed
	7654321098	4567890123	2014-10-15	Meter Read	Meter in Back	Completed

	Bill_ID	Acct_ID	Current_Balance	Past_Due_Amt	Current_Amt_...	Due_Date
▶	111111111111	0123456789	100,00	50,00	50,00	2014-10-12
	111111222222	2233445566	85,00	0,00	85,00	2014-10-25
	222222222222	0123456789	50,00	0,00	50,00	2014-09-12
	222222333333	6677889900	75,00	0,00	75,00	2014-11-25

User_ID	Per_ID	Email_Addr	Password
Bmiller99	8999999999	bmiller@email.com	*****
Jwalker	5666666666	jwalker@email.net	*****
Kjones3	1222222222	kjones@email.net	*****

	Pay_ID	Bill_ID	Pay_Amt	Pay_Date	Pay_Method	CC_Exp_Date
▶	121212121212121	111111111111	100,00	2014-10-11	checking	NULL
	232323232323232	333333333333	87,00	2014-10-14	credit card	2016-05-31
	343434343434343	444444444444	62,00	2014-10-19	checking	NULL
	454545454545454	555555555555	80,00	2014-10-15	credit card	2015-08-31

	Per_ID	Cust_FName	Cust_LName
▶	0111111111	Joe	Smith
	1222222222	Kara	Jones
	2333333333	Jean	Williams

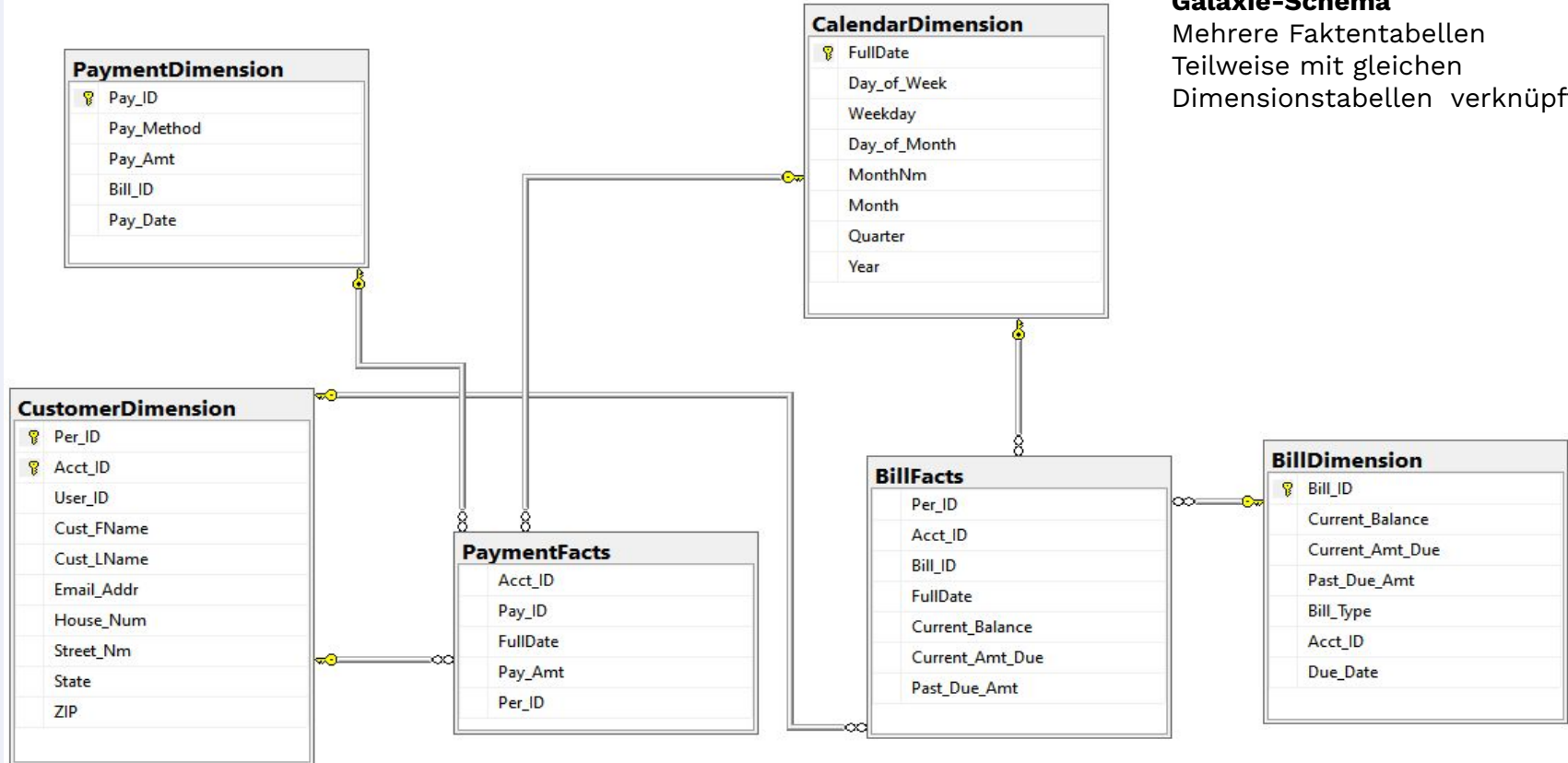
Entscheidung finden

- Analysieren und verstehen Sie Ihre Daten
- Analysieren Sie, wie oft Sie Daten laden müssen
- Definieren Sie eine Datenintegrations -Richtlinie
- Wählen Sie ELT Tools oder ETL
- Wählen Sie, ob Sie es vor Ort oder in der Cloud wünschen

Data Warehouse

1. Zahlungsbeträge sollten mit atomarer Granularität in einer Faktentabelle enthalten sein
2. Fällige Rechnungsbeträge, überfällige Beträge und Kontostände sollten in einer Faktentabelle mit atomarer Granularität enthalten sein.
3. Die CustomerDimension-Tabelle sollte die Personen-ID, die Konto-ID, die Benutzer-ID der Self-Service-Website, den Vornamen, den Nachnamen, die E-Mail-Adresse, die Hausnummer, den Straßennamen, die Stadt, das Bundesland und die Postleitzahl enthalten.
4. Die PaymentDimension sollte die Zahlungs-ID und die Zahlungsmethode enthalten.
5. Die BillDimension sollte die Bill ID und Bill Method enthalten.

DWH Model: DEMO Company



DWH Abfrage

--What's the sum of payments received in October, by type?

```
SELECT sum(PaymentFacts.Pay_Amt) as "Total Paid In October", PaymentDimension.Pay_Method as "Payment Method"
FROM dbo.PaymentFacts
inner join dbo.CalendarDimension ON PaymentFacts.FullDate = CalendarDimension.FullDate
inner join dbo.PaymentDimension ON PaymentFacts.Pay_ID = PaymentDimension.Pay_ID
WHERE CalendarDimension.MonthNm = 'October' GROUP BY PaymentDimension.Pay_Method
```

--What are the first and last names of customers that have past due balances sometime in October?

```
SELECT dbo.CustomerDimension.Cust_FName as "First Name", dbo.CustomerDimension.Cust_LName
FROM dbo.CustomerDimension
INNER JOIN dbo.BillFacts ON dbo.CustomerDimension.Per_ID = dbo.BillFacts.Per_ID
                        AND dbo.CustomerDimension.Acct_ID = dbo.BillFacts.Acct_id
INNER JOIN dbo.CalendarDimension ON dbo.BillFacts.FullDate = dbo.CalendarDimension.FullDate
WHERE dbo.CalendarDimension.MonthNm = 'October' AND dbo.BillFacts.Past_Due_Amt > '0.00'
```

0 %

Ergebnisse Meldungen

	Total Paid In October	Payment Method
1	312.00	checking
2	227.00	credit card

	First Name	Cust_LName
1	Joe	Smith
2	Jean	Williams

Load Dimension

```
-- Load the dimensions
```

```
insert into Ziel_DWH.dbo.CustomerDimension (Per_ID
```

```
, Acct_ID
```

```
, User_ID
```

```
, Cust_FName
```

```
, Cust_LName
```

```
, Email_Addr
```

```
, House_Num
```

```
, Street_Nm
```

```
, State
```

```
, ZIP )
```

```
SELECT PERSON.Per_ID
```

```
, ACCOUNT.Acct_ID
```

```
, SS_WEBSITE_USER.User_ID
```

```
, PERSON.Cust_FName
```

```
, PERSON.Cust_LName
```

```
, SS_WEBSITE_USER.Email_Addr
```

```
, ACCOUNT.House_Num
```

```
, ACCOUNT.Street_Nm
```

```
, ACCOUNT.State
```

```
, ACCOUNT.ZIP
```

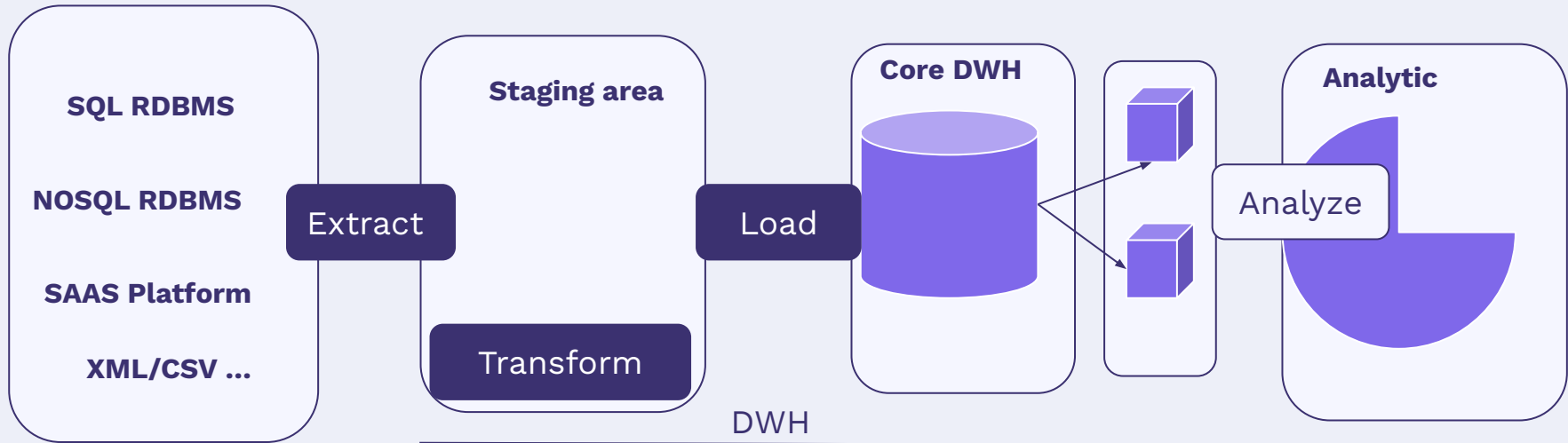
```
FROM Quelle_OLTP_System.dbo.PERSON
```

```
left join Quelle_OLTP_System.dbo.SS_WEBSITE_USER ON PERSON.Per_ID = SS_WEBSITE_USER.Per_ID
```

```
left join Quelle_OLTP_System.dbo.ACCOUNT ON PERSON.Per_ID = ACCOUNT.Per_ID
```

Extraction, Transformation, Loading (ETL)

- Schritt 1: Extrahiert Daten aus heterogenen Quellen.
- Schritt 2: Transformiert Daten über dedizierte Datenflüsse oder im Staging-Bereich.
- Schritt 3: Ladt bereinigte und transformierte Daten in DWH



Extraction, *Transformation*, Loading

Schritt 1

extracting

Schritt 2

consolidating

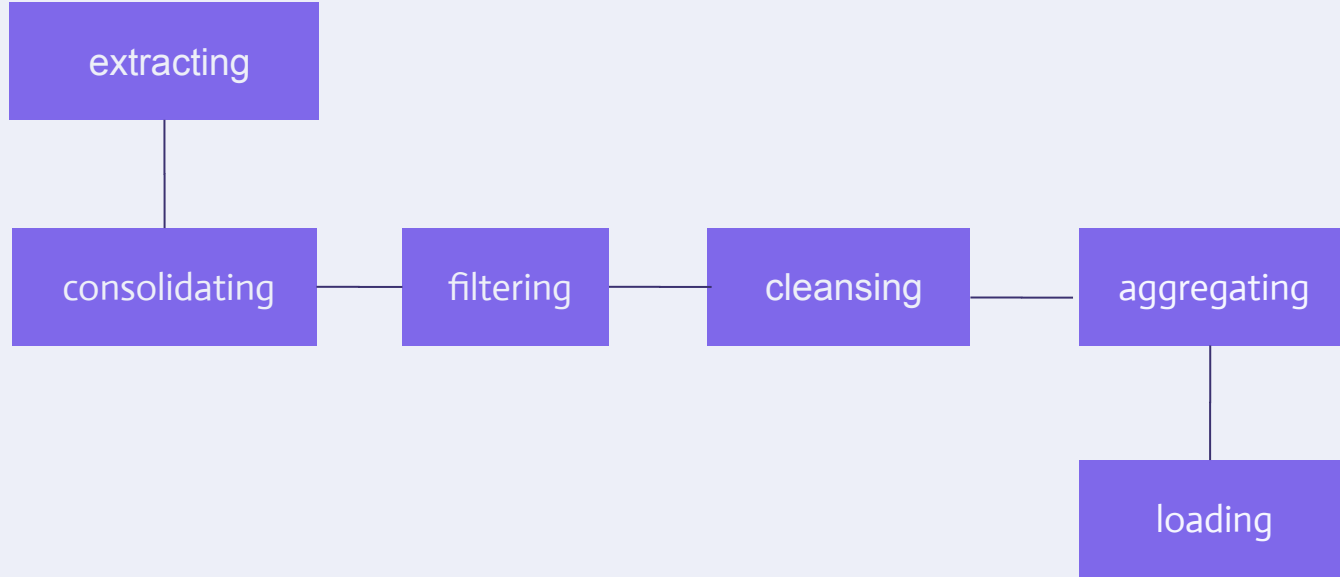
filtering

cleansing

aggregating

Schritt 3

loading



Datenqualität

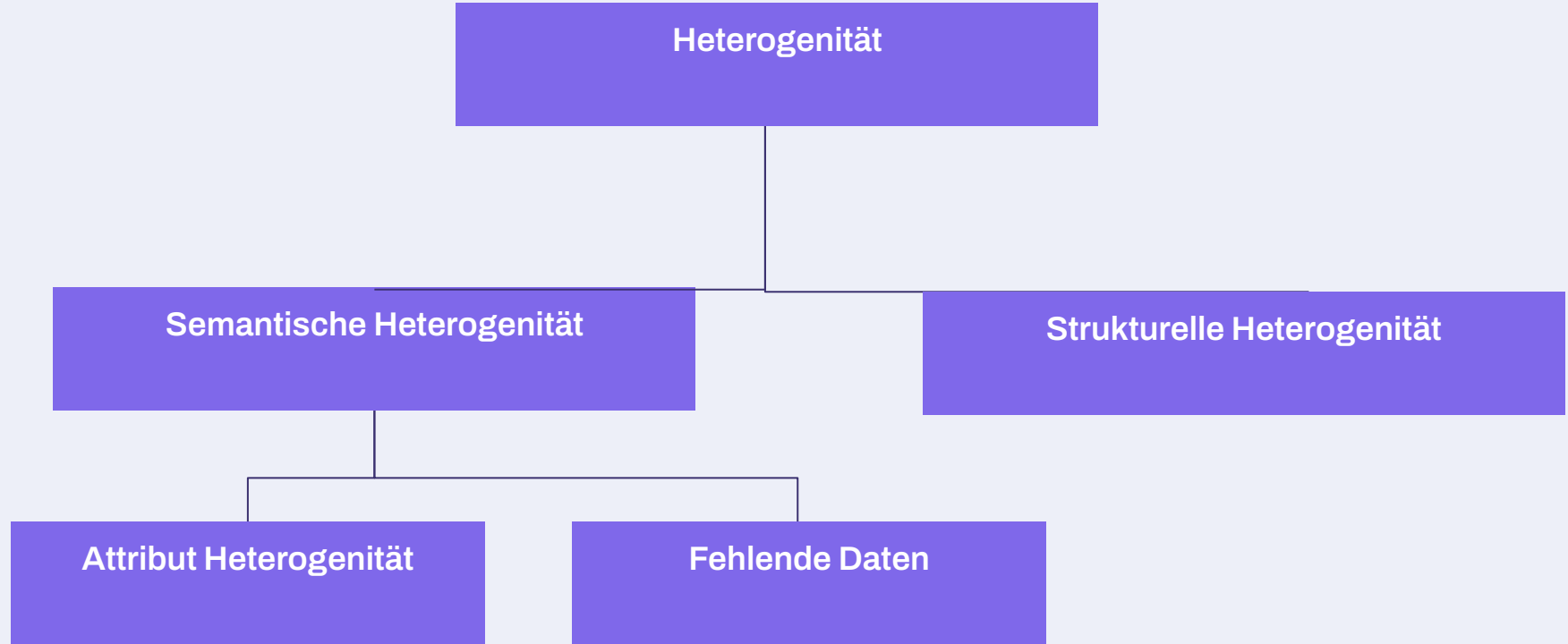
Single-Source-Probleme

- Fehlen von Schemata (z.B. bei Dateien) und von Integritäts-Constraints
- Eingabefehler
- unterschiedliche Änderungsstände
- Unvollständigkeit

Multi-Source-Probleme

- unterschiedliche Repräsentationen der Instanzdaten
 - Wertebereiche (z.B. Geschlecht = {1,2} vs. Gender = {m,w})
 - Einheiten (z.B. Verkauf in EUR vs. Verkauf in Tsd.EUR)
 - Genauigkeiten
- unterschiedliche Änderungsstände und Aggregationsstufen überlappende, widersprüchliche bzw. inkonsistente Daten
- Behandlung von Duplikaten

Arten der Heterogenität



Data Cleaning

- **Datenanalyse / Profiling**
Entdeckung von Datenfehlern und -inkonsistenzen
manuell bzw. Einsatz von Analysetools
- **Definition von Mapping-Regeln und Transformations-Workflows**
Datentransformationen auf Schemaebene
Cleaning-Schritte zur Behandlung von Instanzdaten
- **Transformation**
regelmäßige Ausführung der geprüften Transformationsschritte
- **Rückfluss**
korrigierter Daten in operative Quellsysteme

Datenintegration und -bereinigung

ETL: Integration auf 2 Ebenen

Schema Integration

- Abbildung von Quell Schemata auf Data-Warehouse-Schema: Schema Matching
- Adressierung der semantischen Heterogenität (Namenskonflikte, strukturelle Konflikte)

Datenintegration / Data Cleaning

- Transformation heterogener Daten in einheitliche, durch Data Warehouse Schema vorgeschriebene Repräsentation
- Entdeckung und Behebung von Daten Qualitätsproblemen
- Entdeckung äquivalenter Objekte (Korrespondenzen auf Instanzebene): Entity Matching / Duplikat Behandlung

Namenskonflikte

Synonyme: unterschiedliche Namen für dasselbe Konzept

- Kunde - Partner
- Kunde.Adresse - Partner.Anschrift

Homonyme: gleiche Namen für verschiedene Konzepte

- (Kunde.)Name <> (Firma.)Name (Kunde ist Ansprechpartner)

Hyponymie/Hyperonyme: Unter-/Oberbegriffe

Strukturelle Konflikte

- unterschiedliche (Schlüssel NR - NUM)
- unterschiedliche Attributmengen, fehlende Attribute
- unterschiedliche Abstraktionsebenen (Generalisierung, Aggregation)
- unterschiedliche Realitäts Ausschnitte

Strukturelle Konflikte

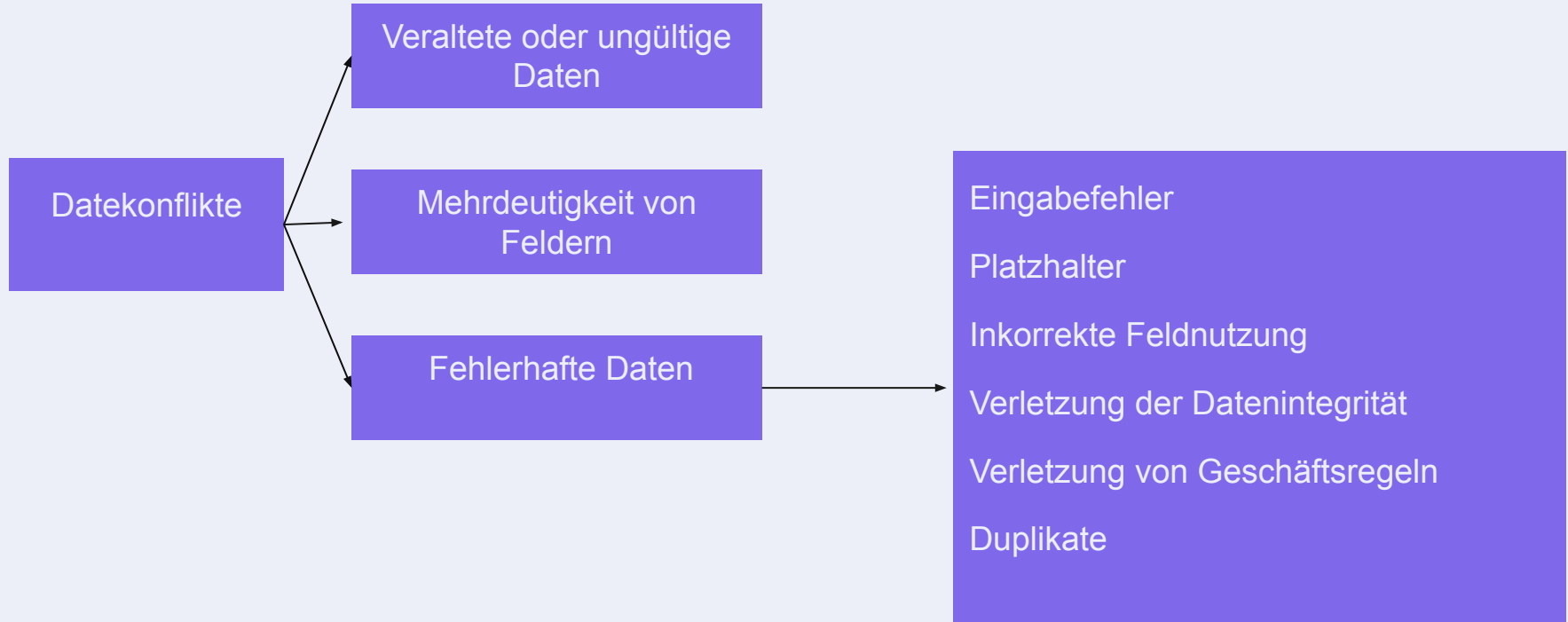
Attribut - Entities Konflikte

- Repräsentation von Attributen als eigenständige Entities/Relationen

Attribut - Attribut-Konflikte

- unterschiedliche Datentypen Preis (Float) vs. Preis (String)
- unterschiedliche Detailgrade Name vs. Vorname und Nachname
- unterschiedliche Einheiten: \$ vs. Euro
- unterschiedliche Genauigkeiten: Tausend Euro vs. Euro
- unterschiedliche Integritätsbedingungen, Wertebereiche, Default-Werte ... Alter >18 vs. Alter > 21
- unterschiedliche Zulässigkeit von Nullwerten

Datenkonflikte



Beispiel

Data Quality Issue	Example
Invalid value	Valid value can be “1” or “2”, but current value is “3”
Cultural rule conformity	Date = 1 Feb 2018 or 1-1-18 or 2-1-2018
Value out of required range	Customer age = 204
Verification	City and State do not correspond to ZIP code
Format inconsistency	Phone = +135432524 or (001)02325355

Übung

Welche Arten von Datenqualitätsfehlern treten in der Relation Kunde und Adresse auf?

KNr	Name	GEB.Datum	Alter	Geschl.	Telefon	PLZ	Email
34	Meier, Tom	21.01.1980	35	M	999-999	39107	t@r.de
34	Tina Möller	18.04.78	29	W	263222	36999	null
35	Tom Meier	32.05.1969	27	F	222231	39107	t@r.de

PLZ	ORT
39108	München
36996	Spanien
95555	Illmenau

Übung (Lösung)

Welche Arten von Datenqualitäts Fehlern treten in der Relation Kunde und Adresse auf?

KNr	Name	GEB.Datum	Alter	Geschl.	Telefon	PLZ	Email
34	Meier, Tom	21.01.1980	35	M	999-999	39107	t@r.de
34	Tina Möller	18.04.78	29	W	263222	36999	null
35	Tom Meier	32.05.1969	27	F	222231	39107	t@r.de

PLZ	ORT
39108	München
36996	Spanien
95555	Illmenau

Prozess der Transformation

Teilprozess	Beschreibung
Filterung	Extraktion und Bereinigung syntaktischer und inhaltlicher Defekte der Daten
Harmonisierung	Betriebswirtschaftliche Abstimmung der gefilterten Daten
Aggregation	Verdichtung der gefilterten und harmonisierten Daten
Anreicherung	Berechnung und Speicherung betriebswirtschaftlicher Kennzahlen

Extraction Technique

Quelle	Technik	Aktualität DWH	Belastung DWH	Belastung Quelle
Erstellt periodisch Dateien	Batchläufe, Snapshots	Je nach Frequenz	Niedrig	Niedrig
Propagiert jede Änderung	Trigger, Replikation (CDC)	Maximal	Hoch	Sehr hoch
Erstellt Extrakte auf Anfrage	Anwendungssteuerung	Je nach Frequenz	Je nach Frequenz	Je nach Frequenz

Replikation

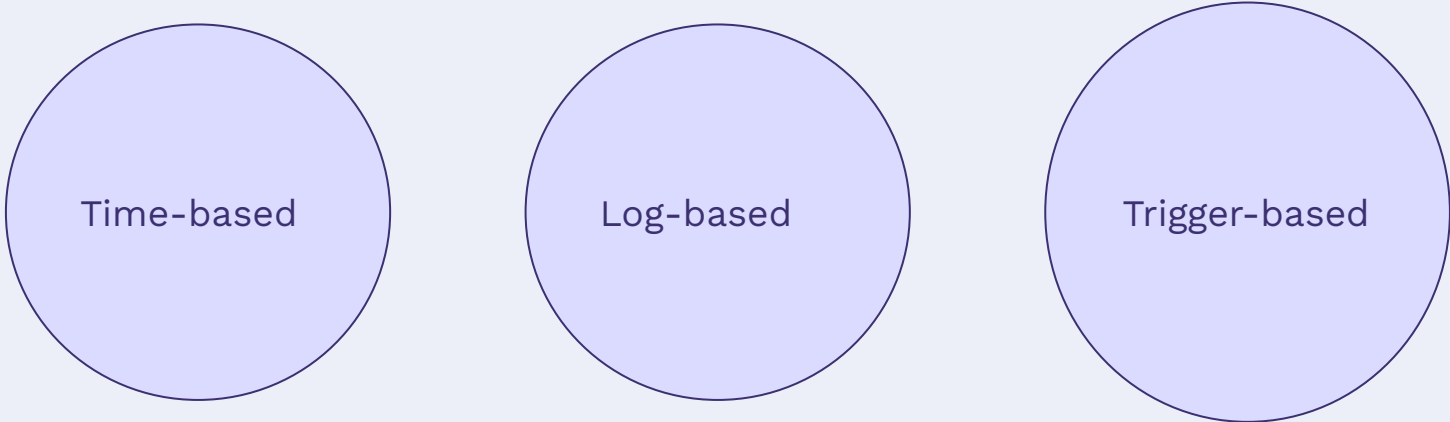
Volständige

Selektive

**Daten-
replikation
in Echtzeit**



Change Data Capture (CDC)



Time-based

The diagram consists of three light purple circles arranged horizontally. Each circle contains a text label representing a different CDC method. The first circle on the left is labeled 'Time-based', the middle circle is labeled 'Log-based', and the third circle on the right is labeled 'Trigger-based'.

Log-based

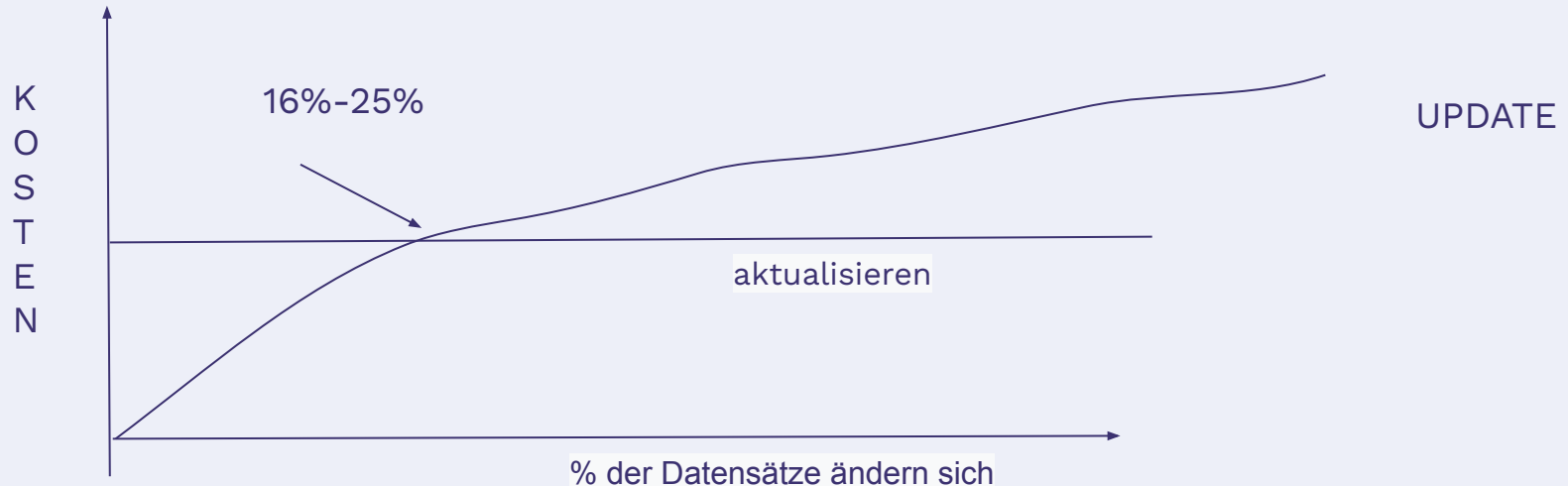
Trigger-based

Change Data Capture (CDC)

- Minimale Auswirkungen auf die Datenbank-Performance, insbesondere im Vergleich zu Datenbank-Triggern.
- Tabellen (und Spalten), die mitgeloggt werden sollen, können explizit ausgewählt werden.
- Vorhandene Tabellen in der Datenbank müssen nicht geändert werden.
- Das Verfahren ist leicht zu aktivieren bzw. zu deaktivieren.
- CDC ist bemerkenswert leicht zu konfigurieren.
- Das Verfahren kann zur Untersuchung von Update-Verhalten genutzt werden.

Wahl der Strategie

- Vollständige Aktualisierung (Refresh)
- Inkrementelle Aktualisierung (Update)



ETL Tools



Integrate.io



Talend



Informatica



Fivetran

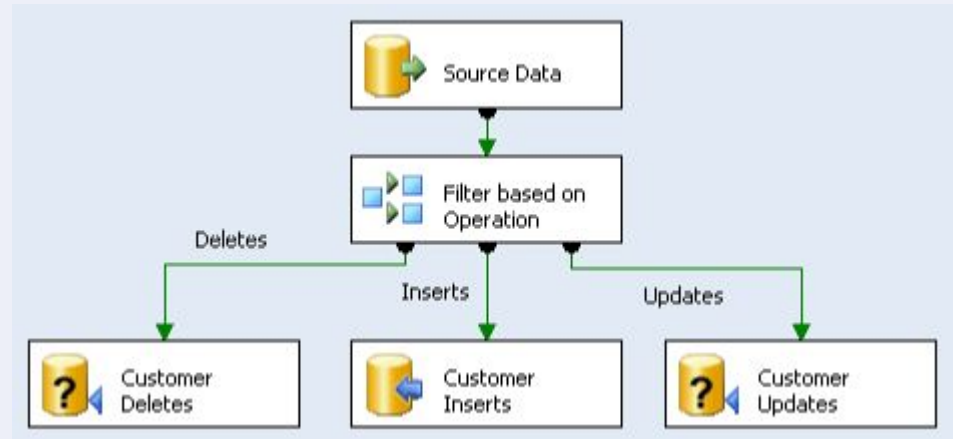


SSIS

SQL Server Integration Services SSIS (Demo)

Funktionen von SSIS:

- Datenvalidierung
- Datenbereinigung
- Aggregation
- Sortierung
- Zusammenführung von Daten aus verschiedenen Quellen
- Datenfilterung, Datenumwandlung
- Ausführen von benutzerdefinierten Skripten.



Performance-Techniken

- Index Strukturen
- Datenkompression
- Datenpartitionierung
- Materialisierte Sichten

ETL-Prozesse beschleunigen

- Mengenbasierte statt datenbasierte Verarbeitung
- Reduktion der Komplexität
- Frühzeitige Mengenbeschränkung
- Parallelisierung
- Datenbank-Konfiguration
- ELT statt ETL
- Data Lakes

DWH-Testen

- Source-to-Target-Tests
- Testen von Metadaten
- Schema-Tests
- Funktionsprüfung
- Benutzerakzeptanztests
- Leistungstest

Herausforderungen beim ETL-Testen:

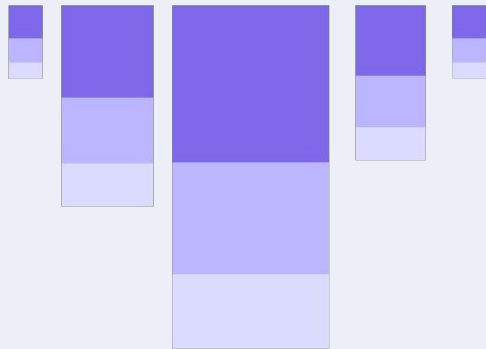
- Heterogenität der Daten
- Hohe Volumina und Skalierbarkeit
- Datenzuordnung und -transformationen

Entscheidung finden für Use Case

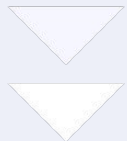
- Analysieren und verstehen Sie Ihre Daten
- Analysieren Sie, wie oft Sie Daten laden müssen
- Definieren Sie eine Datenintegrations-Richtlinie
- Wählen Sie ELT Tools oder ETL
- Wählen Sie, ob Sie DWH vor Ort oder in der Cloud wünschen

Quiz

https://docs.google.com/forms/d/e/1FAIpQLScMx2RcDowfXQzsQ1HEBsTq7h2pF2gVdI92fOtv6d1x3UwSgg/viewform?usp=sf_link



05



-
-
-

Big Data und Data Lakes

-
-
-
-

LEADERSHIP • CIO NETWORK

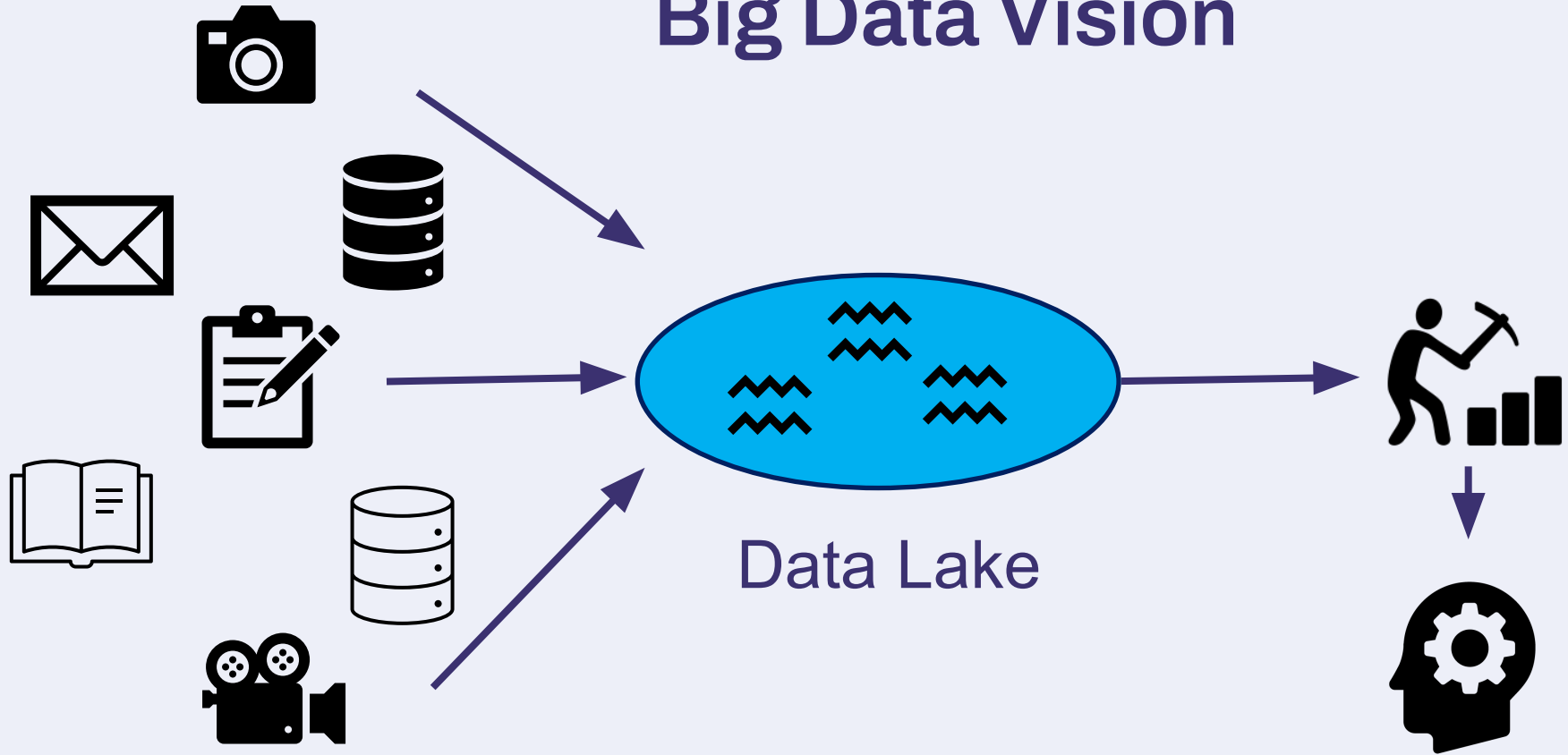
Big Data Requires a Big, New Architecture

Dan Woods Former Contributor

IT Central Contributor Group ⓘ

Jul 21, 2011, 04:30pm EDT

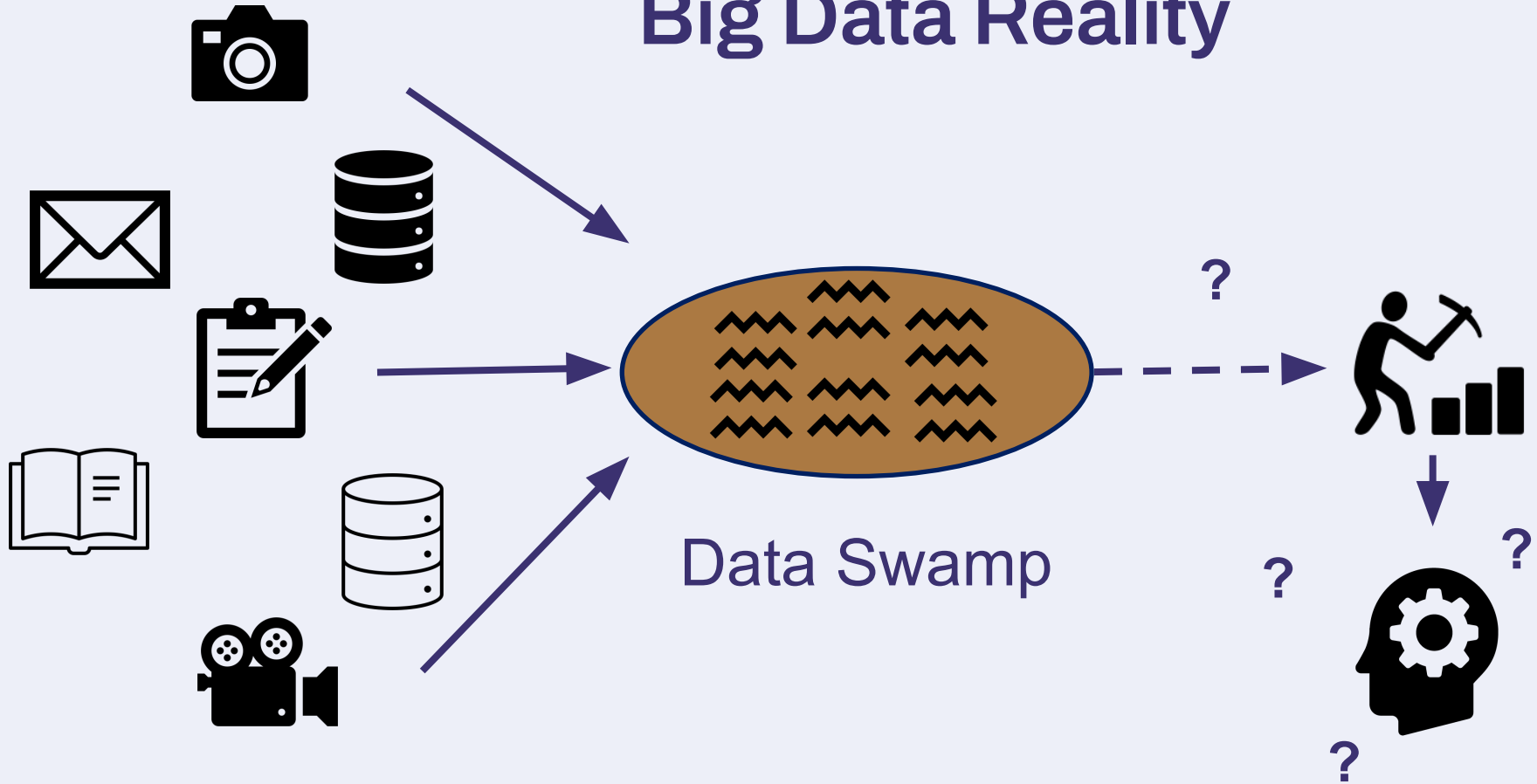
Big Data Vision



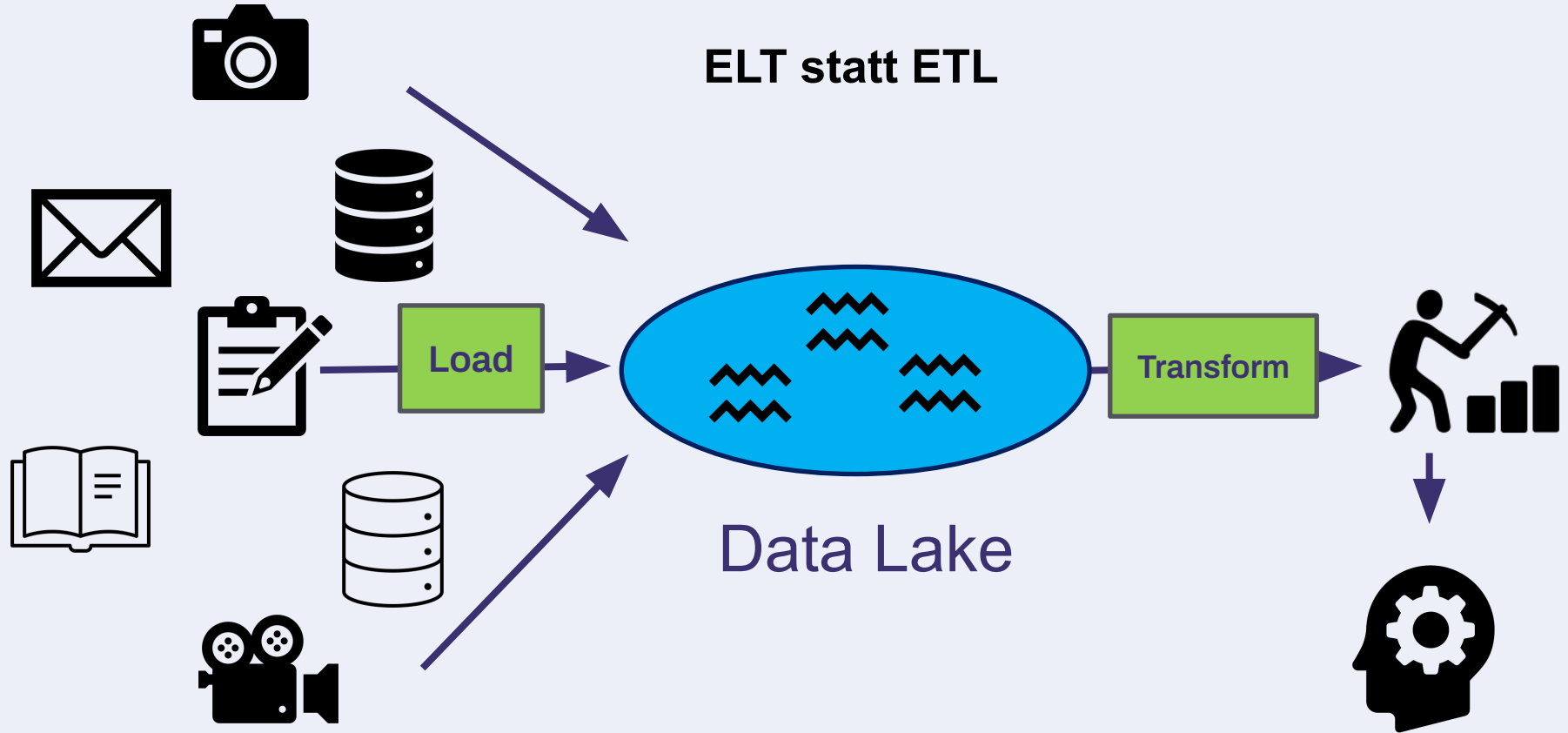
Grundprinzipien Data Lakes

- **Speicherung im rohen Format:** strukturierte, semi-strukturierte oder unstrukturierte Daten
- **Skalierbarkeit:** Data Lakes können mit dem Datenvolumen wachsen.
- **Vielseitigkeit:** Einsatzzweck steht zum Speicherzeitpunkt nicht fest -> vielseitige Analysen / ML Algorithmen möglich.
- **Anpassbarkeit:** Neue Datenquellen rasch integrierbar

Big Data Reality



ELT statt ETL



Vergleich DWH / Data Lake

Aspekt	Data Warehouse	Data Lake
Datenqualität	Daten sind hochgradig kuratiert.	Rohdaten (ggf + Metadaten)
Datennutzung	Definierter Zweck	Zweck oft noch nicht festgelegt
Kosten & Geschwindigkeit	Schnelle Ergebnisse, teuer	Langsamere Ergebnisse, günstiger
Benutzer	Geschäftsanalysten	Datenwissenschaftler, Entwickler und Geschäftsanalysten
Datentyp	Strukturierte Daten aus Geschäftsanwendungen.	Strukturierte und unstrukturierte Daten aus verschiedenen Quellen.
Datenorganisation	Schema-on-Write	Schema-on-Read
Analytik	Batch-Berichterstattung, BI-Applikationen	Maschinelles Lernen, prädiktive Analytik, Data Mining

Herausforderungen Data Lakes (1)

Datenqualität und -konsistenz:

Problem: Ungenauigkeiten, Duplikate und inkonsistente Daten können die Analyseergebnisse verfälschen.

Lösungsansätze: Implementierung von Prozessen zur Datenbereinigung, Validierung und Standardisierung.

Problem: fehlende Zuordenbarkeit, unvollständige Metadaten

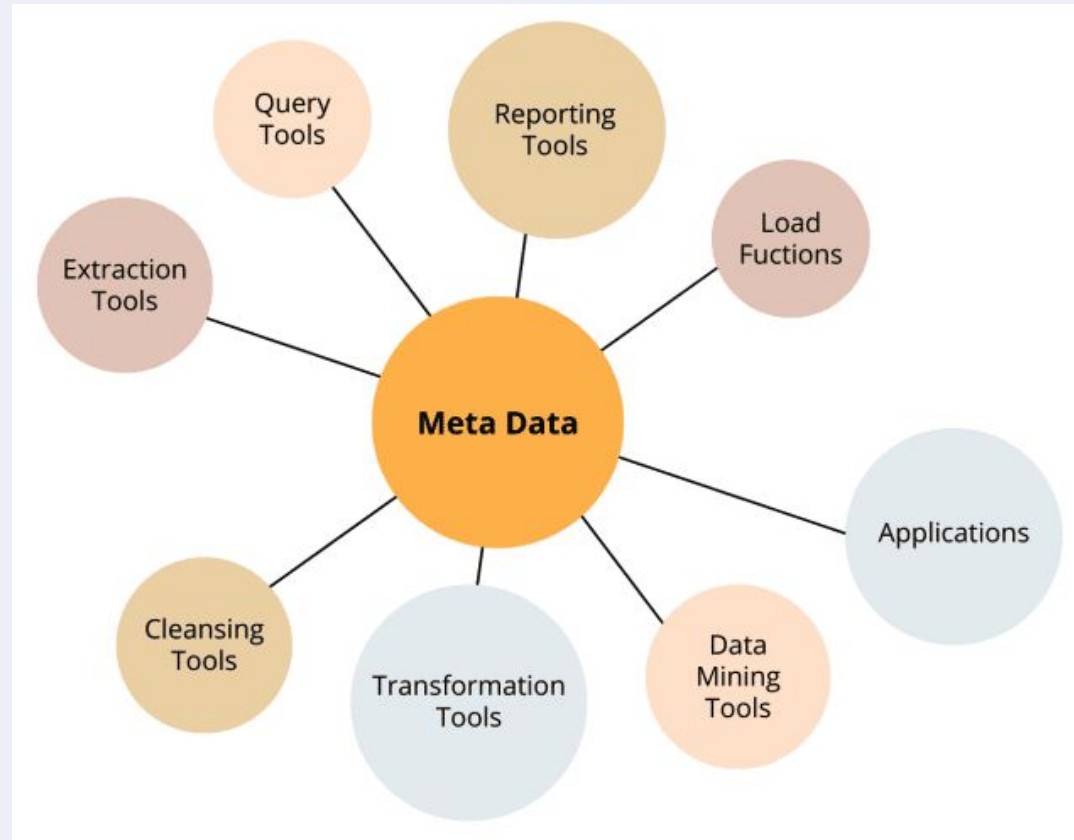
Lösungsansätze: Einsatz von Metadaten-Management-Tools zur Verbesserung der Datenfindung, -klassifizierung und des Datenverständnisses

Metadaten

Deskriptiv

Nutzungsbezogen

Administrativ



Herausforderungen Data Lakes (2)

Sicherheits- und Governance-Fragen:

Problem: Sensible Daten erfordern angemessenen Schutz und Compliance mit Datenschutzbestimmungen.

Lösungsansätze: Etablierung von Sicherheitsrichtlinien, Zugriffskontrollen und regelmäßigen Audits, Anonymisierungen

Erfolgsbeispiel für Data-Lakes

Biopharmazeutika: Astra Zeneca

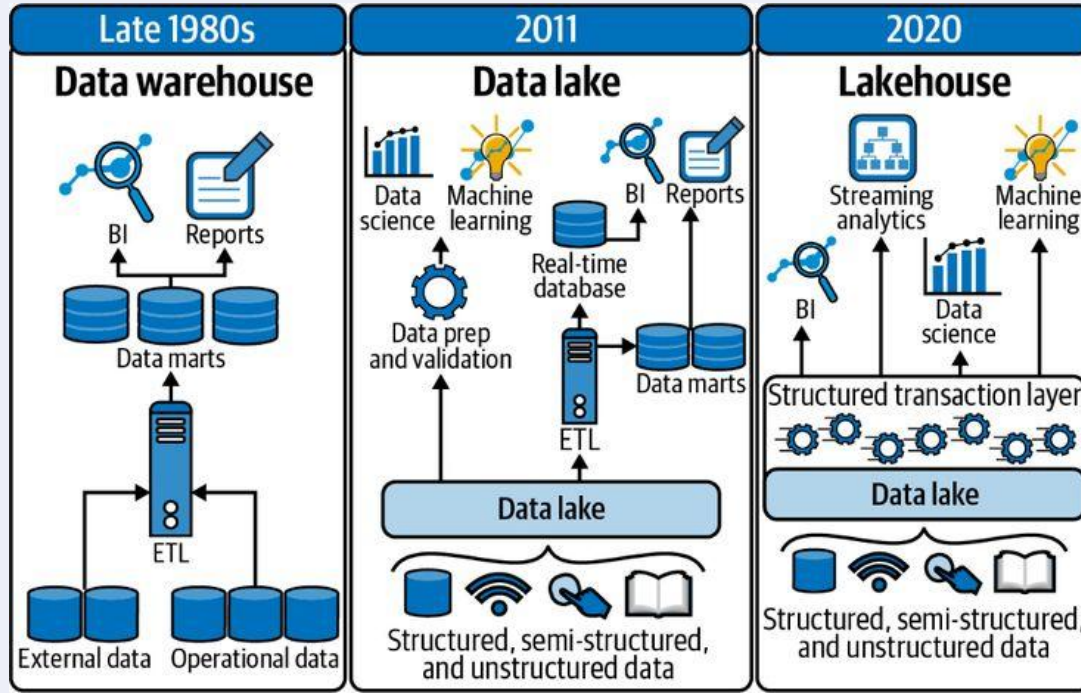
- **Implementierung eines Data Lakes:** Data Lake mit mehr als 20.000 Terabyte Daten unterstützt mehrere interne Stakeholder
- **Schnellerer Datenzugriff:** 90% aller Daten wurden für Analysen schnell verfügbar gemacht.
- **Enorme Zeiteinsparung:** Planungszyklen wurden auf lediglich drei Stunden reduziert, was eine Zeitersparnis von 99% darstellt.
- **Kosteneinsparungen:** Durch die Verkürzung der Dauer aller klinischen Studien um jeweils einen Monat sparte AstraZeneca jährlich 1 Milliarde US-Dollar.

<https://www.talend.com/de/resources/data-lake-architecture/>

Other examples:

<https://www.upsolver.com/blog/examples-of-data-lake-architecture-on-amazon-s3>

Data Lakehouse



Data Lakehouse (Auswahl)



KI im Data Warehousing

Introducing LakehouseIQ:
The AI-Powered Engine that
Uniquely Understands Your
Business



<https://www.databricks.com/blog/introducing-lakehouseiq-ai-powered-engine-uniquely-understands-your-business>

https://www.theregister.com/2023/11/16/databricks_sinks_lakehouse_in_bid/