



Datenbanken

Dipl.-Ing. Jürgen Wemheuer

WS 2011-2012

Inhaltsübersicht



2

- **Einführung in Datenbanksysteme:**
Grundbegriffe, Entwicklung, Aufbau...
- **Relationale Datenbanken:**
Modellierung, CODD-Regeln, Beziehungen...
- **SQL:**
Standards, Syntax, Anwendungen...
- **Weiterführende Themen:**

Was sind Daten?



3

Daten sind:

Gebilde aus Zeichen oder
kontinuierliche Funktionen, die

- zum Zweck der Verarbeitung
- aufgrund bekannter oder unterstellter Vereinbarungen

Informationen darstellen.

(DIN ISO/IEC 2382)

Datenbanken IKA 8/10 - 1

Woraus bestehen Daten?



4

Daten sind im Allgemeinen in der Informationstechnik
gekennzeichnet durch:

- eine **Bezeichnung**
(Bezeichner, Name)
- einen **Datentyp**
(Ganzzahl, Dezimalzahl, Zeichen, Text etc.)
- einen **Wert**
(die informationstechnisch codierte Information an sich)
OBACHT: „Nichts“ ist nicht der Codewert „0“ sondern NULL

Datenbanken IKA 8/10 - 1

Zum Sprachgebrauch



5

lat.: dare (geben), datum = das Gegebene
 Ein „Datum“ wird umgangssprachlich bevorzugt verwendet als Bezeichnung für einen „bestimmten Tag im Kalender“.

In der Informationstechnik ist „Datum“ als Singular für „Daten“ sehr geläufig, für den Kalendertag verwenden wir dann zur Unterscheidung besser den Begriff „Tagesdatum“ oder „Kalenderdatum“.

Datenbanken IKA 8/10 - 1

Datum: Bezeichnung



6

Bezeichner (Namen) werden benötigt, um in einem Rechnersystem ein bestimmtes Datum abzulegen und wiederzufinden (der Rechner wandelt diese Bezeichnung in einen Zeiger auf die entsprechende physikalische Speicherstelle um).

Bei einem einmalig auftretenden Datum reicht die Vergabe eines Namens (z.B. „MeinDatum“), bei ein- oder mehrdimensionalen Daten („Tupel“, „Matrix“) werden ein oder mehrere (i.d.R. durch Punktoperatoren abgetrennte) Namensbestandteile (z.B. Schueler.Vorname, Schueler.Nachname) und/oder ein Index oder mehrere Indizes hinzugefügt (z.B. Listeneintrag[20], SpielfeldMatrix[8][8]).

Datenbanken IKA 8/10 - 1

Datum: Datentyp und Werte



7

Datentypen bestimmen:

- welchen Speicherplatz (Größe, Anzahl Bytes) ein Datum physikalisch (im Arbeitsspeicher, auf einem Speichermedium) beansprucht
- wie ein Wert intern codiert wird:
 - 4 als Ganzzahl: 0000 0000 0000 0100 (2 Byte Short Integer)
 - 4 als ASCII-Zeichen: 0011 0100 (1 Byte Character)
 - 4 als Dezimalzahl: 0100 0000 1000 0000 0000 0000 0000 0010
- wie ein Wert zu interpretieren ist und verarbeitet werden kann: Zahl 5 multipliziert mit Zeichen ‚Y‘ ist unsinnig...
- den Wertebereich, den ein Datum annehmen kann: (z.B. `unsigned short int` für Ganzzahlen von 0...65535)

Datenbanken IKA 8/10 - 1

Einführung in Datenbanksysteme



8

- Warum Datenbanken?
 - Anforderungen an Datenbanksysteme
 - Datenbank-Grundbegriffe
 - Historische Entwicklung
 - Von der Realität zur Datenbank
 - Einordnung des Themengebietes
 - Praktische Anwendungen
 - Wichtige Grundbegriffe
- Kenntnis der Ziele und Einsatzmöglichkeiten von Datenbanken

Datenbanken IKA 8/10 - 1

Warum Datenbanken?



9

- Wozu brauchen wir überhaupt „Datenbanken“?
- Dateien und Dateisysteme reichen doch aus?
- Oder etwa nicht?

Datenbanken IKA 8/10 - 1

Warum Datenbanken?



10

- Problemfelder mit einfachen Dateisystemen
 - Strukturierung: was sagen die Daten aus?
 - Integrität: wie bleiben die Daten konsistent?
 - Redundanz: gleiche Information oft mehrfach geführt
 - Koordination: was passiert bei mehreren Benutzern/Programmen?
 - Sicherheit: wer schützt meine Daten?
 - Leistung: z.B. Suchen, Einfügen, Sortieren
- Solche Probleme lassen sich mit Datenbanken lösen...

Datenbanken IKA 8/10 - 1

Anforderungen an Datenbanksysteme (1)



11

- **Vielfachverwendbarkeit**
 - die jeweilige Datenstruktur einer Datenbank soll für eine möglichst große Anzahl verschiedener (möglicherweise noch gar nicht bekannter) Anwendungen geeignet sein
- **Benutzerfreundlichkeit**
- **Datenunabhängigkeit**
 - kein Einfluss der Speicherungs- und Zugriffstechniken auf die jeweilige Datenstruktur (und umgekehrt)
- **Datenneutralität**
 - keine Festlegung der Struktur zugunsten einer bestimmten Anwendung

Datenbanken IKA 8/10 - 1

Anforderungen an Datenbanksysteme (2)



12

- **Flexibilität**
 - Anwendungen auf der Datenbank können mit verschiedenen Implementierungskonzepten und flexiblen Schnittstellen realisiert werden (z.B. sowohl lokal als auch über Inter- und Intranets)
- **Redundanzfreiheit**
 - keine Mehrfachablage identischer Daten
- **Konsistenz der Daten und Programme**
 - das Datenbanksystem sichert zu, dass die logischen Regeln für die Daten und die Funktionen zur Bearbeitung von Daten durchgehend angewandt werden

Datenbanken IKA 8/10 - 1

Anforderungen an Datenbanksysteme (3)



13

- Unterstützung der Integrität
 - das Datenbanksystem sichert zu, dass Datenstrukturen und gespeicherte Daten unversehrt bleiben
- Datensicherheit
 - das Datenbanksystem schützt die gespeicherten Daten vor Verlust
- Unterstützung des Datenschutzes
 - das Datenbanksystem wahrt die Persönlichkeitsrechte (sofern es personenbezogene Daten speichert)

Datenbanken IKA 8/10 - 1

Datenbanken: Grundbegriffe



14

Ein Datenbanksystem ist ein Informationssystem, das Daten

- nach einem einheitlichen Organisationskonzept („Datenmodell“) erfasst und verwaltet,
- möglichst redundanzfrei speichert (d.h. es gibt nach Möglichkeit keine identischen Datensätze) und
- für unterschiedliche, teilweise nicht vorhergesehene oder vorhersehbare Zwecke bereitstellt.

Datenbanken IKA 8/10 - 1

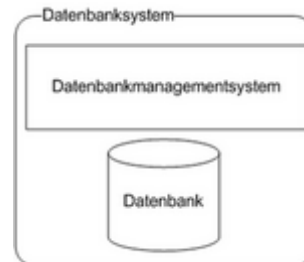
Datenbanken: Grundbegriffe

BBS

15

Eine Datenbanksystem besteht aus:

- **Datenbank / Datenbasis (DB):**
Geschlossener, in sich stimmiger Datenbestand, der einen bestimmten Ausschnitt - eine Miniwelt - der realen Welt beschreibt -> codierte Informationen
- **Datenbank-Verwaltungssystem (DBMS):**
 - kontrolliert SÄMTLICHE Vorgänge im Datenbanksystem!
 - bietet Dienstfunktionen:
 - zum Entgegennehmen, Speichern, Ändern, Löschen, Wiederauffinden, Auswählen und Bereitstellen von Daten,
 - zur Verwaltung und Wartung

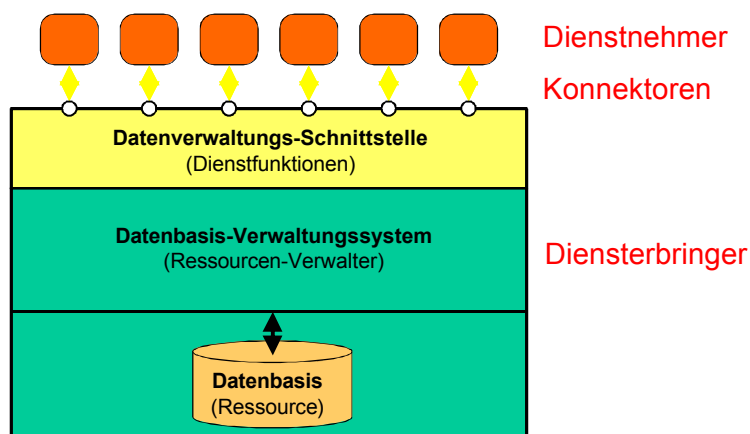


Datenbanken IKA 8/10 - 1

Datenbank-Verwaltungssystem

BBS

16



Datenbanken IKA 8/10 - 1

Aufgaben eines Datenbanksystems (1)



17

- Bereitstellung einer **Datendefinitionssprache** (DDL = Data Definition Language) zur vollständigen Beschreibung von
 - Datenstrukturen
 - Verwendung der definierten Strukturen
 - Beschreibung von Integritätsregeln (Constraints)
- kontinuierliche Führung eines **Datenwörterbuchs** (DD = Data Dictionary, „Metadaten“, gepflegt mit der DDL)
- Bereitstellung einer **Datenmanipulationssprache** (DML = Data Manipulation Language) bzw. Abfragesprache (QL = Query Language) für den interaktiven Zugriff auf die Datenbank-Inhalte
- DCL – Data Control Language für Nutzerberechtigungen

Datenbanken IKA 8/10 - 1

Aufgaben eines Datenbanksystems (2)



18

- Unterstützung der Anwendungsprogrammierung
- Bereitstellung von Routinen zur Erhaltung der logischen und physischen Datenintegrität, insbesondere:
 - Protokollieren von Transaktionen (Transaction logging),
 - Wiederherstellung im Fehlerfall (Recovery).
- Synchronisation des Mehrfachzugriffs
- Zusicherung von Datenschutz und Datensicherheit

Datenbanken IKA 8/10 - 1

Historische Entwicklung (1)



19

Die Urzeit:

- Jede Anwendung speichert ihre eigenen Daten „handgestrickt“
- Für mäßig komplexe Anwendungen verhältnismäßig effizient
- Enormer Programmieraufwand
- Schwer zu wartende Programme (...)
- Jede neue Anfrage muss von einem Programmier-Experten beantwortet werden...

Datenbanken IKA 8/10 - 1

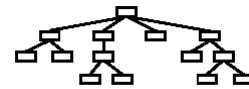
Historische Entwicklung (2)



20

Datenbanken der 1. Generation

- 1970er Jahre: erstmals Trennung zwischen Programmier- und Datenbank-Datenmodell
- Hierarchisches Datenmodell
 - baumstrukturierte Daten
 - siehe: Dateisysteme, LDAP-Verzeichnisdienst
- Netzwerkmodell
 - Verallgemeinertes hierarchisches Modell
- Komplizierte Datenmodelle
- Anfragen müssen von einem Datenbank-Experten beantwortet werden...



Datenbanken IKA 8/10 - 1

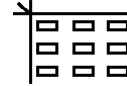
Historische Entwicklung (3)



21

Datenbanken der 2. Generation

- Relationales Datenmodell
 - Theoretische Grundlage: Codd 1970
 - Erstes kommerzielles System: Oracle 1979
 - Verbreitung seit Mitte der 1980er Jahre
 - Vertreter: Oracle, Informix, Postgresql, ...
- Vorteile:
 - Einfach zu begreifendes Datenmodell
 - Mathematische Grundlage
 - Hoher Standardisierungsgrad (SQL-92 / SQL2)
- Erstmals sind verhältnismäßig einfache Ad-Hoc-Anfragen auch durch Nicht-Experten möglich



Datenbanken IKA 8/10 - 1

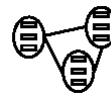
Historische Entwicklung (4)



22

Datenbanken der 3. Generation

- Objektorientierung in zwei Ausprägungen:
 - Rein Objektorientiertes Datenmodell
 - Revolutionärer Ansatz
 - Quasi-Standard: ODMG
 - Vertreter: ObjectStore, O2
 - Objekt-relationales Datenmodell
 - Evolutionärer Ansatz (relationales Modell)
 - Standard: SQL-99
 - Vertreter: Oracle8
- Tendenz: es wird teilweise wieder komplizierter, da Datenmodelle immer ausdruckskräftiger werden



Datenbanken IKA 8/10 - 1

Historische Entwicklung (5)



23

Datenbanken der nächsten Generation:

- Alles ist noch offen...
- Aktuelle Diskussion: XML-Datenbanken
- An der Objektorientierung wird wohl kein Weg vorbeiführen
- Evtl. Aspektorientierung ?
 - Viele auf spezielle Aspekte optimierte Datenmodelle
 - Leistungsfähige DBMS-Werkzeuge zum Verwalten der Modelle

Noch Zukunftsmusik...

Datenbanken IKA 8/10 - 1

Einsatz von DBMS



24

Wie werden Datenbankmanagementsysteme verwendet?

- Betriebliche Anwendungen
- Web-Anwendungen
- ...

Als Teil eines Informationssystems ist die Gesamt-Architektur entscheidend:

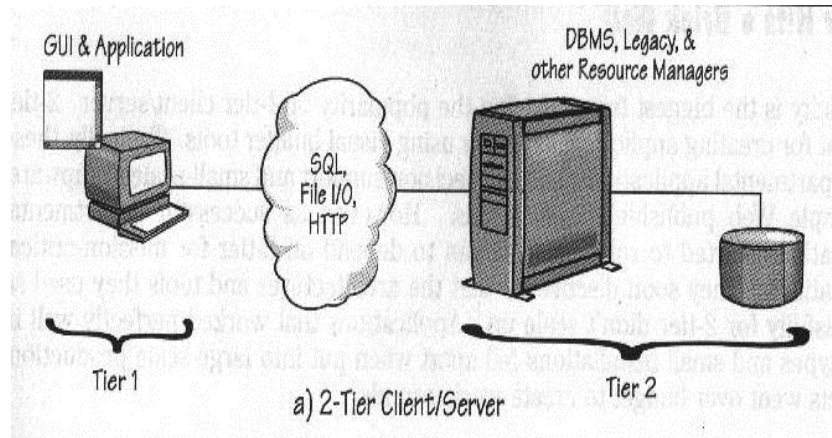
- heute typischerweise Client/Server-Architekturen

Datenbanken IKA 8/10 - 1

DBMS: 2-Tier Client/Server

BBS

25

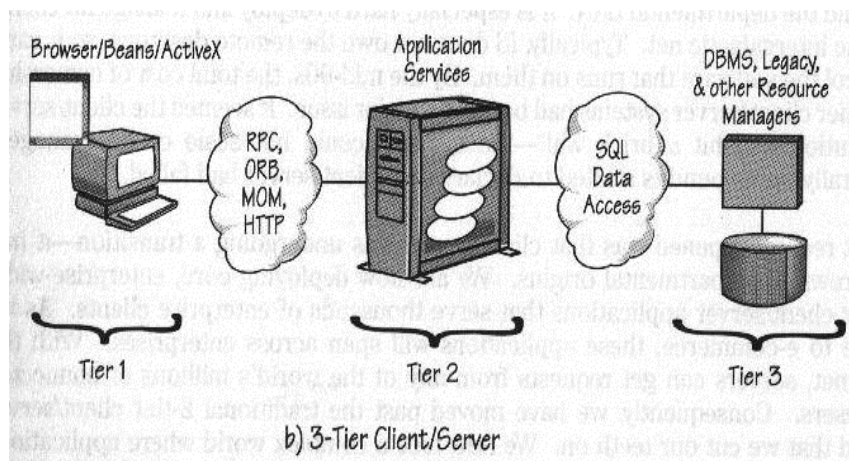


Datenbanken IKA 8/10 - 1

DBMS: 3-Tier Client/Server

BBS

26

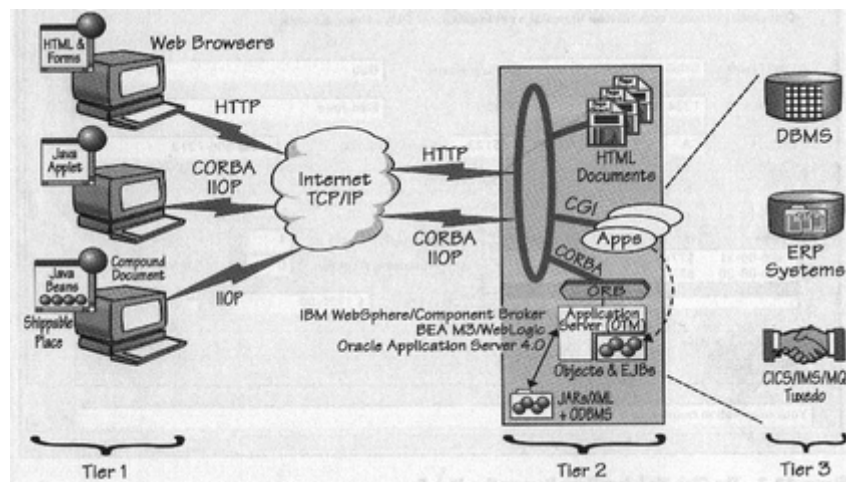


Datenbanken IKA 8/10 - 1

DBMS: "The Big Picture"

BBS

27



Datenbanken IKA 8/10 - 1

Relationale Datenbanksysteme

BBS

28

Alle Informationen werden durch Werte in Tabellen (Relationen) dargestellt, die durch Fremdschlüssel-Beziehungen miteinander verknüpft werden können.

Regeln für das Tabellenschema sind die
5 Normalformen

Regeln für das Verhalten des DB-Systems sind
die **12 Codd-Regeln**

Datenbanken IKA 8/10 - 1

Standard-Datentypen 1



29

• Zahlen- und Wahrheitswerte:

- **Boolean** bzw. **bit**: Boolesche Wahr / Falsch bzw. Ja / Nein - Werte. Dieser Datentyp eignet sich zur Darstellung von zwei einander ausschließenden Werten und kann als Checkbox zum Anhaken oder Pull-down-Liste (Ja/Nein) dargestellt werden.
- **int** bzw. **Integer**: Ganzzahl mit negativen oder positiven Werten. Im Regelfall mit 4 Byte implementiert, damit kann ein Wertebereich von +/- 2^{31} , also -2.147.483.648 bis +2.147.483.648 abgedeckt werden.
- **float** bzw. **real**: Ungefähre Zahlendatentypen, die für numerische Gleitkomma-Daten genutzt werden können. Intern werden diese in der wissenschaftlichen Darstellung $5.79E2$ dargestellt, also als $5.79 * 10^2 = 579$. Diese Datentypen sind ungenau, da nicht alle Werte im zulässigen Zahlbereich genau dargestellt werden. float ist in der Regel doppelt so genau wie real und benötigt 8 statt 4 Byte.

Datenbanken IKA 8/10 - 1

Standard-Datentypen 2



30

• Weitere Zahlenwerte:

- **money**: Währungsdatentyp. Währungen werden mit hinreichend vielen Nachkomma-Stellen gespeichert.
- **decimal[(p[, s])]**, Synonym **numeric**: Dezimal-Datentyp für numerische Datentypen mit einer fest vorgegebenen Genauigkeit und fest definierten Anzahl von Dezimalstellen. p legt die Zahl aller Stellen vor und nach dem Dezimalpunkt fest und kann üblicherweise zwischen 1 und knapp 40 liegen. s fixiert die Nachkommastellen und liegt zwischen 0 und p. s = 0 bedeutet, dass der Datentyp nur Ganzzahlen akzeptiert. decimal(15,5) deklariert einen Datentyp mit fünf Nachkomma- und maximal 10 Vorkomma-Stellen.
- **datetime**: Datums- und Zeitfelder. Beide Informationen werden gemeinsam gespeichert. Wurde nur das Datum angegeben, so wird als Zeit '00:00:00' genommen, bei einer reinen Zeitangabe wird meist als Bezugsdatum der 01.01.1900 verwendet.

Datenbanken IKA 8/10 - 1

Standard-Datentypen 3



31

• Zeichenwerte:

- **char(Länge)**: Zeichenfolge (1-Byte-Zeichen) mit fester Länge (char(5) = fünf Zeichen). Dies erlaubt nur Zeichen aus dem ASCII bzw. den ISO-Codierungen, so dass maximal 256 verschiedene Zeichen gespeichert werden können.
- **nchar(Länge)**: Unicode-Zeichenfolge mit fester Länge, benötigt zwei Byte pro Zeichen.
- **ntext**: Unicode-Zeichenfolge variabler Länge, in der Regel bis zu 2^{30} Zeichen.
- **nvarchar(Länge)**: Unicode-Zeichenfolge variabler Länge, beim MS-QLserver beschränkt auf 4000 Zeichen.
- **text**: Nicht-Unicode-Zeichenfolge variabler Länge, in der Regel bis zu 2^{31} Zeichen.
- **varchar(Länge)**: Zeichenfolge (1-Byte-Zeichen) mit variabler Länge, maximal der durch 'Länge' festgelegte Wert.

Datenbanken IKA 8/10 - 1

Standard-Datentypen 4



32

• Binärdaten:

- **binary(Länge)**: Binärdaten fester Länge. Dieser kann mit analogen Datentypen verwendet werden, um beispielsweise Bilder, Videos, Word- oder PDF-Dokumente abzuspeichern.
- **image**: Binary-Daten variabler Länge, meist mit einer maximalen Länge in der Größenordnung 2^{31} .
- **varbinary(Länge)**: Binary-Daten variabler Länge zwischen 0 und dem durch 'Länge' festgelegten Wert. Beim MS-QLServer existiert eine zusätzliche Obergrenze von 8000 Zeichen.

Die Verwendung der **binary large object (BLOB)** Datentypen **text**, **ntext** und **image** bedeutet, dass in der Zelle nur ein Zeiger gespeichert und der eigentliche Inhalt ausgelagert wird.

Datenbanken IKA 8/10 - 1

Spezielle Datentypen



33

- **int Identity(1, 1)**, eine **Identitätsspalte**: Dies definiert eine Spalte vom Typ Integer, die zusätzlich einen Identitätswert erhält, eine Zahl, die vom DBMS automatisch hochgezählt und deren Eindeutigkeit damit garantiert ist. Solche Spalten sind gute Kandidaten für Primärschlüssel. Die erste Zahl legt den Startwert fest, die zweite Zahl deklariert die Schrittweite.
- **uniqueidentifier**: Ein global eindeutiger Bezeichner (**GUID, Global unique Identifier**). Auto-Werte sind nur innerhalb einer Tabelle eindeutig. Werden global eindeutige Werte benötigt, so können GUIDs genutzt und in unique-identifier-Spalten abgelegt werden.
- **timestamp**: Ein datenbankweit eindeutiger Ausdruck, der bei jeder Änderung der Daten automatisch hochgezählt wird. Wird einer Tabelle eine Spalte vom Typ timestamp hinzugefügt, so kann diese zur Versionskennung der Zeile verwendet werden.

Datenbanken IKA 8/10 - 1

Die 12 Codd-Regeln im Überblick



34

1. **Information Rule** (Darstellung von Informationen):
Alle Informationen werden durch Werte in Tabellen dargestellt.
2. **Guaranteed Access Rule** (Garantierter Zugriff):
Jeder Wert muss über Tabellennamen, Spaltennamen und Wert des Primärschlüssels zugreifbar sein.
3. **Systematic Treatment of Null Values** (Systematische Behandlung von Null-Werten):
Fehlende Werte werden unabhängig vom Datentyp mit NULL dargestellt.
4. **Dynamic On-line Catalog Based on the Relational Model** (Online-Datenkatalog):
Die Datenbank und ihre Inhalte werden in einem Online-Datenkatalog (Data Dictionary) in Form von Tabellen beschrieben.
5. **Comprehensive Data Sublanguage Rule** (Metasprache):
Das relationale System unterstützt Datendefinition, Datenmanipulation und Datenabfrage mit einer einheitlichen Sprache.
6. **View Updating Rule** (Aktualisierung von Sichten):
Sofern theoretisch möglich, müssen Inhalte von Basistabellen auch über Sichten änderbar sein.
7. **High-level Insert, Update and Delete** (Parallele Datenmanipulation):
Innerhalb einer Operation können beliebig viele Tupel bearbeitet werden.
8. **Physical Data Independence** (Physische Datenunabhängigkeit):
Physische Ebene (Speicherung) und konzeptuelles Schema der Datenbank sind unabhängig voneinander.
9. **Logical Data Independence** (Logische Datenunabhängigkeit):
Die Sicht einer Anwendung und das konzeptuelle Schema der Datenbank sind unabhängig voneinander.
10. **Integrity Independence** (Unabhängigkeit der Integrität):
Unumgängliche Regeln (Integritätsbedingungen) müssen im Online-Datenkatalog gespeichert werden können.
11. **Distribution Independence** (Verteilungsunabhängigkeit):
Logische und physische Datenunabhängigkeit müssen ebenso für verteilte Datenbanken gelten.
12. **Nonsubversion Rule** (Kein Unterlaufen der Integrität):
Integritätsbedingungen dürfen auf keinen Fall umgangen werden, auch nicht mit Hilfe von niederen Programmiersprachen.

Datenbanken IKA 8/10 - 1

Die 12 Codd-Regeln (Grundregel)



35

Grundregel (Regel 0):

Ein relationales DBMS muss in der Lage sein, Datenbanken vollständig über seine relationalen Fähigkeiten zu verwalten

Datenbanken IKA 8/10 - 1

Die 12 Codd-Regeln (1-3)



36

1. Darstellung von Informationen:

Alle Informationen in einer relationalen Datenbank (einschließlich Namen von Tabellen und Spalten) sind logisch als Werte in Tabellen darzustellen

2. Zugriff auf Daten:

Jeder Wert einer relationalen Datenbank muss durch eine Kombination von Tabellename, Primärschlüssel und Spaltenname auffindbar sein

3. Systematische Behandlung von Nullwerten:

Das DBMS behandelt Nullwerte durchgängig gleich als unbekannte oder fehlende Daten und unterscheidet diese von Standardwerten

Datenbanken IKA 8/10 - 1

Die 12 Codd-Regeln (4-6)



37

4. Struktur einer Datenbank:

Die Datenbank und ihre Inhalte werden in einem sog. Systemkatalog auf derselben logischen Ebene wie die Daten selbst - also in Tabellen - beschrieben

5. Abfragesprache:

Mindestens eine Abfragesprache mit vollständigem Befehlssatz zur Datendefinition, Sichten-Definition, Autorisierung, Manipulation (interaktiv und über Programme), Integritätsregeln und Transaktionen

6. Aktualisieren von Sichten:

Alle Sichten, die theoretisch aktualisiert werden können, lassen sich auch vom System aktualisieren

Datenbanken IKA 8/10 - 1

Die 12 Codd-Regeln (7-9)



38

7. Abfrage und Bearbeiten ganzer Tabellen:

Das DBMS unterstützt Operationen wie Abfragen, Einfügen, Aktualisieren und Löschen in Form ganzer Tabellen

8. Physikalische Datenunabhängigkeit:

Der logische Zugriff auf die Daten muss unabhängig von physikalischen Zugriffsmethoden oder Speicherstrukturen sein.

9. Logische Datenunabhängigkeit:

Änderungen der Tabellenstrukturen dürfen keinen Einfluss auf die Logik der Anwendungen haben

Datenbanken IKA 8/10 - 1

Die 12 Codd-Regeln (10-12)



39

10. Unabhängigkeit der Integrität:

Integritätsregeln müssen definierbar sein, abgespeichert im Systemkatalog und nicht umgehbar

11. Verteilungsunabhängigkeit:

Logischer Zugriff auf Daten darf sich bei Verteilung der Daten nicht ändern

12. Kein Unterlaufen der Abfragesprache:

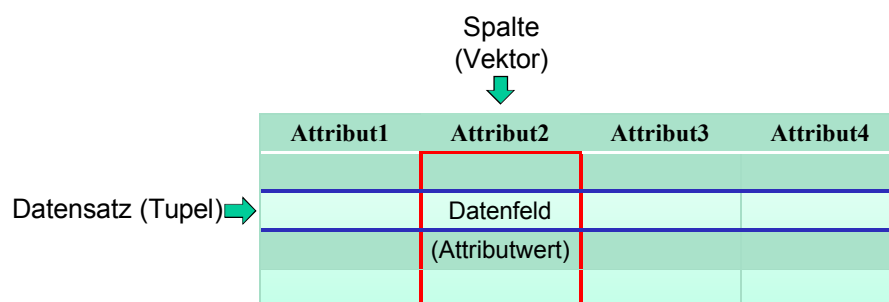
Integritätsregeln dürfen sich nicht durch Low-Level-Sprachen umgehen lassen

Datenbanken IKA 8/10 - 1

Tabellen (Relationen)



40



Jeder Datensatz repräsentiert eine sogenannte Entität:

Eine **Entität** ist ein eindeutig identifizierbares Objekt oder ein eindeutig identifizierbarer Sachverhalt der realen Welt.

Die Entität wird durch ihre Attribute (Eigenschaften) bestimmt.

Ein **Entitätstyp** ist die abstrakte Beschreibung einer Menge von Entitäten mit den selben Attributen.

Datenbanken IKA 8/10 - 1

Schlüssel



41

- Allgemein:
eine Menge von Attributen, die eine Entität (Tupel, Datensatz) eindeutig identifiziert
- Schlüsselkandidat:
eine Mindestmenge an Attributen zur Identifikation
- Primärschlüssel:
ein (möglichst kleiner) ausgewählter Schlüsselkandidat:
 - einfach (besteht aus nur 1 Attribut)
 - zusammengesetzt (besteht aus mehreren Attributen)
 - künstlich (zusätzliches Attribut als einfacher Ersatzschlüssel)
- Fremdschlüssel:
Attribut einer Relation, das auf den Primärschlüssel einer anderen (oder der gleichen) Relation verweist

Datenbanken IKA 8/10 - 1

Von der Realität zur Datenbank



42

- Wie entsteht eine Datenbank?
- Nötige Einzelschritte:
 - logische Datenmodellierung,
 - physische Datenmodellierung,
 - Aufbau der Datenbank
 - Betrieb (Administration, Konfiguration) der Datenbank

Datenbanken IKA 8/10 - 1

1) logische Datenmodellierung



43

Ausgangspunkt ist meist ein Problem aus der Realität

- den relevanten Teil gilt es zu modellieren, als Ergebnis erhält man eine sogenannte Miniwelt

Die Datenmodellierung erfolgt z.B. mittels

- ER-Modell (Entity-Relationship)
- UML (Unified Modeling Language)

Ergebnis: ein Datenmodell der Anwendungsdomäne

Datenbanken IKA 8/10 - 1

2) physische Datenmodellierung



44

Das logische Datenmodell aus Schritt 1) muss dann in ein geeignetes physisches Datenmodell umgesetzt werden

- Hier betrachten wir dafür vor allem relationale Datenmodelle und Datenbanksysteme
- Ergebnis ist ein - auf einem konkreten Datenbanksystem - einsetzbares Modell (z.B. in Form von SQL)

Datenbanken IKA 8/10 - 1

3) Aufbau und Einsatz



45

...die hässlichen Details, von denen man in den Lehrbüchern meist verschont wird:

- Auswahl, Kauf, Installation, Konfiguration eines DBMS
- Anlegen der Datenbank, Einspielen der Daten, ...
- Administration
- Leistungsoptimierung
- Sicherheitsaspekte
- Anwendungsentwicklung
- Backup, Replikation, Clustering, Failover, Recovery, ...