

Datenbanksysteme

Das umfassende Lehrbuch

» Hier geht's
direkt
zum Buch

DAS VORWORT

Vorwort

»Geld regiert die Welt.«

»Das Internet regiert die Welt.«

»Datenbanken regieren die Welt.«

Natürlich sind alle drei Aussagen irgendwie richtig, aber an dieser Stelle ist nur der dritte Aspekt relevant. Tatsächlich sind Datenbanken in unserem Leben allgegenwärtig: Jede Suche im Internet, jeder Einkauf in einem Onlineshop, jede medizinische Untersuchung, jeder Eintritt in ein Konzert, jede Fahrt in einem Skigebiet, jeder Aufenthalt in einer Parkgarage, jede bestandene oder missglückte Prüfung während des Studiums hinterlässt Einträge (*Datensätze*) in einer Datenbank.

In diesem Buch beginne ich damit, Ihnen die Grundlagen und Funktionsweise von Datenbankmanagementsystemen zu erklären. Dieser sperrige Begriff bezeichnet Programme, die Datenbanken speichern – z. B. Oracle Database Server, Microsoft SQL Server oder MySQL. Diese drei Programme sind Vertreter aus der Gruppe *relationaler* Datenbankmanagementsysteme. Solche Systeme garantieren die Sicherheit und die Konsistenz von Daten auch beim gleichzeitigen Zugriff durch mehrere Programme oder Clients. Im Detail werden diese Sicherheitsanforderungen durch die Begriffe *Atomicity*, *Consistency*, *Isolation* und *Durability* umschrieben (zusammen kurz ACID).

Ein Grundkonzept relationaler Datenbanken besteht darin, dass Datensätze unterschiedlicher Tabellen aufeinander verweisen. Das ermöglicht eine gut strukturierte Abbildung komplexer Daten ohne Redundanz. Ein inhaltlicher Schwerpunkt dieses Buchs liegt auf dem optimalen Design relationaler Datenbanken. Das richtige Design hat großen Einfluss darauf, wie gut sich die Daten später durchsuchen lassen und wie effizient Datenbankanwendungen laufen. Fehler in der Designphase können später kaum mehr korrigiert werden.

Bei der Nutzung von Datenbanken tritt die *Structured Query Language* (SQL) in den Vordergrund. Damit formulieren Sie Kommandos, um nach Daten zu suchen (SELECT), um diese zu ändern (INSERT, UPDATE und DELETE) sowie um diverse andere Aktionen durchzuführen – beispielsweise Kommandos, die Transaktionen bündeln, neue Tabellen einrichten oder Benutzer-Accounts anlegen. In mehreren Kapiteln erläutere ich die relationale Algebra, die als wissenschaftliche Grundlage von SQL gilt, sowie die vielfältigen Anwendungsformen von SQL. Dabei gehe ich auch auf neue Entwicklungen und Randbereiche ein, z. B. auf die Formulierung rekursiver Abfragen, auf die Verarbeitung geografischer Daten oder auf den Umgang mit XML- und JSON-Dokumenten.

Mit dem Datenbankentwurf und dem Ausführen von SQL-Kommandos ist es nicht getan: Administratoren müssen sich um einen anhaltend sicheren Betrieb kümmern, Benutzer-Accounts einrichten, Backups durchführen und vielleicht ein Replikationssystem konfigurieren. In der Programmierung sind Sie für die Entwicklung von Desktop-Programmen, Weboberflächen oder Apps zuständig, die die Anwendung von Datenbanken so einfach wie möglich machen. Stored Procedures können dabei helfen, dass innerhalb der Datenbank stets bestimmte Regeln (die sogenannte *Geschäftslogik*) eingehalten werden. Auf all diese Aspekte gehe ich im vierten Teil dieses Buchs kurz ein, um so die ganze Bandbreite der Anwendung von Datenbankmanagementsystemen zu beleuchten.

Das relationale Modell gilt zwar seit über 30 Jahren als Industriestandard – aber es ist nicht frei von Einschränkungen und Nachteilen. Problematisch ist vor allem die mangelhafte Skalierbarkeit für extrem große Datenbanken. Deswegen hat sich in den letzten Jahren in der Datenbankwelt eine Art Gegenbewegung unter dem Sammelbegriff *NoSQL* etabliert. In einem eigenen Kapitel gehe ich darauf ein, durch welche Merkmale sich NoSQL-Systeme von ihren relationalen Vorgängern absetzen und unter welchen Umständen ihr Einsatz zweckmäßig ist. Ein weiteres Kapitel (neu in der 2. Auflage) zeigt anhand von MongoDB die praktische Anwendung eines NoSQL-Systems.

Viel Erfolg!

Mit diesem Buch richte ich mich an alle, die den Umgang mit Datenbanken bzw. mit Datenbankmanagementsystemen lernen und vor allem verstehen möchten. Das Buch ist als Begleitlektüre zum Datenbankunterricht in der IT-Ausbildung konzipiert, eignet sich aber auch für das Selbststudium.

Besonders wichtig war es mir, bei aller notwendigen Theorie möglichst viel Praxis einfließen zu lassen, um Ihnen das Rüstzeug für reale Projekte zu geben. Dementsprechend breit ist die Palette der im Buch präsentierten Beispiele. Viele Kapitel enden mit (oft umfangreichen) Übungs- und Wiederholungsaufgaben, deren Lösungen Sie im Anhang finden.

Ich wünsche Ihnen viel Erfolg beim Erlernen der Datenbanktheorie und bei der praktischen Anwendung dieses Wissens!

Michael Kofler (<https://kofler.info>)

PS: Bedanken möchte ich mich an dieser Stelle bei Prof. Thomas Eppler, bei Holger Jakobs und bei Stefan Pröll, die mir eine Menge wertvolles Feedback zum Manuskript gegeben haben.

Die Rolle von MySQL in diesem Buch

Es ist möglich, Datenbanktheorie quasi in den luftleeren Raum hinein zu vermitteln. Davon halte ich aber nichts. Damit Sie lernen, mit Datenbanken umzugehen, müssen Sie eigene Datenbanken entwerfen, mit Daten füllen und Abfragen für diese Daten testen. Deswegen sollte Ihnen bei der Lektüre dieses Buchs ein *Database Management System* (DBMS) zur Verfügung stehen, um die vielen in diesem Buch präsentierten Beispiele selbst auszuprobieren.

Für dieses Buch habe ich mich für MySQL als Arbeitsumgebung entschieden. Für das Open-Source-Programm sprechen einige Argumente:

- ▶ Der Datenbank-Server ist inklusive des hervorragenden Administrationswerkzeugs *MySQL Workbench* kostenlos für alle gängigen Plattformen verfügbar (Windows, macOS, Linux).
- ▶ MySQL ist sehr weit verbreitet und bildet für viele Websites das Datenbankfundament. MySQL-Kenntnisse sind also für Ihre weitere berufliche Arbeit vorteilhaft.
- ▶ MySQL ist gut dokumentiert.

Wie Sie den MySQL-Server und die MySQL Workbench auf Ihrem Rechner installieren und einige Musterdatenbanken einrichten, beschreibe ich im Anhang. Als zusätzlichen Service ermögliche ich Ihnen den Zugriff auf einige Musterdatenbanken direkt über das Internet. Damit können Sie ohne eine eigene Server-Installation SQL-Abfragen ausprobieren. (Ganz ersetzt dieser Service eine eigene Installation aber nicht: Sie können damit keine Daten ändern, keine eigenen Datenbanken einrichten und auch keine administrativen Funktionen ausprobieren.)

MySQL versus MariaDB

Nachdem MySQL von Oracle übernommen wurde, haben ehemalige MySQL-Entwickler das Projekt MariaDB gestartet. Das Ergebnis ist ein weitgehend kompatibler Datenbank-Server mit ein wenig abweichenden Lizenzbestimmungen. In meinem beruflichen Umfeld arbeite ich häufig mit MariaDB unter Linux.

Für dieses Buch habe ich mich aber ganz pragmatisch auf MySQL festgelegt. Im Vordergrund des Buchs stehen Datenbankgrundlagen. Es fehlt der Platz, um auf die kleinen Unterschiede zwischen MySQL und MariaDB einzugehen. Durch die Beschränkung auf *ein* Programm bleiben Installationsanleitungen, der Umgang mit Beispielen usw. übersichtlicher.

Wenn Sie schon mit MariaDB vertraut sind, können Sie dieses DBMS selbstverständlich auf Ihrem Testrechner verwenden. Fast alle in diesem Buch präsentierten Beispiele funktionieren auch dort.

Glossar

In diesem Buch kommen einige Abkürzungen immer wieder vor, ohne dass ich sie jedes Mal neu erlautere.

Kürzel	Bedeutung
ACID	<i>Atomicity, Consistency, Isolation und Durability</i>
API	Application Programming Interface (Programmierschnittstelle)
DB	Database
DBMS	Database Management System
ER-Modell	Entity-Relationship-Modell
FK	Foreign Key = Fremdschlüssel
JDBC	Java Database Connectivity (Java-Schnittstelle zur Datenbankprogrammierung)
ODBC	Open Database Connectivity (universelle Programmierschnittstelle zur Datenbankprogrammierung)
OODBMS	Object-Oriented Database Management System
ORDBMS	Object-Relational Database Management System
PK	Primary Key = Primärschlüssel
RDBMS	Relational Database Management System
SQL	Structured Query Language (ehemals auch Standard Query Language)
XML	Extensible Markup Language

Tabelle 1 Wichtige Abkürzungen aus der Datenbankwelt