

4 Grundlagen der Datenbankentwicklung

In diesem Kapitel werden wir die Grundlagen der Konzeption von relationalen Datenbanken beschreiben. Dazu werden Sie die einzelnen Entwicklungsschritte von der Problemanalyse bis zum fertigen Datenmodell kennen lernen. Für den Datenbankentwurf werden wir das so genannte *Entity-Relationship-Modell* benutzen. Darüber hinaus werden grundlegende Begriffe des relationalen Datenmodells erläutert und in die Normalisierung von Relationen wird eingeführt.

4.1 Von der Realwelt zur Datenbank

Der Datenbankentwurf ist eine der wichtigsten Aufgaben, die der Datenbankentwickler auszuführen hat. Bevor man mit dem Aufbau einer Datenbank beginnt, muss zunächst ein Datenmodell definiert werden, ein möglichst exaktes Abbild eines Ausschnitts der Realwelt. Dazu ist es erforderlich, dass der Datenbankentwickler eine genaue Analyse der Objekte und Prozesse der Realwelt durchführt (Abbildung 4.1).

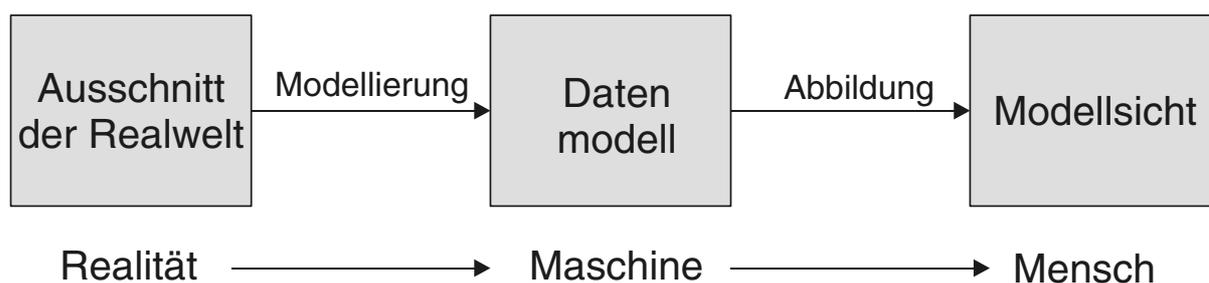


Abb. 4.1: Von der Realwelt zur Datenbank

Natürlich ist klar, dass ein Datenbankspezialist bereits bei der Modellierung eines Problems in Objekten und Relationen denkt. Bei der Analyse eines Problems erkennt man, dass die meisten Dinge und Prozesse in der

Realwelt als Objekte mit entsprechenden Eigenschaften und durch Objektbeziehungen beschrieben werden können.

Wir wollen uns diesen Sachverhalt einmal am Beispiel der Buchausleihe in einer Bibliothek veranschaulichen. Objekte sind in diesem konkreten Beispiel einerseits Bücher und andererseits Personen, die diese Bücher entleihen wollen. In einem Datenmodell fasst man gleichartige Objekte, also beispielsweise alle Bücher oder alle Personen, zu einem Objekttyp, z.B. BUCH bzw. PERSON, zusammen. Der Vorgang einer Buchausleihe in der Bibliothek setzt dabei Objekte vom Objekttyp PERSON mit Objekten vom Objekttyp BUCH in eine Beziehung, nämlich genau dann, wenn eine oder mehrere Personen ein oder mehrere Bücher entleihen wollen. Natürlich merken wir sofort, dass die Beziehung zwischen Personen und Büchern noch etwas näher beschrieben werden muss. Wir definieren daher den Beziehungstyp AUSLEIHE, der die konkreten Ausleihvorgänge beschreibt, beispielsweise welche Person wann welches Buch ausleiht. Für unser Beispiel ergeben sich folgende Objekt- und Beziehungstypen:

- ▶ Objekttypen: PERSON, BUCH
- ▶ Beziehungstyp: AUSLEIHE

Jedem Objekttyp sind bestimmte Eigenschaften zugeordnet, die Attribute eines Objekttyps. Alle Objekte eines Objekttyps besitzen daher die gleichen Eigenschaften. Attribute für Objekte vom Typ PERSON können beispielsweise der Name, der Vorname, das Geburtsdatum, die Adresse oder die Personenkennzahl sein. Für ein Objekt vom Typ BUCH sind dagegen mögliche Attribute der Buchtitel, der Autor, das Erscheinungsjahr oder die ISBN-Nummer. Beziehungstypen können ebenfalls Attribute besitzen. Der Beziehungstyp AUSLEIHE kann beispielsweise durch die Erfassung des Ausleihdatums näher beschrieben werden. Das Datenmodell für unser einfaches Beispiel Buchausleihe ist schematisch in Abbildung 4.2 dargestellt.

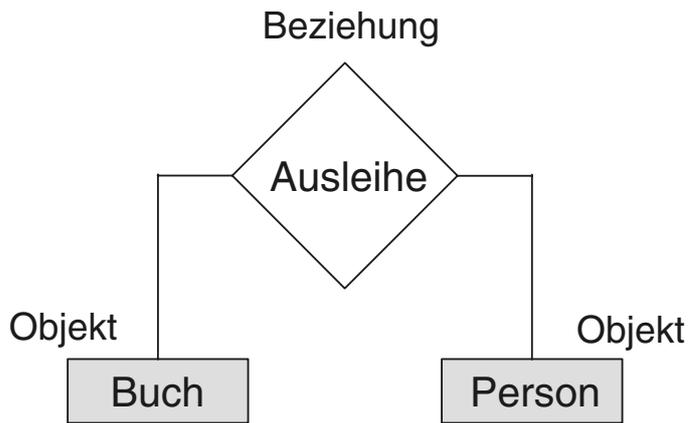


Abb. 4.2: Datenmodell Buchausleihe

Das Datenmodell Buchausleihe bildet dabei folgende Prozesse ab: Eine Bibliothek hat in ihrem Bestand Bücher (Objekttyp BUCH), die von Personen (Objekttyp PERSON) entliehen werden können. Dieser Prozess wird durch den Beziehungstyp AUSLEIHE beschrieben. Diese Art der Darstellung nennt man auch *Entity-Relationship-Modell* oder *ER-Modell*.

An dieser Stelle ist bereits eine wesentliche Aufgabe der Datenbankentwicklung abgeschlossen. Der nächste Schritt besteht nun darin, das Datenmodell an das jeweilige Datenbanksystem anzupassen. Verwenden Sie beispielsweise ein relationales Datenbanksystem, müssen Sie nun auf der Grundlage des ER-Modells Tabellen definieren. Nachdem Sie diesen Schritt ausgeführt haben, können Sie das Datenmodell auf Ihrem Rechner implementieren. Dazu stellen die Datenbanksysteme spezielle Werkzeuge bereit. Für relationale Datenbanken benutzen Sie die standardisierte Datenbankabfragesprache SQL. Nachdem die Tabellen im System definiert worden sind, kann man beginnen, die Datenbank mit konkreten Daten zu füllen. Hat man erst einmal einen Datenbestand angelegt, kann man in diesem Bestand auch recherchieren. In unserem Beispiel wären mögliche Fragestellungen für eine Datenbankrecherche:

- ▶ Welche Bücher hat eine bestimmte Person ausgeliehen?
- ▶ Welche Person hat die Ausleihfrist überzogen?
- ▶ Welche Person hat ein bestimmtes Buch entliehen?

Fassen wir jetzt noch einmal alle Schritte vom Entwurf bis zur fertigen Datenbank zusammen:

1. Analyse des Problems, Definition von Objekttypen und Objektbeziehungen
2. Entwicklung des Datenmodells, beispielsweise als ER-Modell
3. Definition von Tabellen
4. Dateneingabe
5. Datenbankabfrage

4.2 Das Entity-Relationship-Modell

Das *Entity-Relationship-Modell* ist ein grafisches Hilfsmittel für den Entwurf eines Datenmodells. Es beinhaltet grafische Elemente zur Darstellung von Objekten (Objekttypen) und deren Objektbeziehungen (Relationship). Wir werden im nächsten Abschnitt ausführlich beschreiben, wie man ein solches Datenmodell in ein relationales Datenmodell überführt. Die Grundbausteine eines ER-Modells sind Objekttypen, die bestimmte Eigenschaften (Attribute) besitzen. Ein Objekttyp kann in der Realwelt dabei alles Mögliche repräsentieren: Dinge, Prozesse oder Funktionseinheiten. Zwischen Objekten können Beziehungen bestehen, die einen unterschiedlichen Komplexitätsgrad aufweisen können. Wie wir bereits in den vorangegangenen Abschnitten gesehen haben, fasst man in einem Datenmodell konkrete Entitäten, z.B. die Person *Rolf Däßler* oder den Buchtitel *Einsteigerseminar MySQL5*, zu Objekttypen, beispielsweise BUCH oder PERSON, zusammen. Das Gleiche gilt auch für Beziehungen. Konkrete Beziehungen (z.B. die Ausleihe von Buch x durch die Person y) werden durch einen Beziehungstyp charakterisiert. Das ER-Modell stellt dafür die in Tabelle 4.1 aufgeführten grafischen Elemente zur Verfügung:

Element	Bedeutung	Symbol
Entitätentyp	Objekttyp	Rechteck
Beziehungstyp	Objektbeziehungen	Raute
Attribute	Objekteigenschaften	Kreise oder Ovale

Tab. 4.1: Grafische Elemente des ER-Modells

Ein ER-Modell beschreibt ein formales Knoten-Kanten-Modell, das Objekttypen (Knoten) und deren Beziehung (Kanten) visualisiert. Ein wichtiger Schritt der Modellierung ist die Festlegung der Objekttypen. Anders ausgedrückt: jeder Zustand oder jeder Prozess in der Realwelt, der in einer Datenbank abgebildet werden soll, muss zunächst analysiert und in kleinste Komponenten oder Teilprozesse zerlegt werden.

Nachdem man die Objekttypen definiert hat, legt man fest, in welcher Beziehung die Entitäten zueinander stehen. Das kann natürlich für verschiedene Problemstellungen ganz unterschiedlich sein. Nehmen wir beispielsweise die Objekttypen PERSON und BUCH. Versuchen wir in der Datenbank beispielsweise den Prozess der Buchausleihe abzubilden, besteht zwischen einer Person und einem Buch eine andere Beziehung, als wenn eine Person ein Buch in einem Buchladen kauft oder eine Person ein Buch schreibt. In allen drei genannten Fällen treten gleiche Entitäten auf. Die Art der Beziehung wird jedoch durch die Verschiedenheit der Prozesse charakterisiert.

In einem weiteren Modellierungsschritt bestimmt man nun die Eigenschaften der Objekttypen. Die Eigenschaften der Objekte der Realwelt werden durch die Werte der Attribute in einem Datenmodell beschrieben. Dabei ist anzumerken, dass nicht alle Eigenschaften modelliert werden, sondern nur solche, die für eine Problemstellung relevant sind. Damit besitzen wir über die Auswahl der Attribute, falls gleiche Entitäten eine Rolle spielen, noch eine zusätzliche Möglichkeit, das Datenmodell an eine spezifische Problemstellung anzupassen.

Objektbeziehungen

Eine der wichtigsten Aufgaben des ER-Modells ist es, zu zeigen, welche Beziehungen zwischen den Objekten bestehen. Diese Objektbeziehungen können genau wie die Objekttypen Eigenschaften besitzen. Zudem kann die Art der Beziehung zwischen Objekten sehr unterschiedlich sein. Prinzipiell unterscheidet man zwei Arten: eindeutige und mehrdeutige Beziehungen. Bei eindeutigen Beziehungen gibt es immer genau eine Verbindung zwischen zwei Entitäten eines Objekttyps. Bei mehrdeutigen Beziehungen stehen eine oder mehrere Entitäten eines Objekttyps mit einer oder mehreren Entitäten eines anderen Objekttyps in Beziehung. Tabelle 4.2 fasst die verschiedenen Arten der Objektbeziehungen zusammen.

Beziehungstyp	Beschreibung	Beispiel
eindeutig 1:1	Einem Objekt vom Typ A ist genau ein Objekt vom Typ B zugeordnet	Objekttypen: Person (Typ A) und Bibliotheksausweis (Typ B) Jede Person besitzt genau einen Bibliotheksausweis. Jeder Bibliotheksausweis ist genau einer Person zugeordnet.
funktional 1:n	Jedes Objekt vom Typ A kann mit beliebig vielen Objekten des Typs B in Beziehung stehen. Jedes Objekt vom Typ B steht mit genau einem Objekt vom Typ A in Beziehung.	Objekttypen: Verlag (Typ A) und Buch (Typ B) Ein Verlag kann mehrere Buchtitel herausgeben. Jedes Buch hat nur einzigen Verlag.
komplex m:n	Jedes Objekt vom Typ A bzw. vom Typ B kann mit beliebig vielen Objekten des jeweils anderen Typs in Beziehung stehen.	Objekttypen: Buch (Typ A) und Person (Typ B) Ein Buch kann nacheinander von mehreren Personen ausgeliehen werden. Eine Person kann mehrere Bücher ausleihen.

Tab. 4.2: Objektbeziehungen im ER-Modell

Eineindeutige Objektbeziehungen sind in der Realwelt eher selten anzutreffen. Häufiger tritt dagegen die $1:n$ -Beziehung auf. Auch die $m:n$ -Beziehung liefert oft eine adäquate Abbildung der Realwelt. Im Gegensatz zu der $1:1$ - bzw. $1:n$ -Beziehung, die direkt in das relationale Datenbankmodell überführt werden kann, lässt sich die $m:n$ -Beziehung nicht direkt im relationalen Datenmodell abbilden. Liegt eine $m:n$ -Beziehung vor, muss sie zuerst durch die Definition einer neuen Relation in zwei $1:n$ -Beziehungen überführt werden.

Abbildung 4.3 zeigt noch einmal wichtige Beziehungstypen in einer Übersicht.

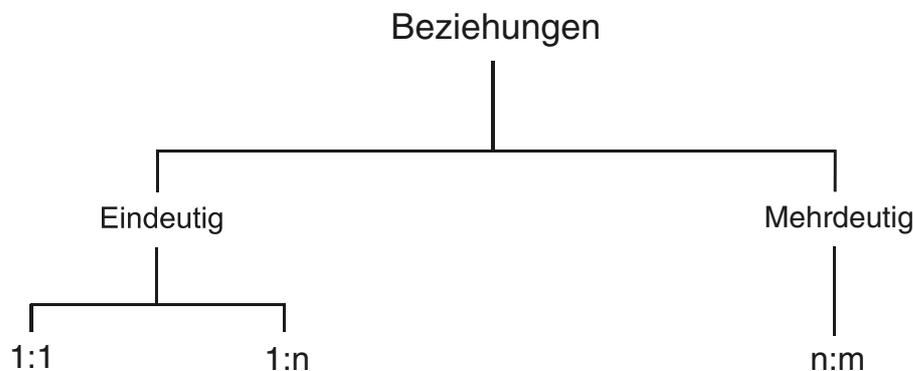


Abb. 4.3: Typologie wichtiger Objektbeziehungen

Das ER-Modell Buchausleihe

Wir wollen uns jetzt ein Beispiel für ein Datenmodell anschauen, das mit Hilfe eines ER-Modells veranschaulicht wird: eine einfache Buchausleihe in einer Bibliothek. Wir beschränken das Problem auf vier Objekttypen: BUCH, PERSON, VERLAG und AUTOR und untersuchen die Beziehungen zwischen diesen Objekttypen. Obwohl ein Autor auch eine Person ist, ist es trotz ähnlicher Eigenschaften sinnvoll, beide als unterschiedliche Objekttypen zu definieren, da wir beide in einer unterschiedlichen Beziehung zum Objekttyp BUCH betrachten wollen: Eine Person leiht ein Buch aus, während ein Autor ein Buch schreibt. Untersuchen wir die Beziehung Person-Buch, so scheint es zunächst, dass es sich um eine

$m:n$ -Beziehung handelt, da prinzipiell eine Person mehrere Bücher entleihen kann und ein Buch durch mehrere Personen entliehen werden kann. Je nach Problemstellung kann es sich aber auch um eine $1:n$ -Beziehung handeln, da zur gleichen Zeit jeweils nur eine Person ein und dasselbe Buch entleihen kann. Bei der Beziehung zwischen einem Buch und dem Verlag handelt es sich um eine $1:n$ -Beziehung, da jedem Buch jeweils nur ein Verlag zugeordnet ist. Komplizierter gestaltet sich dagegen die Beziehung zwischen Buch und Buchautor. Hier handelt es sich in jedem Fall um eine $m:n$ -Beziehung, da einerseits ein Buch von mehreren Autoren geschrieben werden kann und andererseits ein einzelner Autor auch mehrere Bücher schreiben kann (Abbildung 4.4).

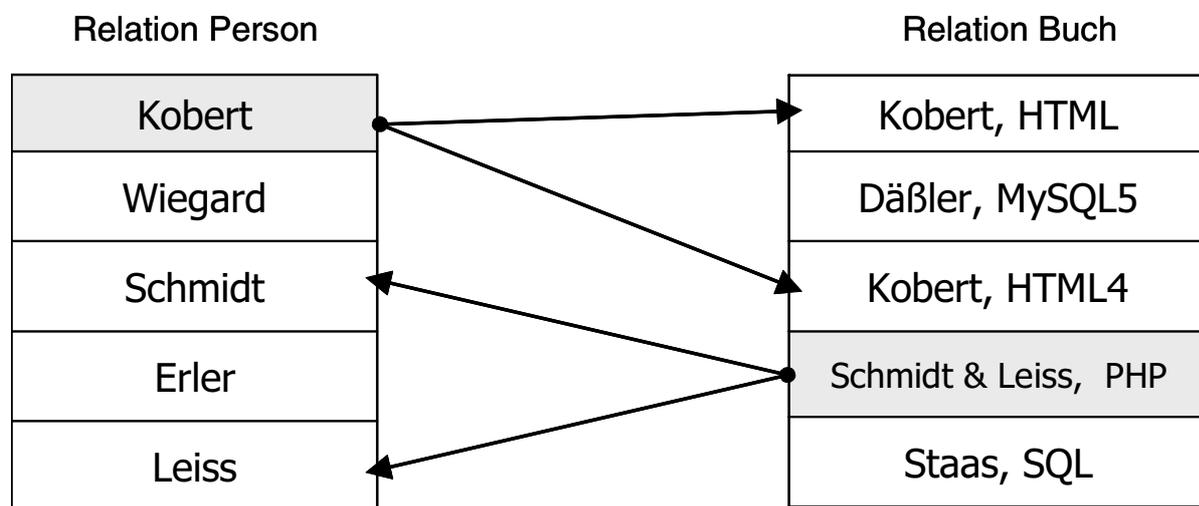


Abb. 4.4: $m:n$ -Beziehung zwischen Autor und Buch im Modell Buchausleihe

Aus dieser Problemanalyse ergibt sich das in Abbildung 4.5 dargestellte Diagramm. Zur Vereinfachung verwenden wir in diesem Schema nur die wichtigsten Attribute.

Das ER-Modell in Abbildung 4.5 beschreibt folgendes semantisches Modell:

- ▶ Bücher werden von Personen ausgeliehen. Eine Person kann mehrere Bücher gleichzeitig ausleihen. Ein und dasselbe Buch kann nacheinander von mehreren Personen ausgeliehen werden.

- ▶ Ein Autor schreibt ein oder mehrere Bücher. An einem Buch können auch mehrere Autoren beteiligt sein.
- ▶ Ein Verlag verlegt ein oder mehrere Bücher. Jedes Buch wird von genau einem Verlag verlegt.

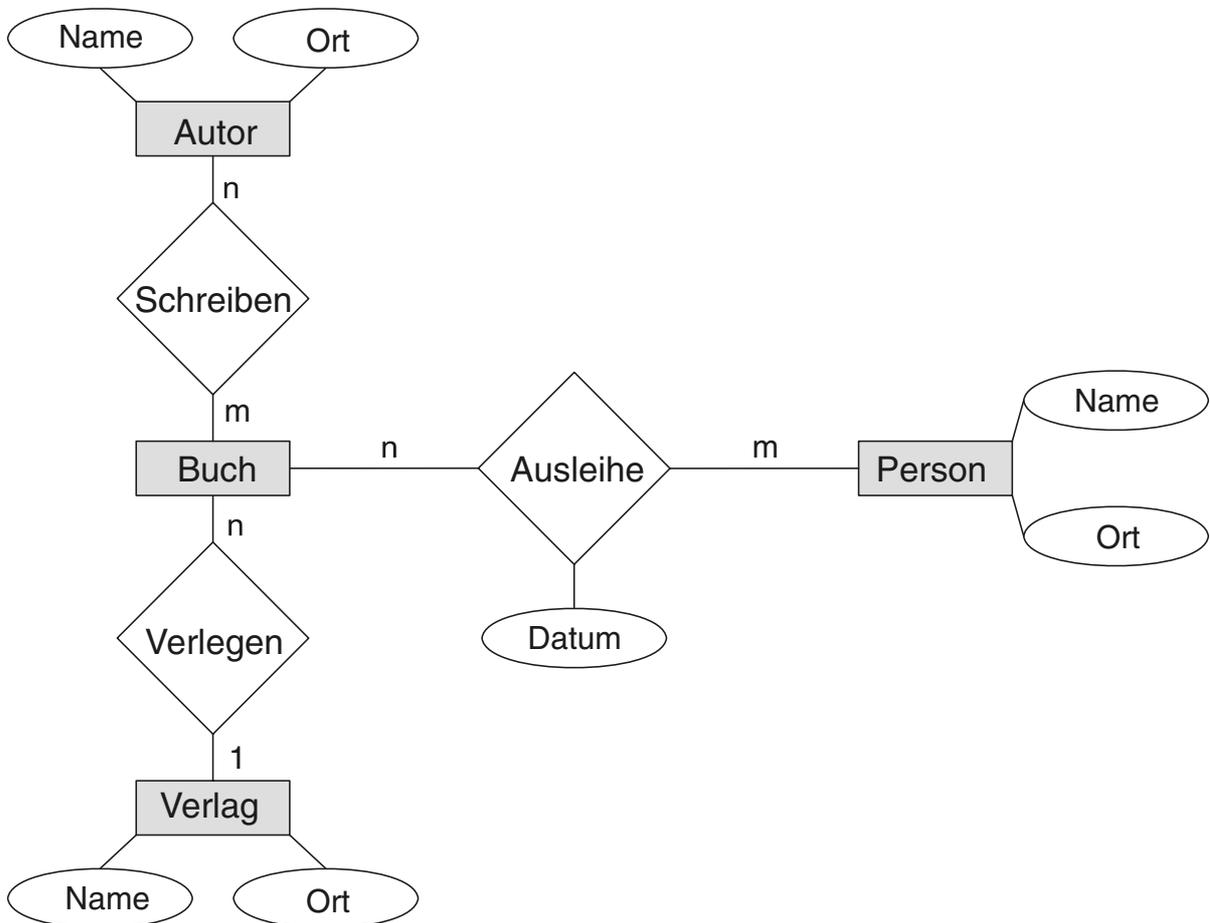


Abb. 4.5: ER-Modell Buchausleihe

4.3 Normalisierung von Relationen

Zur Gewährleistung einer widerspruchsfreien Datenverwaltung auf der Grundlage des relationalen Datenmodells wurde ein Regelwerk festgelegt, die sogenannte *Normalisierung*, nach dem ein relationales Datenmodell entwickelt werden sollte. Die verschiedenen Normalformen charakteri-