



Moderne Datenbanken

Organisation und Einführung

Agenda

1	Begrüßung und Organisation	2
2	NoSQL-Datenbanken	10
3	Auswahl von Datenbanken	17
4	Gruppenbildung	30



Prof. Dr. Inga Marina Saatz

Raum : FH Dortmund, C.1-48

Telefon: 0231/9112-6765

eMail: inga.saatz@fh-dortmund.de

Hinweis:

Auch bei der Arbeit mit LLMs (ChatGPT & ...)
ist das Urheberrecht zu beachten!
Unter das Urheberrecht fallen alle Veranstaltungsunterlagen in ILIAS.

→ Diese dürfen nicht auf Webseiten, in LLMs, ... hochgeladen werden

Veranstaltungsübersicht

Woche	Datum	Vorlesung	Praktikum
1	20.04.2026	Einführung und Organisation Überblick NoSQL-Datenbanken und DB-Auswahl	kein Praktikum: LM Grenzen relationaler DB
2	27.04.2026	OR-DBMS	OR-DBMS & Projektproposal
3	04.05.2026	InMemory-DB und Key-Value DB	Redis
4	11.05.2026	Dokumentenorientierte DB	MongoDB
	18.05.2026	Blockwoche	
	25.05.2026	Pfingsten	
5	01.06.2026	Graph-Datenbanken	Neo4j
6	08.06.2026	Zwischenpräsentation	
7	15.06.2026	Multimodale Datenbanken	Projekt
8	22.06.2026	Datenbankarchitekturen	
9	29.06.2026	Spaltenorientierte Datenbanken	
10	06.07.2026	Polyglot Persistence	
11	13.07.2026	Endpräsentation	
12	20.07.2026	Klausurvorbereitung	kein Praktikum

Prüfungsleistung	Was?	Bewertung
Projekt	Projektproposal	2% SPL
	Zwischenstandspräsentation inkl. Dokument.	10% SPL
	Präsentation Datenbankauswahl inkl. Livedemo	10% SPL
	Dokumentation der Marktrecherche, Nutzwertanalyse inkl. Performancevergleichs Kommentierter Quellcode	8% SPL
Schriftliche Prüfung	Klausur	70% PL

Themenliste der Anwendungsszenarien und Buchungspool in ILIAS

Inhalte des Projektproposals (Upload in ILIAS):

- Projektbeteiligte
- Anwendungsszenario (inkl. ER-Modell) und betrachtete Anwendungsfälle und Integritätsbedingungen
- Vergleichskriterien für die Datenbankauswahl mit Begründung basierend auf dem Anwendungsszenario)
- Aufgabenteilung innerhalb der Projektgruppe
- Meilensteine und Terminplan
- Umfang: 600-800 Wörter

Zwischenstandspräsentation inkl. Dokumentation:

- Inhalte:
 - Anwendungsszenario
 - Dokumentation der Datenmodellierung (Key-Value, Dokumentorientiert, Graph DB-Modell) des Anwendungsszenarios *(hier können relevante Ausschnitt des ER-Modells verwendet werden)*
 - Vergleich der Datenmodellierungen
 - Vor- und Nachteile der einzelnen Modelle sind durch Beispiele aus der Modellierung des Anwendungsszenarios zu belegen
 - Fazit: Auswahl des NoSQL-Datenbankmodells (inkl. Begründung)
- Umfang: Dokumentation 6-8 Seiten
- Präsentationsform: Pecha-Kucha
(20 Folien mit 20sec pro Folie, d.h. 6,5 Minuten insgesamt zzgl. Diskussion)

■ Abschlusspräsentation

• Inhalte

- Marktrecherche zum ausgewählten NoSQL-Datenbankmodell (→ 6 DBs, davon ≥ 1 Multimodale DB)
- Datenbankauswahl (inkl. Begründung durch eine Nutzwertanalyse)
(3 aus 6) „eine weitere DB pro Nase → Vergleich mit der gewählten DB aus der Zwischenpräsentation“
- Performancevergleich der ausgewählten Datenbanken für 2 Anwendungsfälle
- Live-Demo
- Fazit: Handlungsempfehlung und Reflexion (lessons learned)

• Präsentationsform

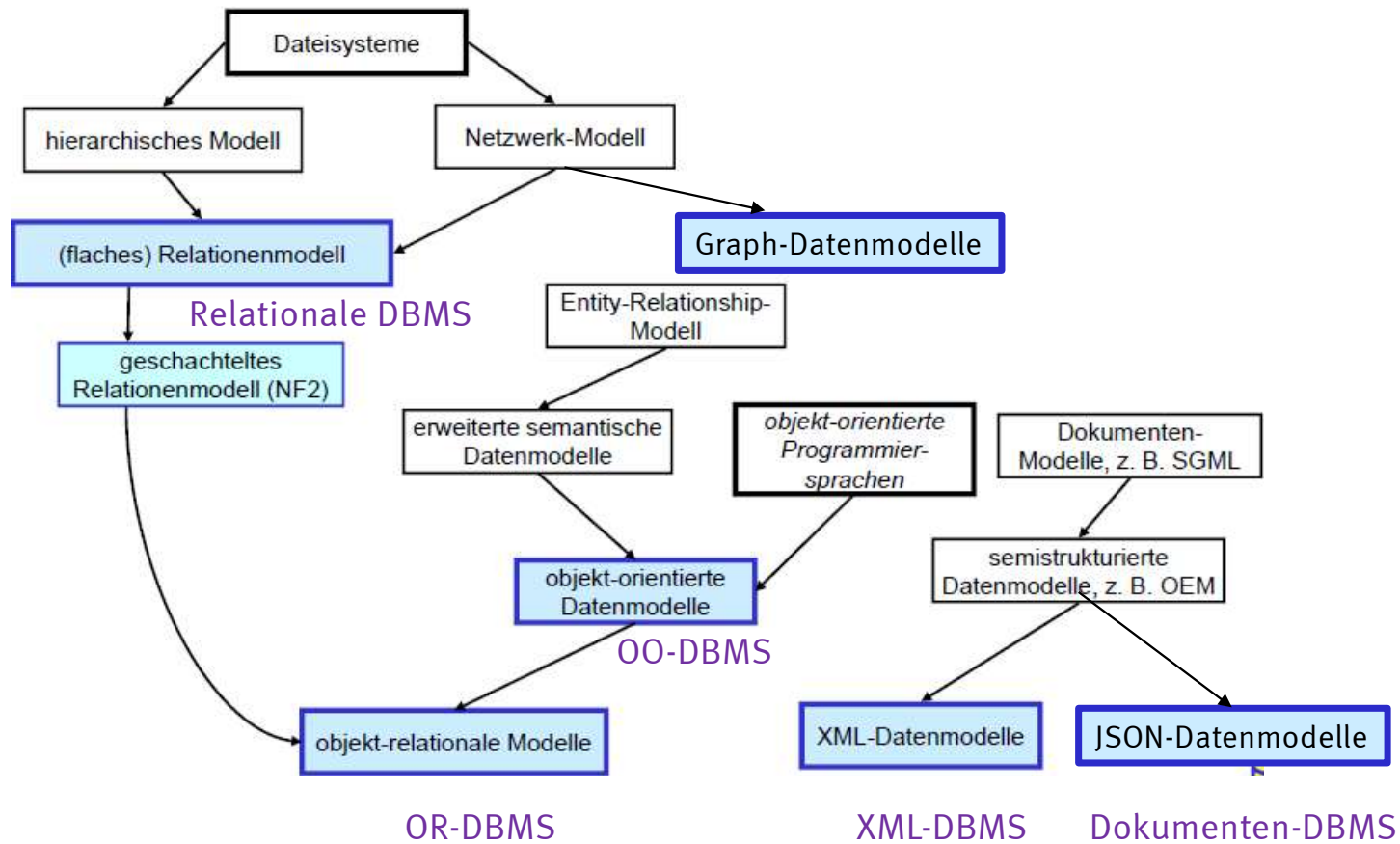
- Umfang der Dokumentation 8-12 Seiten (Upload in ILIAS)
- Posterpräsentation
 - Das Poster (Din A0)
 - Handouts (ausgedruckt zum Verteilen)
 - Upload von Poster und Handout in ILIAS

Agenda

1	Begrüßung und Organisation	2
2	NoSQL-Datenbanken	10
3	Auswahl von Datenbanken	17
4	Gruppenbildung	30

Jedem DBS liegt ein **Datenmodell** (Konzept) zugrunde, das festlegt,

- welche **Eigenschaften** die Datenelemente besitzen,
- welche **Struktur die Datenelemente** besitzen dürfen,
(z.B. Relationen, Tupel ...)
- welche **Konsistenzbedingungen** einzuhalten sind,
(Integritätsbedingungen zur weiteren Einschränkung der Struktur)
- welche **Operationen** zum Speichern, Suchen, Ändern und Löschen von Datenelementen existieren.
- Beispiele für Datenmodelle:
 - Hierarchisches Datenbankmodell
 - Netzwerkdatenbankmodell
 - **Relationales Datenbankmodell**
 - Objektorientiertes Datenbankmodell
 - Deduktive Datenbanken (→ logische Regeln, ...)
 - sowie Mischformen, u.a.
 - Objekt-Relationales Datenbankmodell
 - XML-Datenbanken



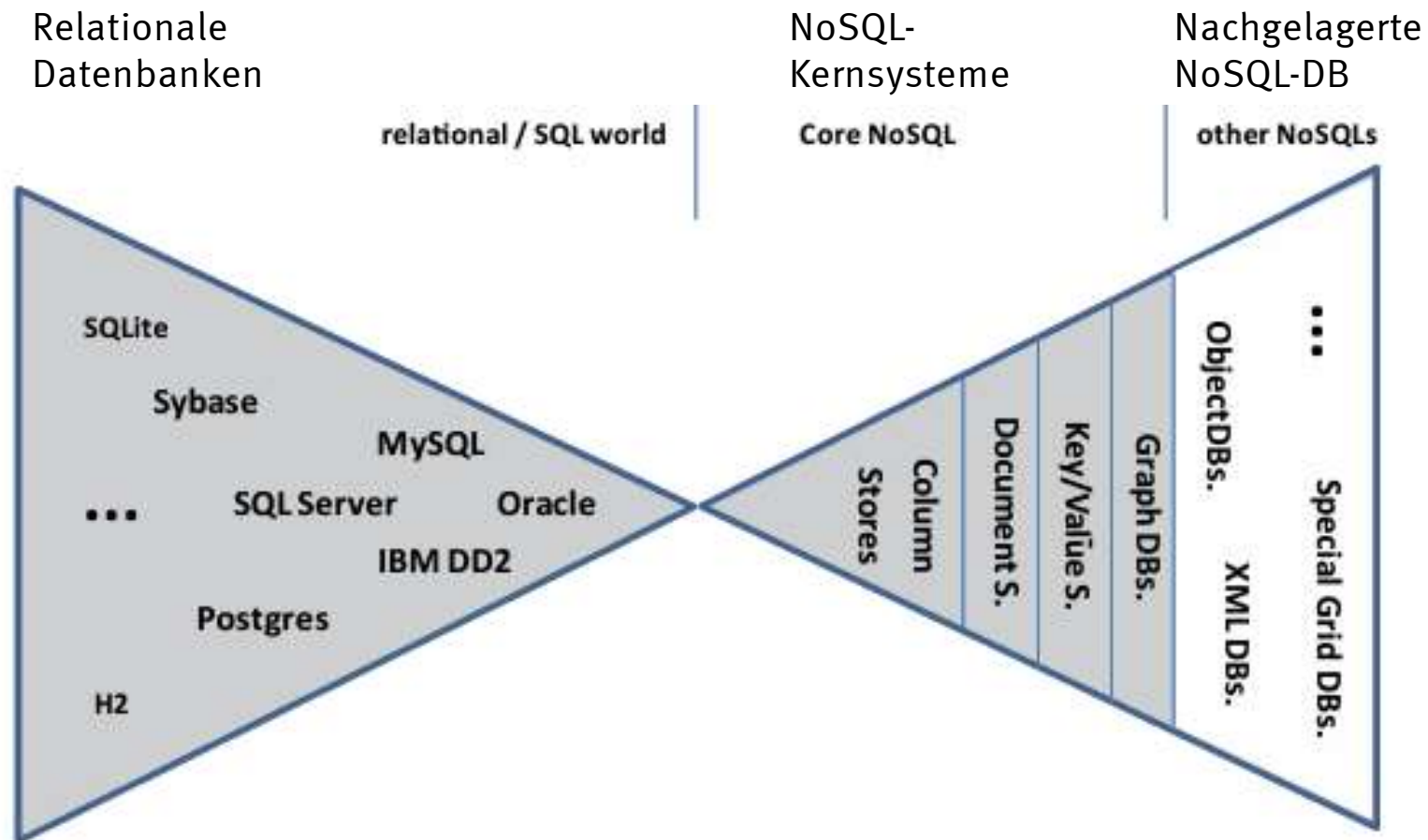
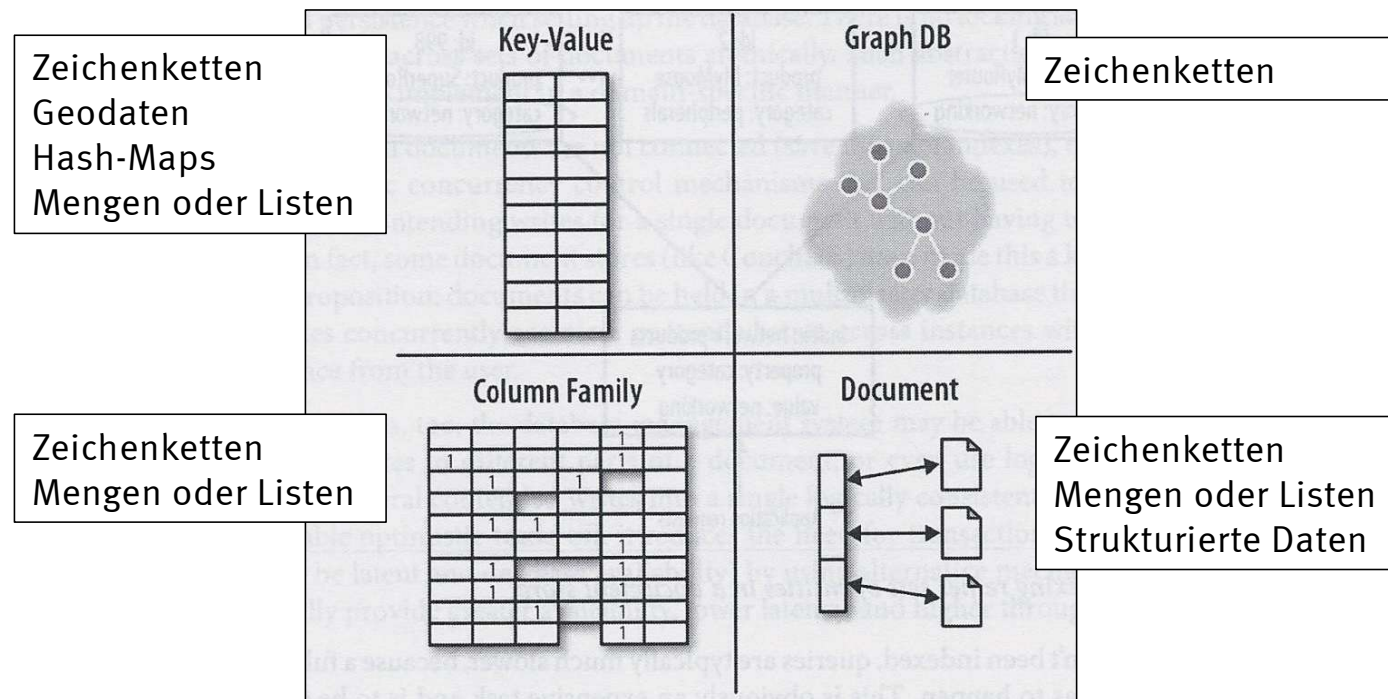


Abb. Entnommen aus (Edlich 2011: 6)

Die von NoSQL-Datenbanken unterstützten reichen von einfachen Schlüssel Wertspeichern bis hin zu komplexen Objekten und strukturierten Dokumenten.



Multimodale Datenbanken basieren auf kombinierten Datenbankmodelle.

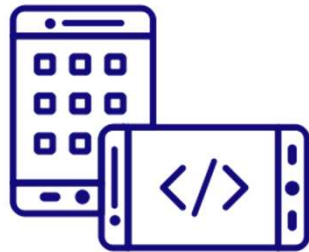
- 1979 Key/Hash Datenbank DBM (Ken Thompson)
- 1980er Erste NoSQL-Systeme Lotus Notes, BerkleyDB, GT.M
- 1998 Begriff NoSQL erstmals verwendet von Carlo Strozzi für eine relationale DB ohne SQL-API
-
- seit 2000 Erhöhte Anforderungen durch Verarbeitung großer Datenmengen im Web2.0
- 2000-2004 Google Inc. entwickelt Map/Reduce –Ansatz und BigTable-DBMS (2004) auf einem eigenen Filesystem (GFS)
- Bis 2005 Weitere Ansätze
 - db4o: objekt-orientierte DB
 - Neo4j: Graphorientierte DB
 - memcached: Key/Value-System
 - InfoGrid: Graphorientierte DB
- 2006-2009 Entstehen weiterer DBMS (z.B. HBase, CouchDB, MongoDB, ...)
- Mai 2009 Begriff NoSQL verwendet für einen Event am 11.06.2009 in San Francisco zu „distributed data storage“-Systemen (Evans 2009)
- 2018 >225 NoSQL-Datenbanken sind im NoSQL-Archiv (Edlich 2018)

Unter **NoSQL** („Not Only SQL“) werden Datenbanksysteme **mit einigen** der folgenden Eigenschaften verstanden:

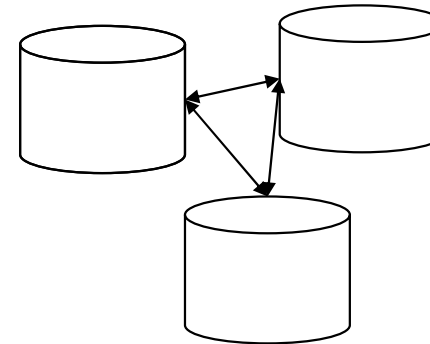
1. Nicht-Relationales Datenbankmodell
2. Schemaflexibilität
3. Einfache API (z.B. REST-API, MapReduce-Abfragen)
4. Extreme Skalierbarkeit
5. Einfache Datenreplikation basierend auf einer verteilten Architektur
6. Verwendung eines (alternativen) Konsistenzmodells, z.B. BASE

Nach (Endlich 2011: 2)

mobile Datenbank CouchBaseLite



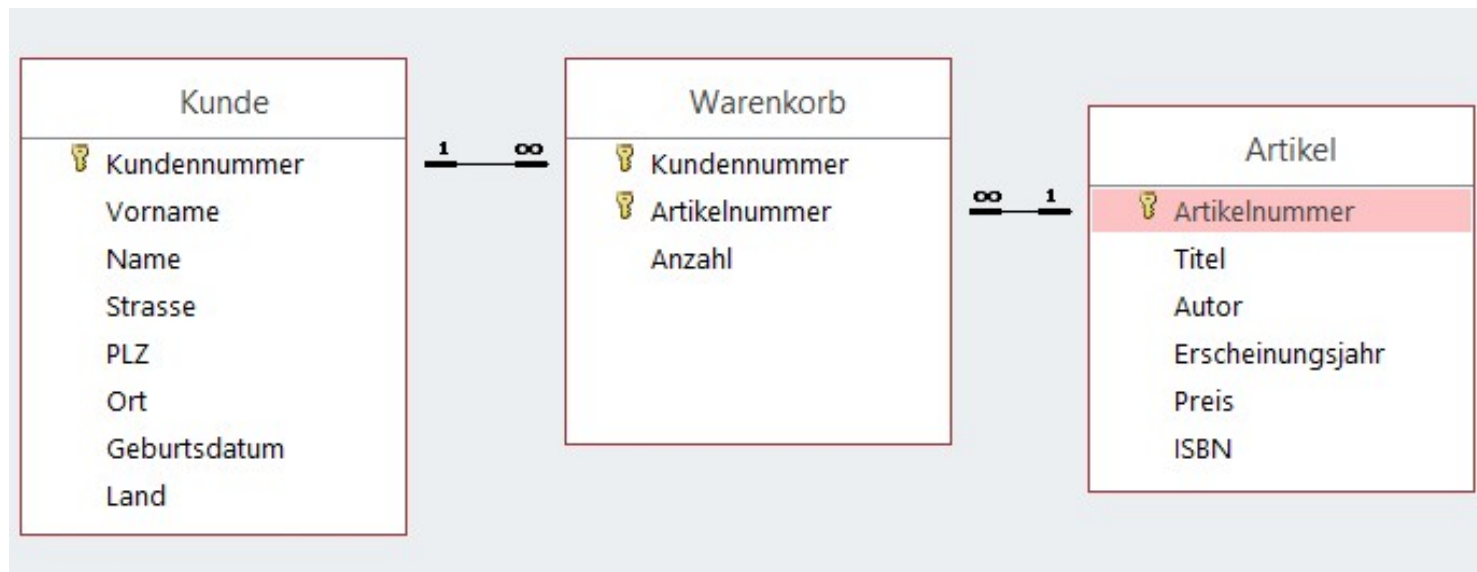
Cassandra-Cluster



Agenda

1	Begrüßung und Organisation	2
2	NoSQL-Datenbanken	10
3	Auswahl von Datenbanken	17
4	Gruppenbildung	30

Buchhandlung Buchfix



Probleme:

- Es kann immer nur ein Mitarbeiter mit der Datenbank arbeiten
- Performanzprobleme

Auswahlkriterien ermitteln und gewichten

Auswahlkriterien	Muss-Kriterium	Gewicht
Mehrbenutzerbetrieb	ja	5
ACID-Transaktionen	ja	5
Hohe Performanz	ja	4
Geringe Lizenzkosten	ja	2
SQL-Anfragen	nein	3
Severbasierte Datenbank	nein	3
Usability	nein	1



The screenshot shows the homepage of the DB-Engines website. The header features the 'DB-Engines' logo on the left and the 'GraphDB by InfluxDB' logo on the right. Below the logos, a navigation bar contains links for 'Home', 'DB-Engines Ranking', and 'System'. A 'Featured Products' section lists 'DataStax' and 'Couchbase'. The main content area is titled 'DB-Engines' and contains three paragraphs of text. The first paragraph describes the initiative's purpose. The second paragraph mentions the 'DB-Engines Ranking'. The third paragraph discusses the 'Datenbankmanagementsysteme' overview. The fourth paragraph, partially visible, mentions the 'Datenbank-Enzyklopädie'.

DB-Engines

Informationen zu relationalen und NoSQL Datenbankmanagementsystemen

[Home](#) | [DB-Engines Ranking](#) | [System](#)

Featured Products: [DataStax](#) [Couchbase](#)

DB-Engines

DB-Engines ist eine Initiative zur Sammlung und Präsentation von Informationen zum Thema Datenbankmanagementsysteme (DBMS). Neben den bewährten relationalen DBMS bilden Systeme und Konzepte aus dem stark wachsenden NoSQL-Sektor einen Schwerpunkt.

Das [DB-Engines Ranking](#) ist eine Rangliste der aktuellen Popularität einzelner Systeme, die monatlich aktualisiert wird.

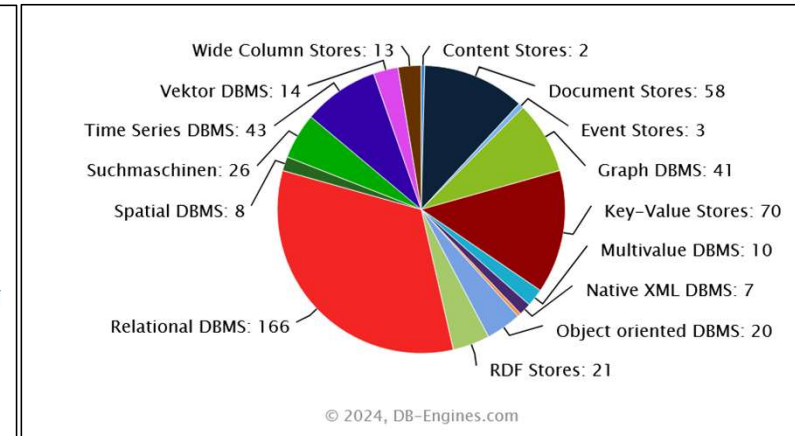
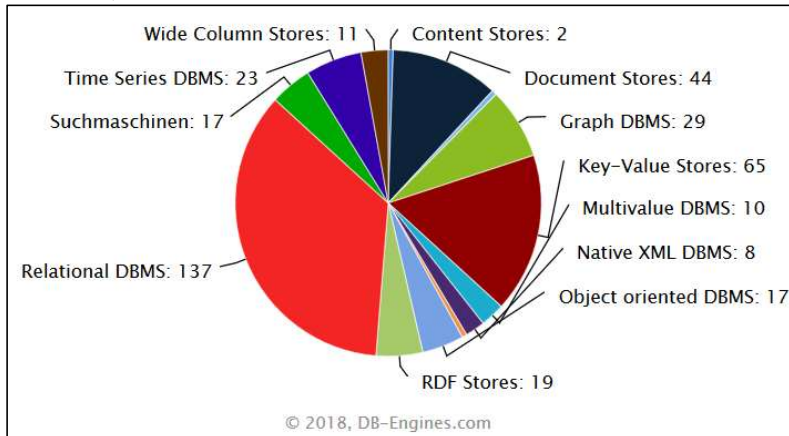
In der Übersicht über [Datenbankmanagementsysteme](#) sind die wichtigsten Eigenschaften zahlreicher Systeme dargestellt. Diese können einfach abgefragt und auch für mehrere Systeme übersichtlich miteinander verglichen werden.

In der [Datenbank-Enzyklopädie](#) werden Begriffe und

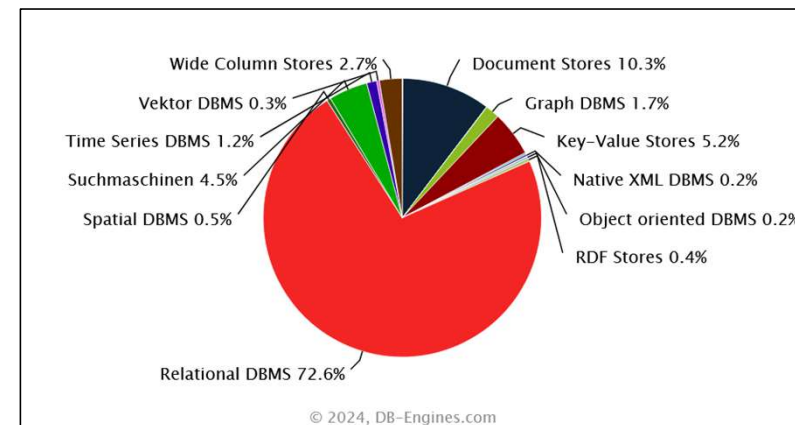
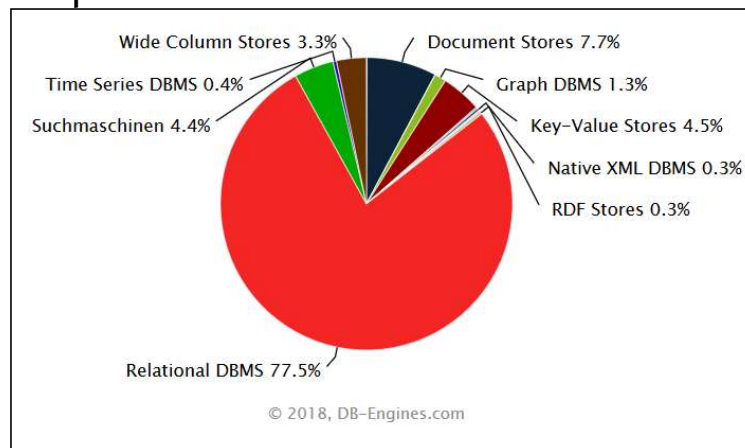
www.db-engines.com/de

Überblick – Systeme und Popularität (2018 vs 2024)

Anzahl der DB



Popularität

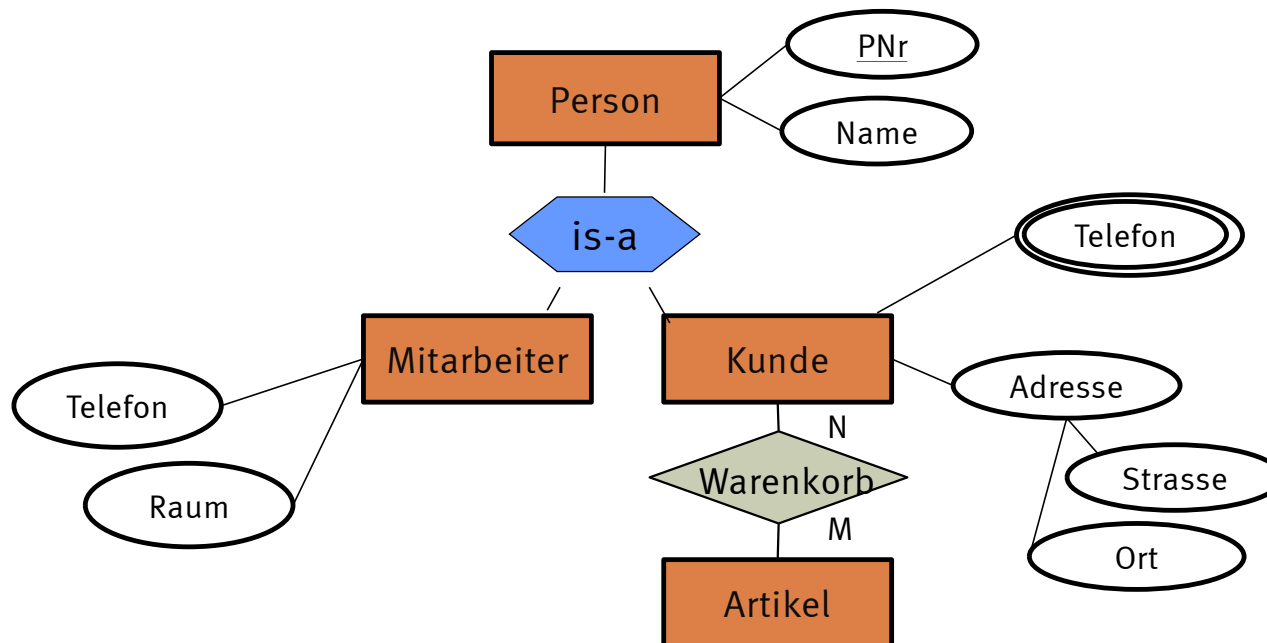


https://db-engines.com/de/ranking_categories

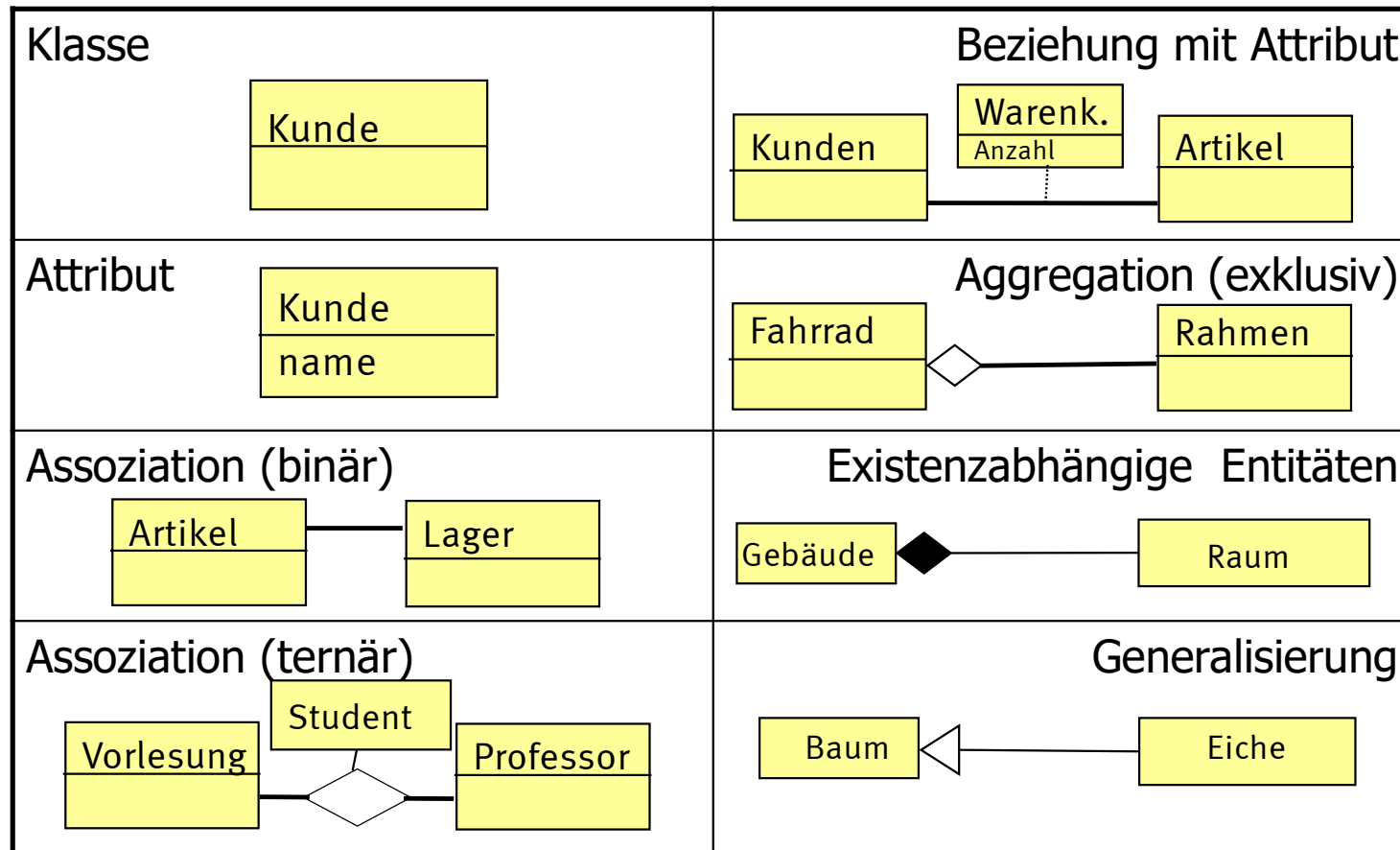
1. Welche Datenarten sollen gespeichert werden?
 - Text
2. Welches ist das geeignete Daten- und Speichermodell?
 - In welcher Form liegen die Daten vor? Relational
 - → geeignetes Datenmodell: Relational (zeilen- oder spaltenorientiert)
3. Wie viele Beziehung bestehen zwischen den Daten?
 - Kunde → Bestellung → Artikel
4. Wie groß ist die zu speichernde Datenmenge?
 - $100 \times 365 \times 100k = 4 \text{ GB pro Jahr}$
 - Es sind etwa 100 Bestellungen pro Tag zu bewältigen
 - → Es ist keine Replikation erforderlich
5. Wie komplex sind die zu speichernden Daten?
 - Wie komplex ist das Datenmodell
 - Werden durch den Anwender neue Datentypen definiert?
 - Können O/R-Mapper die Datenkomplexität auf ein relationales Modell abbilden?

Erweiterung des ER-Diagramms, beispielsweise um:

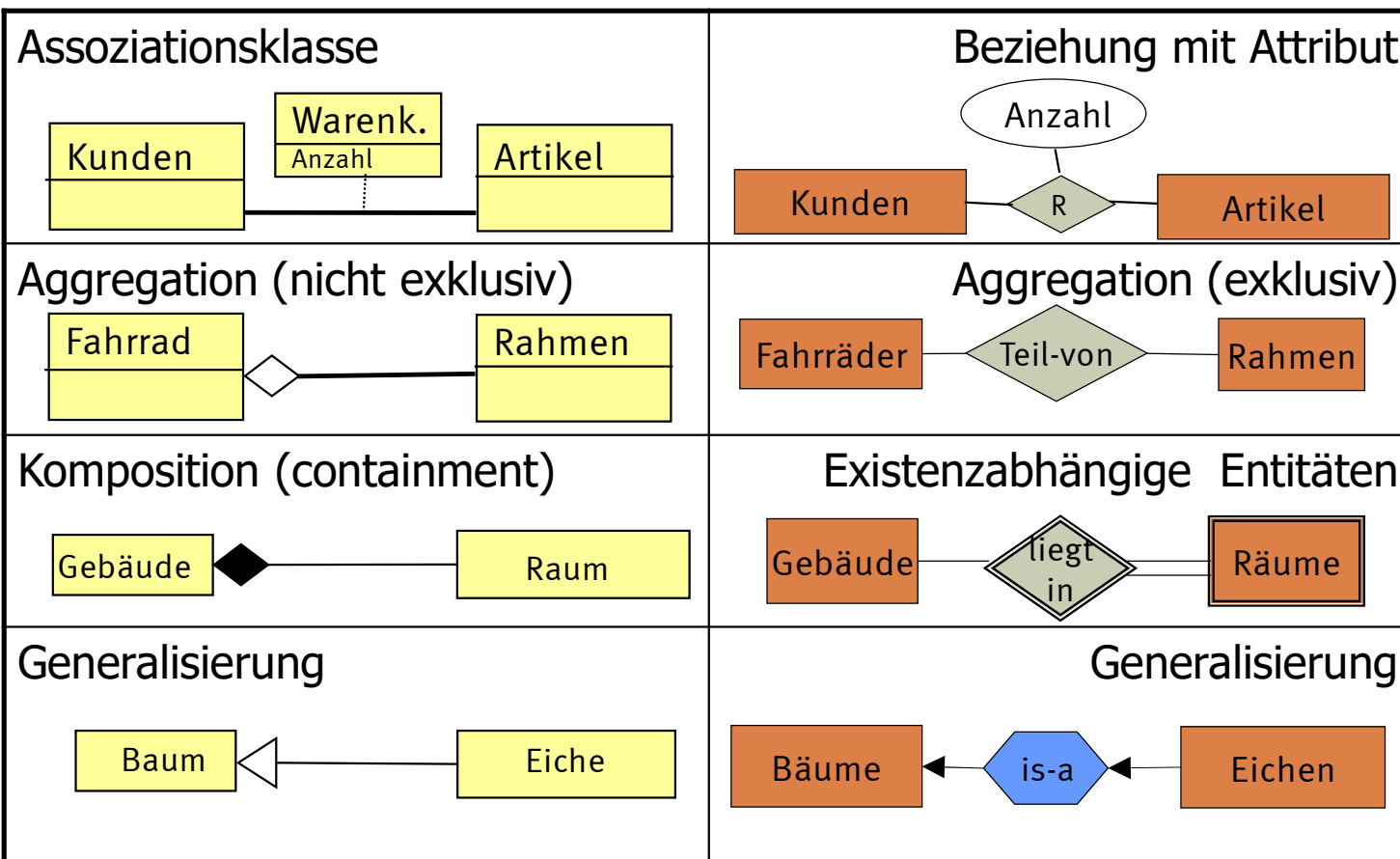
- Mehrwertige Attribute
- Strukturierte Attribute
- Vererbungsbeziehungen



EER-Modell vs UML-Klassendiagramm



UML-Modell vs. ER-Modell



1. Kann das Szenario durch das Datenbankmodell modelliert werden?
2. Werden die weiteren Muss-Kriterien erfüllt?

Auswahlkriterien	Muss	Access	MariaDB	MonetDB
Mehrbenutzerbetrieb	ja	✗	✓	✓
ACID-Transaktionen	ja	✓	✓	✓
Hohe Performanz	ja	✗		
Geringe Lizenzkosten	Ja	✓	✓	✓
SQL-Anfragen	nein	✓	✓	✓
Severbasierte Datenbank	nein	✗	✓	✓
Usability	Nein			

Vorauswahl der Kandidaten: Vergleich der Systemeigenschaften

Auswahlkriterien	Muss	Gewicht	MariaDB	MonetDB
Mehrbenutzerbetrieb	ja	5	2	2
ACID-Transaktionen	ja	5	2	2
Serverbasierte Datenbank	nein	4	2	2
Geringe Lizenzkosten	ja	2	2	2
SQL-Anfragen	nein	3	2	2
Hohe Performanz	ja	3	?	?
Usability	Nein	1	?	?
Zwischensumme			42	42

Absolutes Bewertungsschema

0 Punkt nicht vorhanden

1 Punkt eingeschränkt vorhanden

2 Punkte vorhanden

Vorauswahl der Kandidaten: Evaluation der Datenbanken

Auswahlkriterien	Muss	Gewicht	MariaDB	MonetDB
Mehrbenutzerbetrieb	Ja	5	2	2
ACID-Transaktionen	Ja	5	2	2
Serverbasierte Datenbank	nein	4	2	2
Geringe Lizenzkosten	Ja	2	2	2
SQL-Anfragen	Nein	3	2	2
Hohe Performanz	Ja			
- Lesen		2	?	?
- Schreiben		1	?	?
Usability	Nein	1	?	?
Zwischensumme			42	42

Evaluationsprototyp: Benchmark-Test

Ziel: Messung der Performanz
-beim Lesen
-beim Schreiben

Auswahlkriterium	Gewicht	MariaDB	MonetDB
Performanz			
- Lesen	2	?	?
- Schreiben	1	?	?
Usability	1	?	?

Vergleichendes Bewertungsschema

0 Punkte

nicht vorhanden

1 Punkt

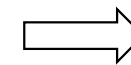
ist schlechter

2 Punkte

beide gleich gut

3 Punkte

ist besser



Praxistest

Agenda

1	Begrüßung und Organisation	2
2	NoSQL-Datenbanken	10
3	Auswahl von Datenbanken	17
4	Gruppenbildung	30

- Dem ILIAS-Kurs beitreten
- Gruppenbildung
- Auswahl eines Projekts und Buchung im Themenpool
- Lernmodul „Einführung in Docker“
- Installation von Docker

**Vielen Dank
für Ihre Aufmerksamkeit**