

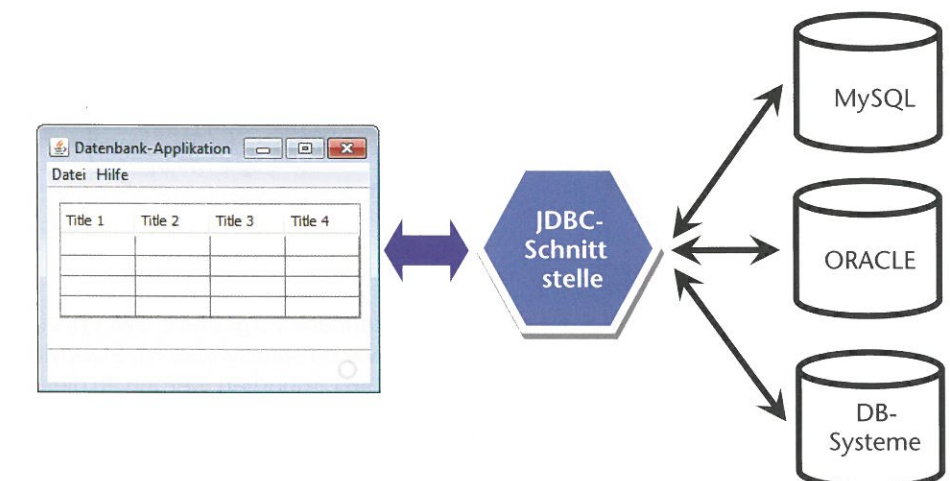
## 9 Datenbankzugriff mit Java

### 9.1 Datenbankzugriff mit Java

Das Speichern von Daten ist eine elementare Aufgabe von Anwendungsprogrammen. Natürlich kann eine Anwendung das Speichern von Daten mithilfe von Dateioperationen selbst durchführen. Für wenige Daten ist das wahrscheinlich auch die beste Wahl bei der Entwicklung einer Anwendung, weil sie damit relativ unabhängig ist. Wenn allerdings viele Daten (oder Datensätze) zu speichern sind und die Daten zusätzlich einen komplizierten Aufbau haben, dann ist die Speicherung in einer Datenbank in Betracht zu ziehen. Der große Vorteil bei einer Datenbankanbindung ist die Unabhängigkeit der Anwendung von der technischen Umsetzung der Datenspeicherung. Das erledigt die Datenbank im Hintergrund. Auch das Ändern oder Löschen von Daten ist bequem durch die entsprechenden Datenbankbefehle zu realisieren. Die bereits ausführlich behandelte Abfragesprache **SQL** spielt hierbei eine wichtige Rolle.

#### 9.1.1 Datenbankanbindung mit JDBC

Java bietet eine Vielzahl von Klassen, um die Anbindung an eine Datenbank zu realisieren. Diese Klassen sind unter dem Oberbegriff **JDBC (Java Database Connectivity)** gesammelt. Für die meisten Datenbanken existieren **JDBC-Schnittstellen**, die den Datenbankzugriff aus einer Java-Anwendung in die entsprechenden Befehle der Datenbank umwandeln. Damit ist es für den Java-Programmierer im Prinzip egal, mit welchem Datenbanksystem im Hintergrund gearbeitet wird – der Zugriff ist gleich. Die folgende Abbildung zeigt das Grundprinzip dieses Zugriffs:



#### 9.1.2 JDBC-Treiber laden und eine Verbindung aufbauen

Um eine Verbindung zu einer Datenbank aufzubauen, muss der entsprechende Treiber vorliegen. Das Laden des Treibers in den Speicher kann dann mit einer Methode der Klasse **DriverManager** erfolgen. Diese Klasse ist Bestandteil des **java.sql**-Paketes, welches in das Java-Projekt importiert werden sollte. Nach dem erfolgreichen Laden des Treibers kann dann eine Verbindung zur Datenbank aufgebaut werden. Das geschieht mithilfe der Klasse **Connection**, die über den **DriverManager** eine Verbindung zur Datenbank erhält. Je nach Datenbank sind Nutzernamen und Passwörter anzugeben. Über ein **Statement**-Objekt kann dann eine Abfrage gestartet und mit einem **ResultSet**-Objekt ausgelesen werden.