



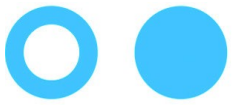
Teil I

Von EDV-Anwendungen und Ihren Anforderungen

Robert Hartmann (SoSe 2024)

basierend auf Folien von
Prof. Dr. Harm Knolle

Fachbereich Informatik
Hochschule Bonn-Rhein-Sieg



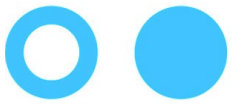
- Einführung -

Inhalt

- 0 - Vorbemerkungen
- Teil I - Von EDV-Anwendungen und Ihren Anforderungen
 - 1 - Einführung
- Teil II - Von Daten und ihren Modellen
 - 2 - Prozess des Datenbankentwurfs
 - 3 - Semantische Datenmodelle
 - 4 - Logische Datenmodelle
 - 5 - Datenbankmodelle
 - 6 - Datenanfrage und Datenänderung
- Teil III - Von Datenbanken und ihren Systemen
 - 7 - Datenbanksysteme
 - 8 - Speicherstrukturen

Überblick

- ♦ Datenbankgestützte Anwendungen
- ♦ Berufsfelder
- ♦ Historie



- Kapitel 1 - Einführung -

Ziel

- ♦ Wie sind Datenbanksysteme EDV-technisch einzuordnen ?
- ♦ Wo liegen ihre Einsatzbereiche?
- ♦ Was sind ihre Grundmerkmale?
- ♦ Was sind ihre Grundprinzipien?
- ♦ Wie haben sie sich entwickelt?

Hilfsmittel

- ♦ Datenbanken
- ♦ Datenbankmanagementsysteme
- ♦ Datenbanksysteme
- ♦ Datenstrukturdeklarationen

Inhalt

- ♦ Datenbankgestützte Anwendungen
- ♦ Berufsfelder
- ♦ Historie

Literatur

- ♦ KeEi15, Kapitel 1: „Einleitung und Übersicht“
- ♦ SSH18, Kapitel 1: „Grundlegende Konzepte“
- ♦ Ku15, Kapitel 1: „Datenbanken: Grundlagen und Überblick“

- Datenbankgestützte Anwendungen -

Inhalt

- ♦ **Datenbankgestützte Anwendungen**
- ♦ Berufsfelder
- ♦ Historie

Überblick

- ♦ Klassifikation von Daten
- ♦ Datenbanksystem, Datenbankmanagementsystem, Datenbank
- ♦ Einsatz von Datenbanksystemen
- ♦ Merkmale von Datenbanksystemen
- ♦ Definition einer Datenbank
- ♦ Datenbanken kritisch betrachtet
- ♦ EDV-technische Einordnung von Datenbanksystemen

- Klassifikation von Daten (I) -

Daten allgemein

- ♦ physische Repräsentationen, denen eine bestimmte Bedeutung bzw. Information unterstellt werden kann

Unstrukturierte Daten

- ♦ interne Struktur der Daten kennt nur der Anwender bzw. das Anwendungssystem, z.B.
 - Texte
 - Bilder
 - Audio
- ♦ Interpretation der Daten nur durch Anwender bzw. durch das Anwendungssystem

Semi-strukturierte Daten

- ♦ Strukturbeschreibung für Speicherung und Interpretation (Datenaustausch) ist in den Daten enthalten
- ♦ kann aber für die einzelnen Datensätze unterschiedlich sein
- ♦ Beispiel JSON

```
{ „Studenten“ : [  
  { „Vorname“ : „Hans“, „Name“ : „Müller“ },  
  { „Name“ : „Schulze“, „GDat“ : „22.06.1996“ },  
  { „Vorname“ : „Susanne“, „GDat“ : „18.03.1995“ }  
] }
```

- ♦ Beispiel XML

```
<Studenten>  
  <Student> <Vorname> Hans </Vorname>  
    <Name> Müller </Name> </Student>  
  <Student> <Name> Schulze </Name>  
    <GDat> 22.06.1961 </GDat> </Student>  
  <Student> <Vorname> Susanne </Vorname>  
    <GDat> 18.03.1995 </GDat> </Student>  
</Studenten>
```

- Klassifikation von Daten (II) -

Strukturierte Daten

- ♦ Daten haben einen festen Satzaufbau
- ♦ Datensätze sind streng typisiert
- ♦ typgleiche Datensätze haben immer denselben Satzaufbau
- ♦ Beispiel CSV-Datei „Studenten.csv“

```
Vorname,Name,Gdat  
Hans,Müller,  
,Schulze,22.06.1998  
Susanne,,18.03.1995
```

- ♦ Beispiel Tabelle „Studenten“

Vorname	Name	GDat
Hans	Müller	leer
leer	Schulze	22.06.1996
Susanne	leer	18.03.1995

Gegenüberstellung

- ♦ erhebliche Unterschiede
- ♦ je höher der Grad der Strukturierung,
 - desto weniger Strukturbeschreibung muss gespeichert werden
 - desto effizienter kann die Suche nach bestimmten Daten unterstützt werden
 - desto mehr kann ein Datenbanksystem leisten
- ♦ je geringer der Grad der Strukturierung,
 - desto flexibler kann der Satzaufbau erfolgen
 - desto variabler lassen sich die Daten speichern
 - desto aufwendiger ist die Suche nach bestimmten Daten
 - programmiertechnisch aufwändiger
- ♦ diese Lehrveranstaltung: Konzentration auf hohen Grad der Strukturierung

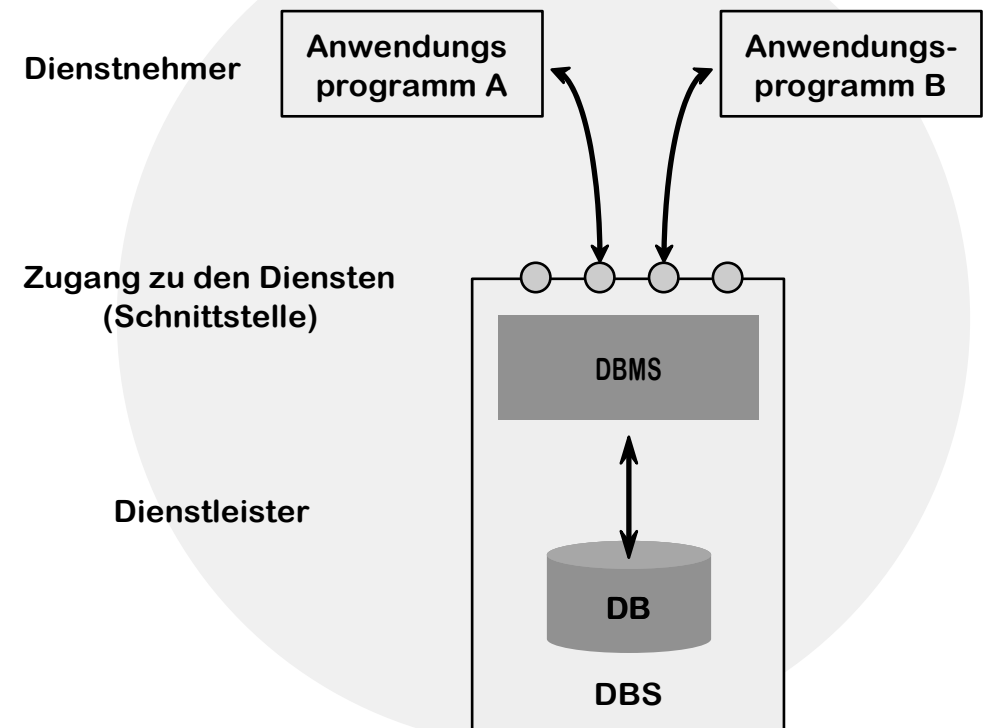
- Datenbanksystem, Datenbankmanagementsystem, Datenbank -

Verwalten von Datenbeständen

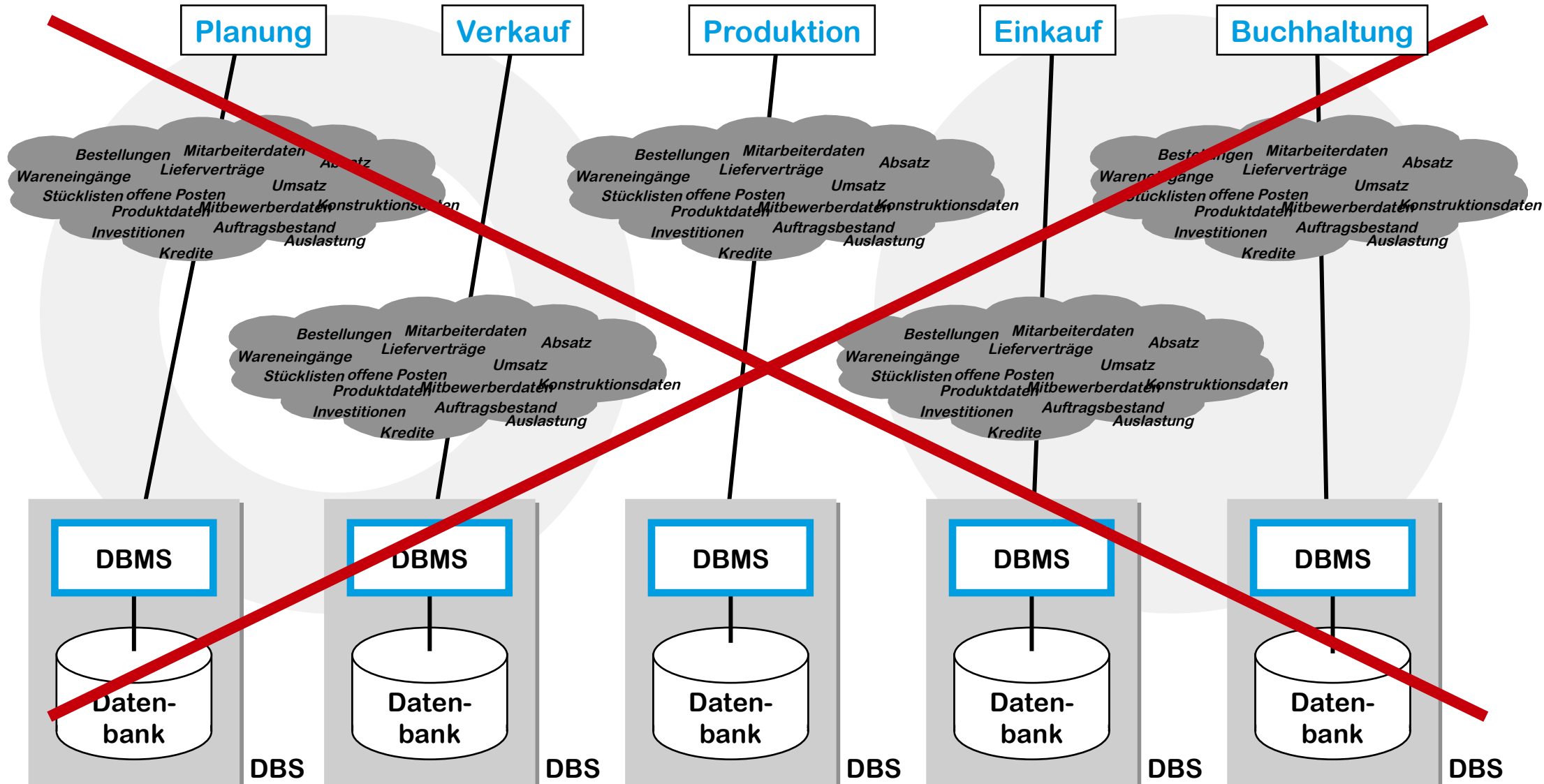
- ♦ Überwindung zeitlicher Distanzen
- ♦ Dienstfunktionen
 - Speichern
 - Erstellen, Modifizieren, Löschen
 - Lesen
 - Wiederauffinden, Auswählen, Bereitstellen

Datenbanksystem (DBS)

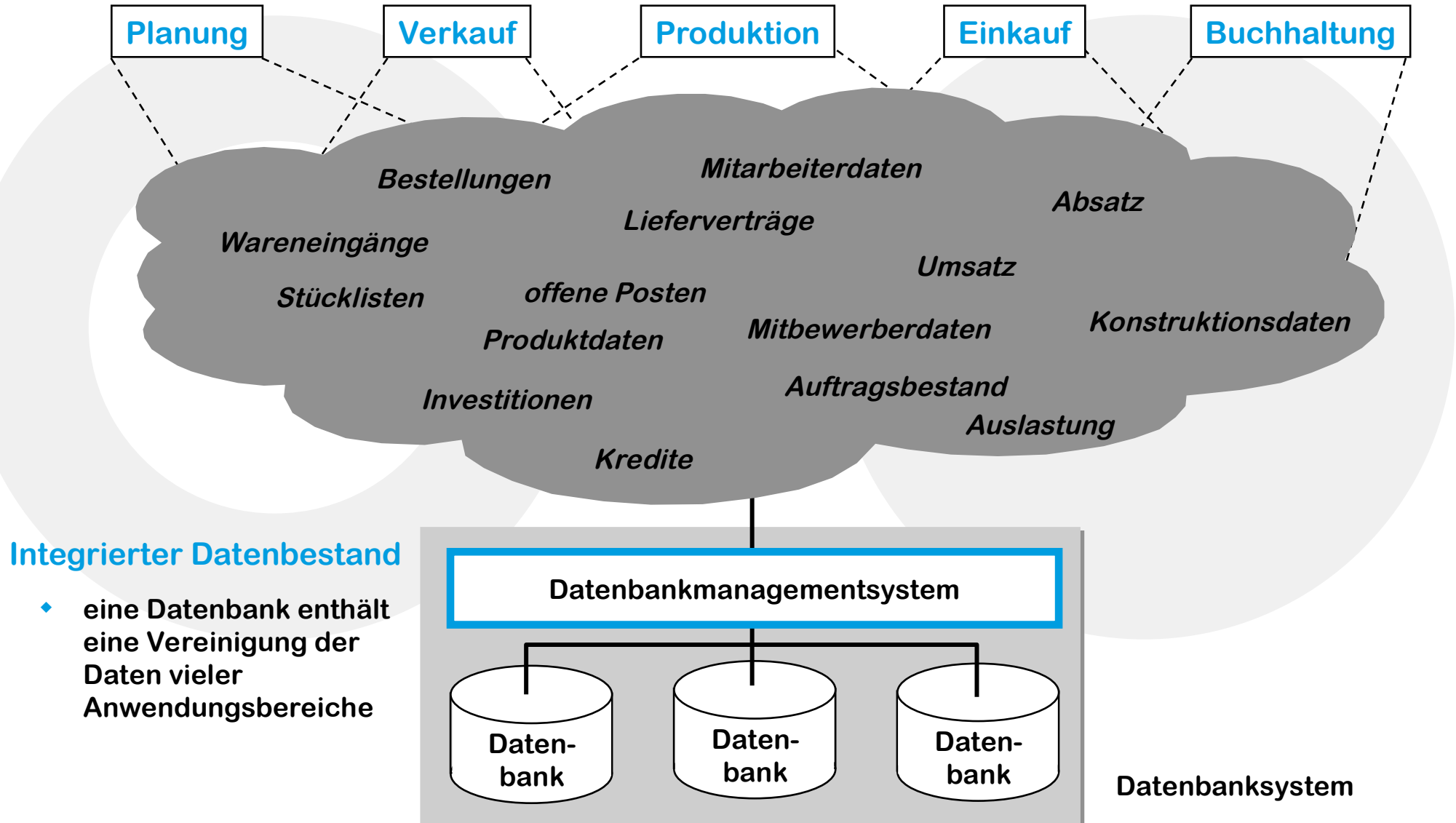
- ♦ System mit den o.g. Dienstfunktionen
 - Datenbank (DB)
 - Menge der gespeicherten Daten
 - Datenbankmanagementsystem (DBMS)
 - System zur Erfüllung der o.g. Dienstfunktionen mit der Datenbank



- Einsatz von Datenbanksystemen (I) -



- Einsatz von Datenbanksystemen (II) -



- Merkmale von Datenbanksystemen -

Integrierter Datenbestand

- ♦ eine Datenbank enthält eine Vereinigung der Daten vieler Anwendungsbereiche

Modellinhärente Konsistenz

- ♦ eine Datenbank garantiert die Einhaltung von Gesetzmäßigkeiten sowohl einzelner Daten als auch zwischen den Daten untereinander

Redundanzfreiheit des Datenbestands

- ♦ gleiche Daten unterschiedlicher Anwendungen werden nicht doppelt gespeichert

Datenunabhängigkeit

- ♦ der Zugriff auf die Daten erfolgt ohne Kenntnis der physischen Speicherstrukturen und ohne Kenntnis der physischen Speichermedien (**physische Datenunabhängigkeit**)
- ♦ die Anordnung der Speicherstrukturen kann verändert werden, ohne dass dieses Auswirkungen auf die Anwendungsprogramme hat (**logische Datenunabhängigkeit**)

Deskriptive Abfragesprache

- ♦ der Datenzugriff erfolgt lediglich über eine Beschreibung der gewünschten Daten, nicht aber auch noch, wie man sie bekommt

Anwendungsspezifische Sichten

- ♦ gleiche Daten werden für unterschiedliche Anwendungen in unterschiedlichen semantischen Zusammenhängen präsentiert

Datenschutz / Datensicherheit

- ♦ bestimmte Daten sind für bestimmte Benutzer oder Benutzergruppen geschützt
- ♦ nach Verlust oder Systemzusammenbrüchen immer Wiederherstellbarkeit der Datenbank

Transaktionen (semantische Integrität)

- ♦ Modifikationen bringen die Datenbank jeweils von einem gültigen Zustand in einen anderen gültigen Zustand

- Definition einer Datenbank -

Eine Datenbank ist das Modell eines Modells

- ♦ nicht das Modell der Realität
 - Modell davon, wie Anwendungen welchen Ausschnitt der Realität sehen
 - der Detaillierungsgrad der Datenbank ist abhängig davon, wie detailliert die Anwendungen ihren Ausschnitt der Realität sehen wollen
- ♦ das Modell der Datenbank ist dynamisch
 - Veränderungen am Modell werden von sogenannten Transaktionen vorgenommen (Datenmodifikation)
 - Transaktionen ermöglichen aber auch Einblicke auf den aktuellen Zustand des Modells (Datenabfrage)

Eine Datenbank beschreibt sich selbst

- ♦ zusätzlich zu den Daten der Anwendungen werden Informationen über diese Daten verwaltet (Meta Daten)
 - Anwendungen benötigen keine Strukturdeklarationen (**physische Datenunabhängigkeit**)
 - Änderung der Datenstrukturen über Änderung der Meta Daten (**logische Datenunabhängigkeit**)

Eine Datenbank ist eine integrierte Sammlung unterschiedlicher Speicherstrukturen

- ♦ Benutzerdaten
- ♦ Meta Daten
- ♦ Indexe (Optimierung von Zugriffen)
- ♦ Anwendungslogik



- Datenbanken kritisch betrachtet -

Potentielle Nachteile von Datenbanken

- ♦ oft sehr kostspielig in der Anschaffung, Wartung und in der Administration
- ♦ oft "zu" umfangreiche Dokumentation (hohe Einarbeitungszeit)
- ♦ oft sehr großer Bedarf an primären und sekundären Speichermedien
- ♦ in der Regel nicht so schnell, wie individuell zugeschnittenes und optimiertes Programm
- ♦ oft starke Abhängigkeiten vom Hersteller des Datenbanksystems

Vielleicht besser überhaupt keine Datenbank ...

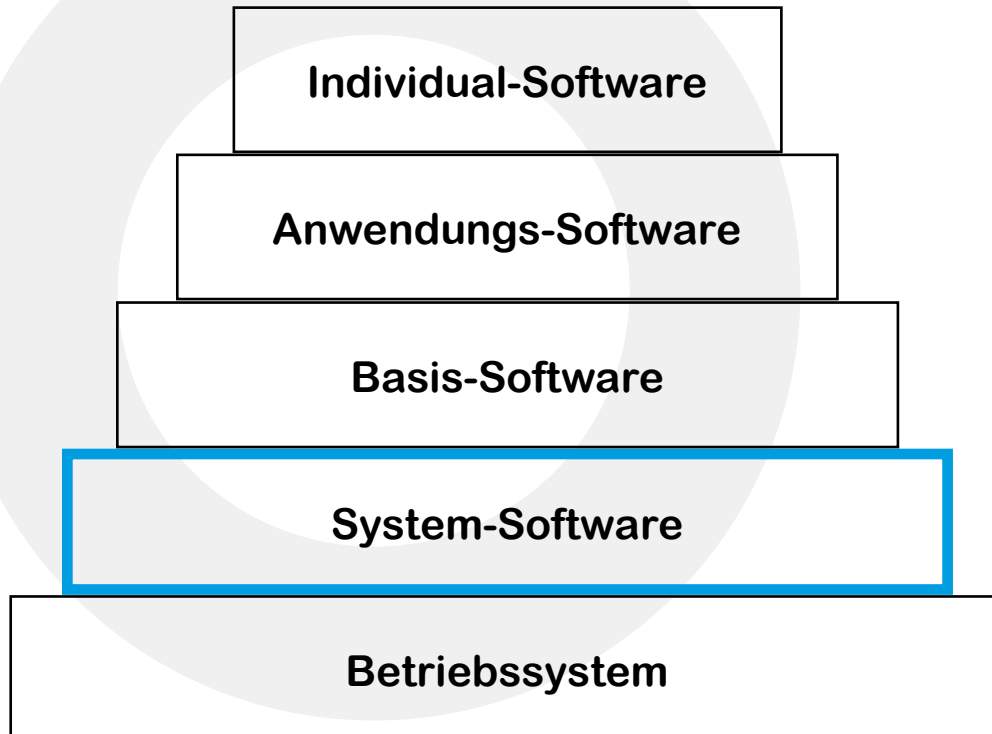
- ♦ wenn Datenbestand ...
 - sehr klein ist
 - einfach oder semi-strukturiert ist
 - nur von einer Anwendung benötigt wird
- ♦ wenn keine Erweiterungen der Anwendungen auf den Daten geplant sind
- ♦ wenn kein Mehrbenutzerbetrieb erforderlich ist
- ♦ wenn kein Echtzeitbetrieb vorliegt

... oder vielleicht sogenannte NoSQL-Datenbanken

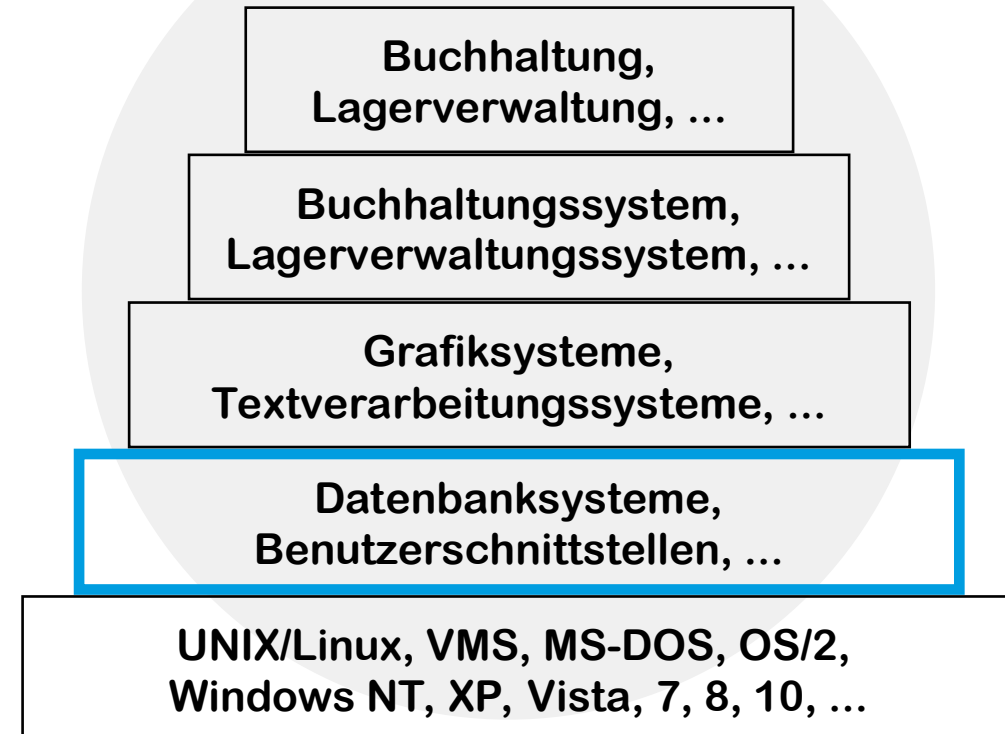
- ♦ bei einfach oder semi-strukturierten Daten
- ♦ bei speziellen Anwendungsfällen
 - spezielle Web 2.0 Datenbanken
 - Big Data

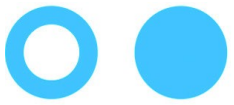
- EDV-techn. Einordnung von Datenbanksystemen -

Schichtenarchitektur



Beispiele





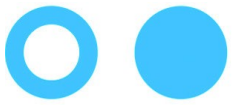
- Berufsfelder -

Inhalt

- ♦ Datenbankgestützte Anwendungen
- ♦ **Berufsfelder**
- ♦ Historie

Eine Auswahl von Berufen

- ♦ **Data Engineer / Dateningenieur :**
Datenaufbereitung „des Wissens der Quelle“.
- ♦ **Data Scientist / Datenwissenschaftler:**
Schafft neues Wissen aus vorhandenen Daten.
- ♦ **Database Engineer / Datenbankingenieur:**
Entwickelt und designet die Datenbank
- ♦ **Database Administrator / Datenbankadministrator:**
Ist für den reibungslosen technischen Betrieb zuständig.



- Historie -

Inhalt

- ♦ Datenbankgestützte Anwendungen
- ♦ Berufsfelder
- ♦ **Historie**

Überblick

- ♦ In einer Welt vor unserer Zeit ...
- ♦ Anfang der 60er Jahre - Dateisysteme
- ♦ Ende der 60er Jahre - Dateiverwaltungssysteme
- ♦ Ab 70er Jahre - Datenbanksysteme

- In einer Welt vor unserer Zeit ... -

Entwicklung datengestützter Anwendungen

- ♦ lange Entwicklungszeiten für Anwenderprogramme
- ♦ geringe Funktionalität durch individuelle Datenhaltung
- ♦ hohe Fehleranfälligkeit
- ♦ keine stabile Bezugsbasis für kooperatives Arbeiten

Aspekte der Integration

- ♦ keine Integration der Daten
 - keine gemeinsame, anwendungsunabhängige Darstellung (kein logisches Schema)
 - Redundanz
- ♦ keine Integration der Anwendungen
 - kein kontrolliert paralleler Zugriff
 - keine Änderung von gemeinsamen Datenbeständen

Datenschutz und Datensicherheit

- ♦ individuelle Lösungen
- ♦ aufwendiges Wiederherstellen nach einem Fehlerfall (wenn überhaupt ...)

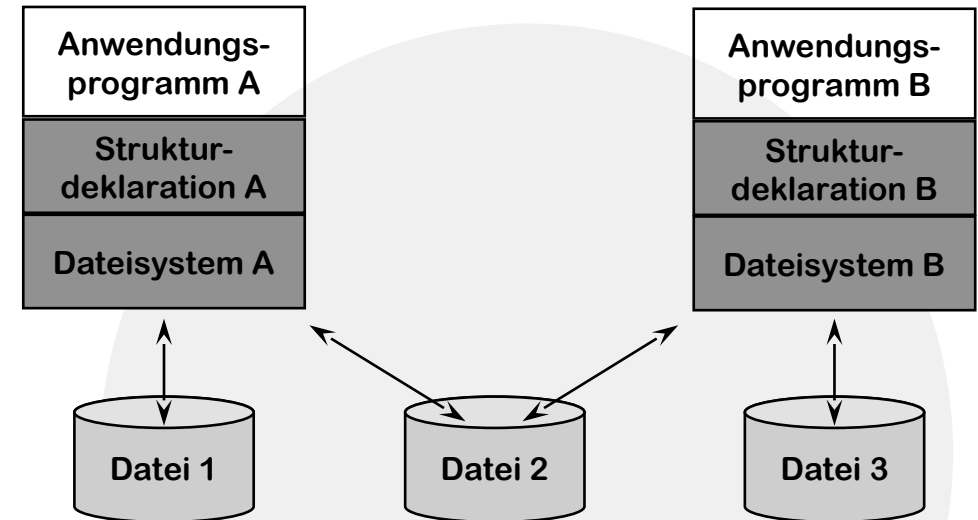
- Anfang der 60er Jahre - Dateisysteme (I) -

Idee

- ♦ Trennung von Anwendungslogik und Datenverwaltung
- ♦ anwendungsspezifische Auslagerung von dauerhaften Programmdaten auf ein sekundäres Speichermedium
- ♦ Zugriff von unterschiedlichen Anwendungen auf die selben Daten

Merkmale

- ♦ Dateisystem
 - Teil der Anwendungssoftware der Speichern und Zugreifen von Daten auf Sekundärspeichern realisiert
- Datei
 - Ansammlung von Sätzen gleicher oder unterschiedlicher Struktur in einer Verwaltungseinheit

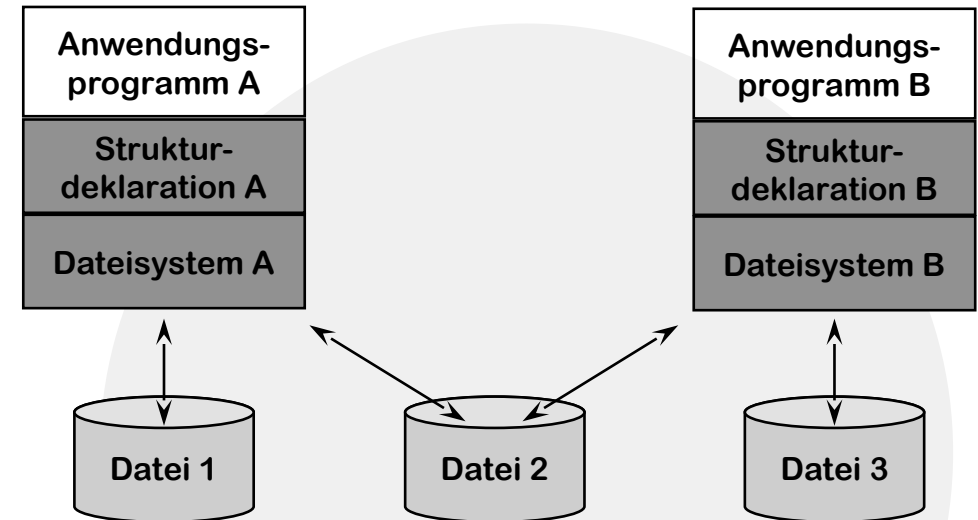


- Satz
 - Zugriffs und Verarbeitungseinheit für das Dateisystem
 - Struktur des Satzes wird im Anwendungsprogramm festgelegt
- ♦ Strukturdeklaration
 - anwendungsspezifische Festlegung
 - Bestimmung der Felder eines Satzes und deren Reihenfolge

- Anfang der 60er Jahre - Dateisysteme (II) -

Ungelöste Probleme bei anwendungs-spezifischer Datenspeicherung

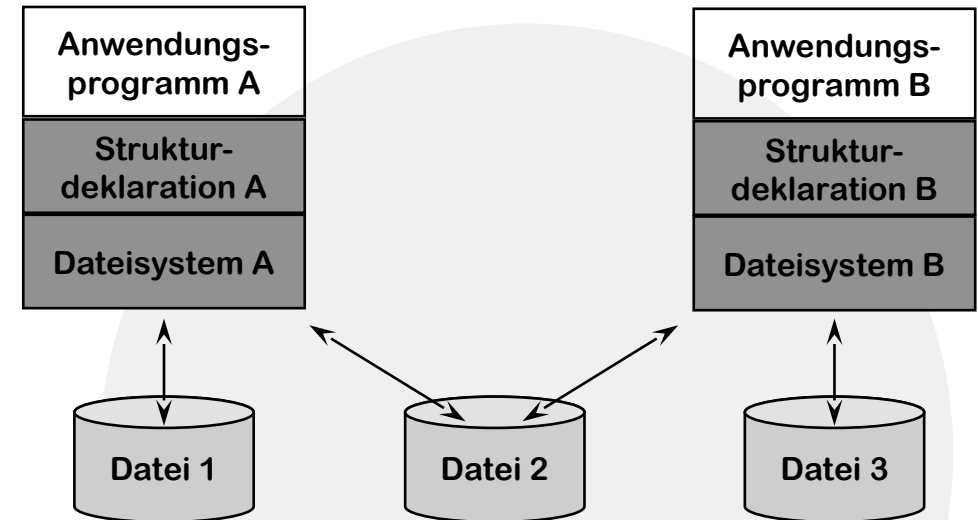
- ◆ fehlende Geräteunabhängigkeit
 - Abhängigkeit vom Speichermedium
- ◆ fehlende Trennung von Anwendung und Daten (keine Datenunabhängigkeit)
 - Strukturdeklaration ist Teil des Anwendungsprogramms
 - Änderungen der Satzstruktur haben Auswirkungen auf alle Anwendungen, die mit der betreffenden Datei arbeiten
- ◆ fehlende Flexibilität bei . . .
 - Auswertung des gesamten Datenbestands
 - Generierung neuer Anwendungen
 - Anpassung an geänderte bzw. neue Problemstellungen



- Anfang der 60er Jahre - Dateisysteme (III) -

Ungelöste Probleme bei anwendungs-spezifischer Datenspeicherung (Forts.)

- ♦ **Redundanz**
 - durch anwendungsspezifische Strukturdeklaration nur geringe Wiederverwendbarkeit der Daten
 - gleiche Daten in verschiedenen Dateien (Redundanz) führen zur Speicherverschwendung
 - Redundanzen sind nur schwer kontrollierbar und sind häufig die Ursache für Inkonsistenzen



- ♦ **Inkonsistenz**
 - logische Übereinstimmung der gleichen Daten (Konsistenz) in unterschiedlichen Dateien ist schwer kontrollierbar
 - Datenänderungen müssen stets alle gleichen Daten in den verschiedenen Dateien umfassen
 - Während der Änderung können andere Anwendungen noch auf alte Daten operieren

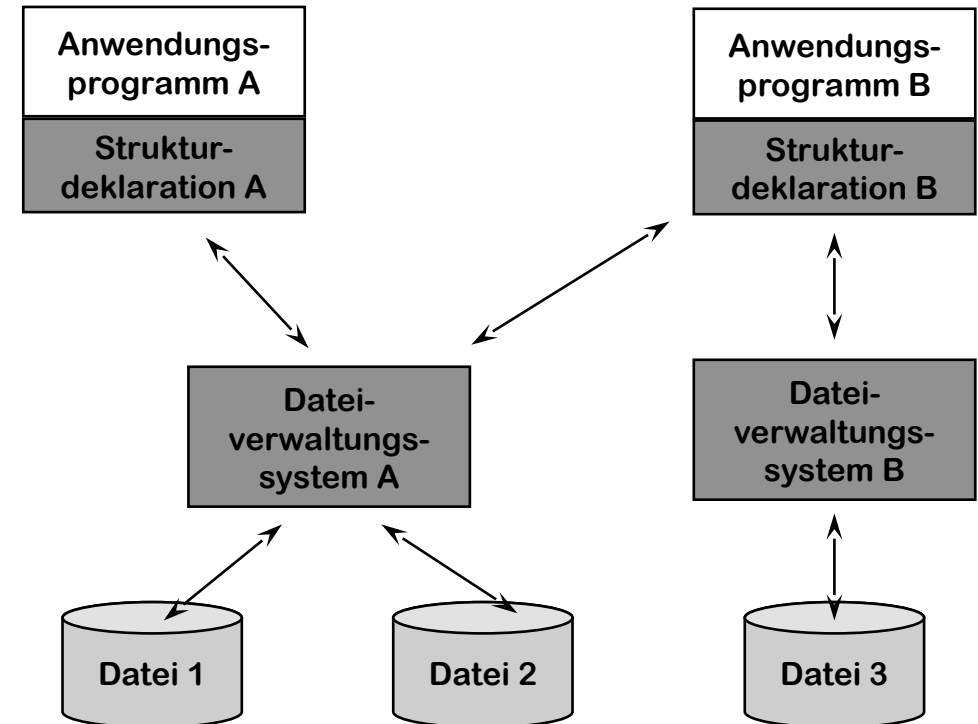
- Ende der 60er Jahre - Dateiverwaltungssysteme -

Idee

- ◆ zusätzliche Trennung von Hard- und Software
 - Isolierung der rechnertypischen Hardware von der Anwendersoftware
 - Änderung der Rechnerkonfiguration ohne Auswirkung auf Anwendung
 - Dienstprogramme z.B. zum Sortieren

Beispiele

- ◆ Sequential Access Method (SAM):
 - sequentielle Dateizugriffe
- ◆ Index Sequential Access Method (ISAM):
 - indexsequentielle Dateizugriffe



Weiterhin ungelöste Probleme

- ◆ fehlende Datenunabhängigkeit
 - Anwendungsprogramm muss wissen, wie die Daten physisch strukturiert sind
- ◆ fehlende Flexibilität
- ◆ Inkonsistenz

- Ab 70er Jahre - Datenbanksysteme -

Idee

- ♦ zusätzlich vollständige Trennung von Datenstrukturierung und Datenverwaltung vom Anwendungsprogramm
 - Datenunabhängigkeit
 - Redundanzfreiheit
 - konsistente Datenhaltung
 - anwendungsneutrale Datendeklaration über ein DBMS
 - Datenzugriff nur über ein DBMS

Merkmale

- ♦ DB (Datenbank): Gesamtheit aller (Anwendungs-) Daten
- ♦ DBMS (Datenbankmanagementsystem): Gesamtheit aller Module, die die Verwaltung einer DB übernehmen
- ♦ DBS (Datenbanksystem): DBMS mit DB

