

1 Datenbanken

Dieses Kapitel wird sich mit dem Thema **Datenbanken** beschäftigen. Bevor wir uns aber mit der Materie auseinandersetzen ein paar interessante Fakten:

Definition aus dem „Duden Informatik“ System zur Beschreibung, Speicherung und Wiedergewinnung von umfangreichen Datenmengen, die von mehreren Anwendungsprogrammen genutzt werden. *ff.*

Historie der Datenspeicherung

- 50er Jahre Dateisysteme auf Band
- 60er Jahre Dateisysteme auf Platte
 - Dateiverwaltungssysteme (z.B. ISAM)
- 70er Jahre Erste Datenbanksysteme (DBMS)
 - hierarchisches DB-Modell
 - Netzwerkmodell
- 80er Jahre Relationale DBS (RDBS)
- 90er Jahre Postrelationale DBS
 - Objekt-Relationale Systeme (ORDBS)
 - Objektorientierte DBS (OODBS)

Ziel der Entwicklung von DBMS

Bis dato gab es zwei Arten der Datenspeicherung: Dateisysteme und Dateiverwaltungssysteme.

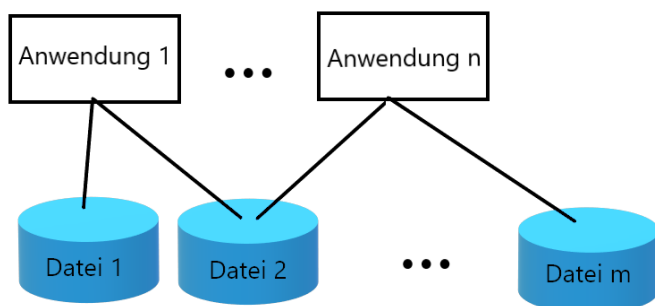


Abb. 1: Dateisysteme

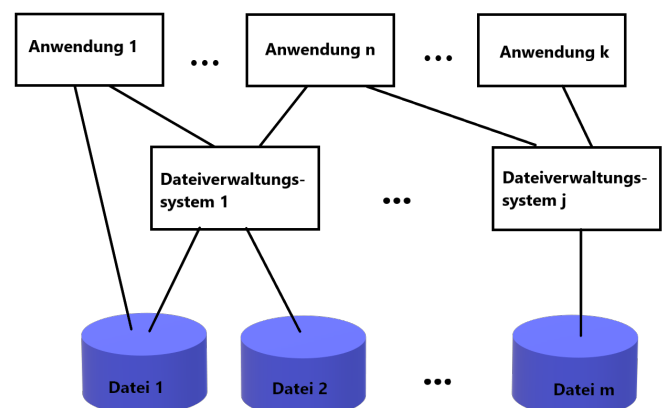


Abb. 1: Dateiverwaltungssysteme

Probleme beim Einsatz von Dateisysteme:

- **Datenredundanz**

Die gleichen Daten werden von mehreren verschiedenen Dateien gespeichert, z.B. Adressen von Kunden werden von mehreren Anwendungen wie Auftragsabwicklung, Buchhaltung usw. verwaltet.

Problem: Speicherplatzverschwendung, Inkonsistenzen

- **Schlechte Effizienz**

- **Problem bei Mehrbenutzerbetrieb**

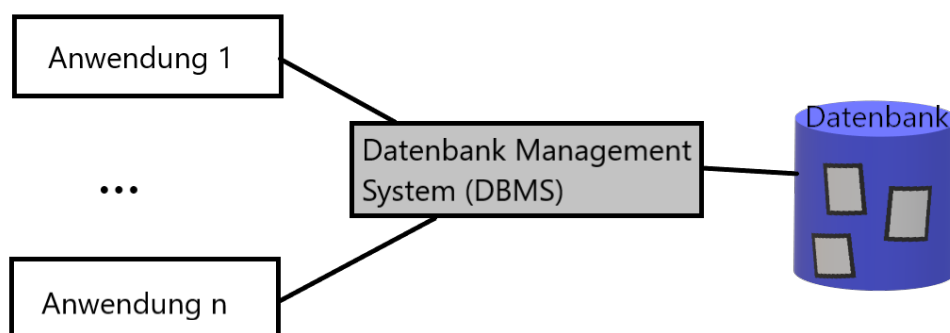
- **Keine Datenunabhängigkeit**

Anwendungsprogrammierer und Anwender müssen die interne Darstellung der Daten und ihre Organisation (Speicherort etc.) kennen.

- Zugriffskontrolle und Datensicherheit sind problematisch

Bei der Entwicklung von Datenbank-Management-Systemen (DBMS) spielte die **Datenintegration** eine große Rolle. Das bedeutet:

- Alle System- und Anwendungsprogramme arbeiten auf denselben Daten, die von einer zentralen Datenkomponente ("Datenbank") verwaltet werden.
- Dadurch kann nicht nur das Problem der Datenredundanz beseitigt werden, sondern auch andere Probleme können gelöst werden
 - Zugriff auf die Daten ohne Kenntnis der konkreten internen Darstellen über standardisierte Zugriffsmechanismen (z.B. Abfragesprachen)
 - „Geordneter“ Mehrbenutzerbetrieb (Transaktionskonzept)
 - Zugriffskontrolle und Datensicherheit (z.B. durch zentrale Datensicherungsmechanismen)
 - Gesteigerte Effizienz durch interne Optimierungsmöglichkeiten des DBMS



Anforderungen an ein DBMS

Für einen zuverlässigen und effizienten Einsatz von DBMS hat *Edgar F. Codd* neun Anforderungen definiert:

- **Integration** einheitliche, nicht-redundante Verwaltung aller Anwendungsdaten
- **Operationen** Operationen wie Speicher, Ändern, Löschen und Suchen auf den Daten müssen möglich sein
- **Katalog** dieser ermöglicht Zugriffe auf die Datenbeschreibung (*Data Dictionary*)
- **Benutzersichten** Für unterschiedliche Anwendungen werden unterschiedliche Sichten auf den Datenbestand ermöglicht
- **Konsistenzüberwachung** Integritätssicherung: Gewährleistung der Korrektheit des Datenbestandes und der korrekten Ausführung von Änderungen
- **Zugriffskontrolle** Ausschluss von unautorisierten Zugriffen auf die gespeicherten Daten
- **Transaktionen** Zusammenfassung von Datenbankänderungen, die als Ganzes ausgeführt werden müssen und deren Effekt bei permanent in der DB gespeichert werden soll
- **Synchronisation** Regelung des gleichzeitigen Zugriffs mehrerer Benutzer
- **Datensicherung** Ermöglichen der Wiederherstellung des Datenbestandes z.B. nach Systemfehlern

Bei der Architektur eines DBMS gibt es drei wichtige Merkmale, die zu beachten sind:

- **Datenunabhängigkeit**
Unabhängigkeit der Daten von den darauf arbeitenden Programmen
- **Physische Datenunabhängigkeit**
Physische Organisation der Daten ist für die Programme transparent. Der Entwickler eines Programms braucht keine Zugriffspfade etc. zu wissen. Spätere Änderungen des physischen Datenbankaufbaus brauchen in den Anwendungsprogrammen nicht berücksichtigt zu werden.
- **Logische Datenunabhängigkeit**
Ein Programmierer oder Benutzer muss nur die Daten und Beziehungen kennen, die für sein Programm von Bedeutung sind. Andere Felder können hinzugefügt, gelöscht und verändert werden.