

Entwurf von Datenbanken (Normalisierung)

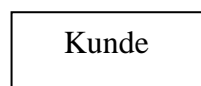
Bevor man mit der Implementierung einer Datenbank in einem real existierenden Datenbanksystem beginnt, ist es notwendig, die Datenbank auf einer konzeptionellen Ebene zu modellieren und anschließend die Datenbankstrukturen zu optimieren. Das Entity-Relationship-Modell und die Normalisierung sind geeignete Werkzeuge zur Modellierung einer Datenbank.

Entity-Relationship-Modell:

Das ERM wurde 1976 von Peter Chen entwickelt und dient als Modellierungswerkzeug für Datenbanken. Ziel des ERM ist es, auf einer konzeptionellen Ebene (d. h. unabhängig von der datenbankspezifischen Modellierung in einem Datenbanksystem) die Darstellung von Daten und Beziehungen zu ermitteln. Das Modell beinhaltet zwei Grundelemente, nämlich Entitäten (=Entities) und Beziehungen (=Relationships).

Entitiy-Typ: Ein Entity ist ein realer Gegenstand, ein abstraktes Konzept oder ein Ereignis. Beispielsweise ist ein Kunde, ein Artikel, ein Mitarbeiter oder eine Maschine ein Objekt, das ein Entity darstellt. Mehrere Gleichartige Entities (z. B. Kunde Weiler, Kunde Schmidt, Kunde Meier) werden zu einem Entitytyp zusammengefasst.

Darstellung: Entitytypen werden im ERM als Rechteck dargestellt. In das Rechteck schreibt man die Bezeichnung des Entitytypen.



Relationship: Eine Relationship ist eine Beziehung zwischen zwei Entities, die meist durch ein Verb beschrieben werden kann. Beispielsweise **erteilt** ein Kunde einen Auftrag, ein Mitarbeiter **arbeitet** an einer Maschine oder ein Schüler **gehört zu** einer Klasse.

Darstellung: Beziehungen werden im ERM in Form von Rauten dargestellt. In die Raute selbst schreibt man den Namen der Beziehung (=Verb)

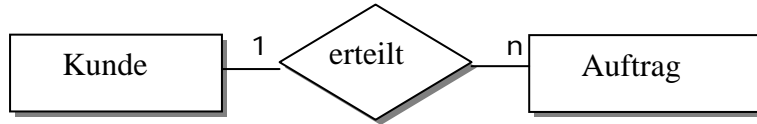
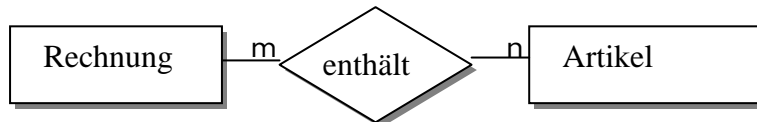


Wenn wir die Darstellung einer Beziehung im Zusammenhang betrachten, so erkennen wir, dass die Raute die über die Beziehung verknüpften Entitytypen (mit einer Linie) verbindet.



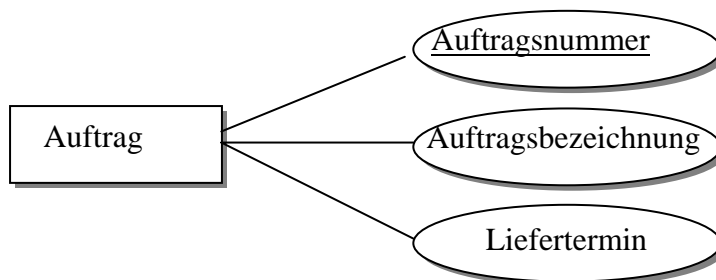
Komplexität von Beziehungen (Kardinalitäten): Die Komplexität einer Beziehung gibt das Verhältnis der beiden Entitytypen in Bezug auf die Beziehung an. Beispielsweise ist in unserem Kulturkreis ein Ehemann mit einer Ehefrau verheiratet. Es handelt sich hierbei um eine 1:1 Beziehung. Bei der Analyse der Beziehung zwischen Kunde und Auftrag erkennen wir, dass ein Kunde mehrere Aufträge erteilen kann, ein bestimmter Auftrag auf der anderen Seite genau einem Kunden zugeordnet werden kann. Hier haben wir eine 1:n Beziehung. Kann auf beiden Seiten eine Beziehung zwischen mehreren Objekten bestehen (verschiedene Rechnungen enthalten mehrere Artikel) so spricht man von einer m:n – Beziehung.

Darstellung: Die Komplexität einer Beziehung wird an den Verbindungslinien zwischen zwei Entitätsmengen verzeichnet. Die Beziehungen zwischen den Entitytypen lassen sich folgendermaßen darstellen:

Beziehungstypen in unserem Beispiel:**1:1 – Beziehung****1:n – Beziehung****m:n – Beziehung**

Attribute (Properties): Attribute (=Eigenschaften) beschreiben die Entitytypen näher. Der Entitytyp Auftrag wird z. B. durch die Attribute Auftragsnummer, Auftragsbeschreibung, Liefertermin usw. beschreiben. Es gibt Nichtschlüssel-Attribute wie z. B. die Auftragsbeschreibung, den Liefertermin oder die Zahlungsbedingungen und es gibt ein oder mehrere Attribute, die der eindeutigen Identifizierung des Entities innerhalb eines Entitytyps dienen (=Schlüsselattribut). Die minimal identifizierende Attributkombination ist ein Schlüsselkandidat und kann als Primärschlüssel festgelegt werden.

Darstellung: Attribute werden in Form von Ellipsen dargestellt. Wenn das Attribut den Primärschlüssel darstellt, wird es unterstrichen.

**Zusammenfassendes Beispiel: Vereinsverwaltung**

Der Kassierer eines Sportvereines will die Mitgliederverwaltung, die derzeit konventionell mit Karteikarten durchgeführt wird, rationalisieren. Hier ein Beispiel einer Erfassungskartei:

Name	Mustermann	Mitgliedsnummer	1000
Vorname	Hans	Eintrittsdatum	2.1.1990
Straße	Bergstr. 1	PLZ	12345
Ort	Musterdorf	Geburtstag	1.1.1970
Beitragsart	Vollmitglied	Abteilungen	Fußball, Tennis
Beitrag	20,00 €		
BLZ	444 333 00	Konto	1234567
Bankname	Stadtsparkasse Musterdorf	Bankort	Musterdorf

- Neben Fußball gibt es noch die Abteilungen Tennis (1), Handball (2), Baseball (3), Leichtathletik (4), Schwimmen (5), Tischtennis (6) und seit kurzem auch Judo (7).
- Die Beitragssätze sind unterschiedlich. Vollmitglieder (VM) zahlen 20,00 €, Ehrenmitglieder (EM) sind von der Beitragszahlung befreit. Jugendliche (J) zahlen 10,00 €, volljährige Familienmitglieder (VF) 15,00 €. Passive Mitglieder (PM) zahlen 10,00 €, der Jugendwart (JW) 12,00 € sowie Trainer (T) 8,00 €.

Stellen Sie das gegebene Beispiel als Entity-Relationship-Modell dar.

Ziele der Normalisierung

Die Ziele des Normalisierungsprozesses sind:

- Erzeugen einer flexiblen, stabilen Datenbank, bei der Veränderungen leicht durchzuführen sind.
- Vermeiden von Regelwidrigkeiten (Anomalien) beim Ändern (=Änderungsanomalie), Löschen (=Löschanomalie) und Einfügen (=Einfügeanomalie) von Datensätzen.
- Möglichst geringe Redundanz und somit geringer Speicherplatzbedarf.

Unnormalisierte Datenbank

Ausgangspunkt des Normalisierungsprozesses ist die unnormalisierte Datenbank. Betrachten wir dazu ein Beispiel: die Daten zur Erzeugung einer Rechnung sollen in einer Datenbank abgespeichert werden. Die Rechnung hat folgendes Aussehen:

PictoNeilCom Systemhaus GmbH Bahnhofstr. 33				
48653 Coesfeld				
RECHNUNG			Datum:	28.03.01
Karl Subke Feldbergstr. 21			Kundennummer:	1223
48249 Dülmen			Rechnungsnummer	123/01
Bei Zahlung bitte angeben.				
Artikelnummer	Artikelbezeichnung	Preis/Einheit	Stückzahl	Gesamtpreis
10001	NoName CD-ROM	0,99 DM	100	99,00 DM
10050	Toner	75,00 DM	2	150,00 DM
10402	Etiketten	25,00 DM	10	250,00 DM
Netto:				499,00 DM
+ 16% MwSt:				79,84 DM
Brutto				578,84 DM
Bitte überweisen Sie den Betrag auf eines unserer Konten innerhalb von 14 Tagen netto ohne Abzug				

Die Analyse der Rechnung führt zu einer Tabelle mit folgenden Feldern:

Rechnung unnormalisiert
Kundennummer
Kundenname
Kundenadresse
RNR
Datum
Netto
ANRPos1
ArtBezPos1
StueckzahlPos1
PreisProEinheitPos1
ANRPos2
ArtBezPos2
StueckzahlPos2
PreisProEinheitPos2
ANRPos3
ArtBezPos3
StueckzahlPos3
PreisProEinheitPos3

Die Werte für die Mehrwertsteuer und den Bruttobetrag sowie der Nettobetrag einer Rechnungsposition werden nicht als Felder in der Tabelle **Rechnung unnormalisiert** angelegt, da sie sich jederzeit wieder berechnen lassen. Das Feld Rechnungsnummer (**RECHNR**) ist der Primärschlüssel.

Die Tabelle **Rechnung unnormalisiert** ist gegenüber Anomalien sehr anfällig, wie die folgenden Beispiele zeigen:

- **Einfügeanomalie (Insertanomalie):** Die Daten für einen Artikel können nur dann aufgenommen werden, wenn es eine Rechnung gibt, in der dieser Artikel berechnet wird. Produkte unseres Sortiments, für die es gegenwärtig keine Bestellung gibt, können nicht erfasst werden.
- **Löschanomalie (Deleteanomalie):** Wird aus der Tabelle die (einzige) Rechnung eines Kunden gelöscht, gehen alle Kundendaten verloren. Bei einer neuen Bestellung müssen alle Kundeninformationen neu erfasst werden.
- **Änderungsanomalie (Updateanomalie):** Ändert sich die Anschrift eines Kunden, für den bereits mehrere Rechnungen erfasst sind, müssen in allen betroffenen Zeilen die Werte korrigiert werden. Wird das an einer Stelle vergessen, entsteht ein inkonsistenter Datenbestand (der Kunde hat zwei verschiedene Adressen, von denen nur eine richtig ist).

Zusätzlich enthält die Tabelle viele redundante Daten. Erhält ein Kunde mehrere Rechnungen, sind die Kundendaten für jede einzelne Rechnung erfasst. Gleiches gilt für die Artikeldaten, wenn ein Artikel in mehreren Rechnungen erscheint.

1. Normalform

Damit eine Tabelle in der 1. Normalform (1NF) vorliegt, müssen folgende Bedingungen erfüllt werden:

- Alle Attributwerte müssen atomar sein (Felder dürfen sich nicht weiter untergliedern lassen).
- Es dürfen keine Wiederholungsgruppen existieren.

Die Tabelle Rechnung widerspricht zwei Bedingungen der 1. Normalform:

1. Die Felder Kundenname und Kundenadresse sind nicht atomar.

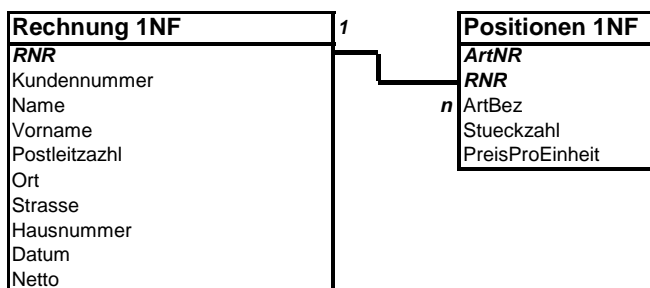
Das Feld Kundenname lässt sich untergliedern in die Felder Kundenvorname (Vorname) und Kundennachname (Nachname). Das Feld Kundenadresse ist ebenfalls nicht atomar. Es lässt sich in die Felder *Postleitzahl*, *Ort*, *Strasse* und *Hausnummer* aufsplitten. Diese Aufteilung in atomare Attributwerte ist wichtig, wenn wir beispielsweise alle Kunden anschreiben wollen, die ihren Geschäftssitz in einem bestimmten Ort haben. Durch die Aufteilung in einzelne Felder lässt sich das recht einfach realisieren.

2. Es existiert eine Wiederholungsgruppe in der Tabelle Rechnung

In der Beispielrechnung hat der Kunde drei unterschiedliche Artikel bezogen. Für jeden Artikel musste deshalb je ein Feld für die Artikelnummer, die Artikelbezeichnung, die Stückzahl und den Preis pro Einheit angelegt werden. Theoretisch müsste man die Struktur der Tabelle noch erweitern, so dass für eine maximale Anzahl von Rechnungspositionen (z. B. in einer Rechnung werden maximal 20 Positionen ausgewiesen) diese vier Felder jeweils verfügbar sind.

Bei der Dimensionierung mit der maximalen Anzahl von Rechnungspositionen wird viel Speicherplatz verschwendet, da die Felder für jede Rechnung angelegt werden (auch für solche mit weniger Positionen) und es besteht auf der anderen Seite immer die Gefahr, dass die Datenbankstruktur gesprengt wird, wenn mehr als 20 Positionen auf einer Rechnung ausgewiesen werden sollen. Das Problem von Wiederholungsgruppen lässt sich durch die Zerlegung der ursprünglichen Tabelle in mehrere Relationen beheben.

Tabellen der Datenbank Rechnung in 1. Normalform



Die sich wiederholenden Attribute (*ArtNR*, *ArtBez*, *Stueckzahl*, *PreisProEinheit*) wurden aus der Ursprungstabelle entfernt und die neue Tabelle *Positionen 1NF* übertragen. Um eine Verknüpfung mit den Kundendaten herstellen zu können (jetzt Tabelle *Rechnung 1NF*) muss zusätzlich das Attribut *RECHNR* in die Tabelle *Positionen 1NF* aufgenommen werden.

In der Tabelle *Rechnung 1NF* ist das Feld *RECHNR* das Primärschlüsselfeld. In der Tabelle *Positionen 1NF* setzt sich der Primärschlüssel aus den Felder *ArtNR* und *RECHNR* zusammen, denn weder die Artikelnummer noch die Rechnungsnummer identi-

fizieren eine Zeile eindeutig. Die Rechnungsnummer wiederholt sich, wenn eine Rechnung mehrere Positionen enthält und die Artikelnummer wiederholt sich, wenn ein gleicher Artikel in verschiedenen Rechnungen berechnet wird.

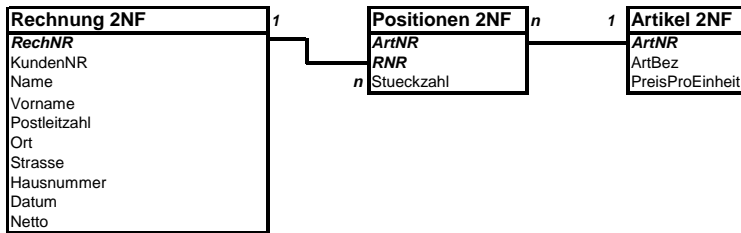
1. Normalform (1NF) Eine Tabelle liegt in der 1. Normalform vor, wenn an jeder Stelle der Tabelle (d. h. in jeder Zelle) nur ein Wert steht.

2. Normalform

Wegen der nach wie vor bestehenden Redundanz sind die Tabellen in der 1NF noch anfällig für Anomalien. Beispielsweise wiederholt sich in der Tabelle *Positionen 1NF* für einen in mehreren Rechnungen enthaltenen Artikel die Artikelbezeichnung. Die Ursache hierfür liegt darin, dass der Artikelname nicht vom gesamten Schlüssel, sondern nur von einem Teil des Schlüssels abhängig ist. In der 2. Normalform (2NF) muss die Datenbank neben den Bedingungen der 1. Normalform noch folgender Anforderung genügen:

Jedes Nichtschlüsselattribut muss voll funktional abhängig vom Gesamtschlüssel, nicht aber von Teilen des Gesamtschlüssels sein.

Tabellen der Datenbank Rechnung in 2. Normalform



In der Tabelle *Rechnung 1NF* sind alle Nichtschlüselfelder vom Primärschlüssel abhängig, denn der Primärschlüssel besteht aus einem einzigen Feld. Für die Relation *Position 1NF* gilt das jedoch nicht. In dieser Tabelle sind die Felder *ArtBez* und *PreisProEinheit* nicht vom Gesamtschlüssel (Felder *ArtNR* und *RECHNR*) abhängig, sondern nur von einem Teil des Schlüssels, nämlich vom *ArtNR*.

Um die Datenbank in die 2. Normalform zu bringen, müssen wir die Information der Relation *Positionen 1NF* auf zwei Tabellen verteilen. Die Relation *Positionen 1NF* wird in der 2. Normalform auf die beiden Tabellen *Positionen 2NF* und *Artikel 2NF* verteilt. In der neuen Relation *Artikel 2NF* wird ein Primärschlüssel für das Feld *ArtNR* definiert. Über dieses Feld lässt sich die Verknüpfung zur Tabelle *Positionen 2NF* herstellen.

2. Normalform (2NF)

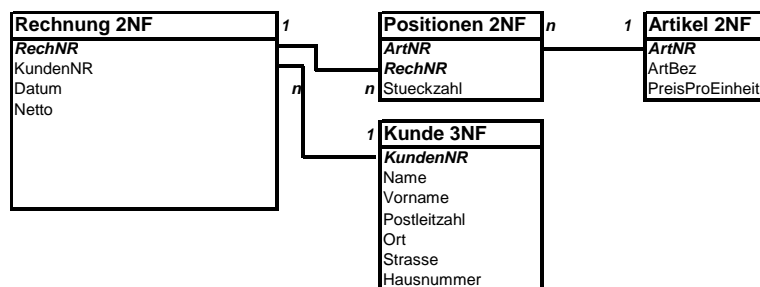
Eine Tabelle liegt in der zweiten Normalform vor, wenn sie sich in der 1NF befindet und jedes Nicht-Schlüsselattribut vom Gesamtschlüssel und nicht bereits von einem Teil des Gesamtschlüssels voll funktional abhängig ist.

3. Normalform

Auch eine Tabelle in der 2NF ist noch nicht ausreichend vor Anomalien geschützt. Ein Kunde kann beispielsweise nach wie vor nicht erfasst werden, wenn er keine Rechnung erhält. Dieses Problem tritt auf, weil die Daten eines Kunden in der Tabelle *Rechnung 2NF* nicht direkt vom Schlüssel *RECHNR* abhängen.

In der 3. Normalform (3NF) muss die Datenbank neben den Bedingungen der 1. und der 2. Normalform noch folgende Bedingung erfüllen: Es dürfen keine funktionalen Abhängigkeiten zwischen Attributen existieren, die nicht als Schlüssel definiert sind, d. h. alle Attribute, die nicht zum Primärschlüssel gehören, müssen direkt von diesem abhängig sein. Es dürfen keine indirekten („transitiven“) Abhängigkeiten bestehen.

Tabellen der Datenbank Rechnung in 3. Normalform



Die Tabelle *Rechnung 2NF* enthält noch Abhängigkeiten zwischen Nichtschlüsselattributen. Die Felder *Name*, *Vorname*, *Postleitzahl*, *Ort*, *Strasse* und *Hausnummer* sind nicht vom Primärschlüssel dieser Tabelle (*RECHNR*) sondern vom Feld *KundenNR* abhängig.

Auch transitive Abhängigkeiten können aufgelöst werden, indem die Felder der Tabelle auf verschiedene Relationen verteilt werden. In der neuen Tabelle *Kunde* werden die Felder *KundenNR*, *Name*, *Vorname*, *Postleitzahl*, *Ort*, *Strasse* und *Hausnummer*

definiert. Alle Felder sind von der *KundenNR* abhängig. Dieses Feld muss als Primärschlüssel in der Relation *Kunde* definiert sein. Nach diesen Veränderungen befinden alle Tabellen in der 3. Normalform.

3. Normalform (3NF)

Eine Tabelle liegt in der dritten Normalform vor, wenn sie sich in der 2NF befindet und jedes Nicht-Schlüsselattribut direkt (nicht transitiv) vom Primärschlüssel funktional abhängig ist.

Aufgabe:

Bei der Abspaltung einer Tabelle in eine weitere Tabelle muss die neu geschaffene Tabelle auf alle drei Normalformen hin überprüft werden. Welchen Vorschlag können Sie für die Tabelle *Kunde 3 NF* machen, um auch diese Tabelle korrekt zu normalisieren?

Übungsaufgaben Normalisierung:

- 1) Folgende unnormalisierten Datenbestände sind für die Persona-Zeitarbeit GmbH in einer Excel-Tabelle aufgeführt. Überführen Sie diese Datenbestände schrittweise in die dritte Normalform. Achten Sie dabei darauf, dass abgesonderten Tabellen über einen Primärschlüssel (Darstellung: Unterstreichen des Schlüsselattributs) eindeutig den jeweiligen Datensatz bestimmen. Stellen Sie das Ergebnis der Normalisierung in einem Entity-Relationship-Modell dar.

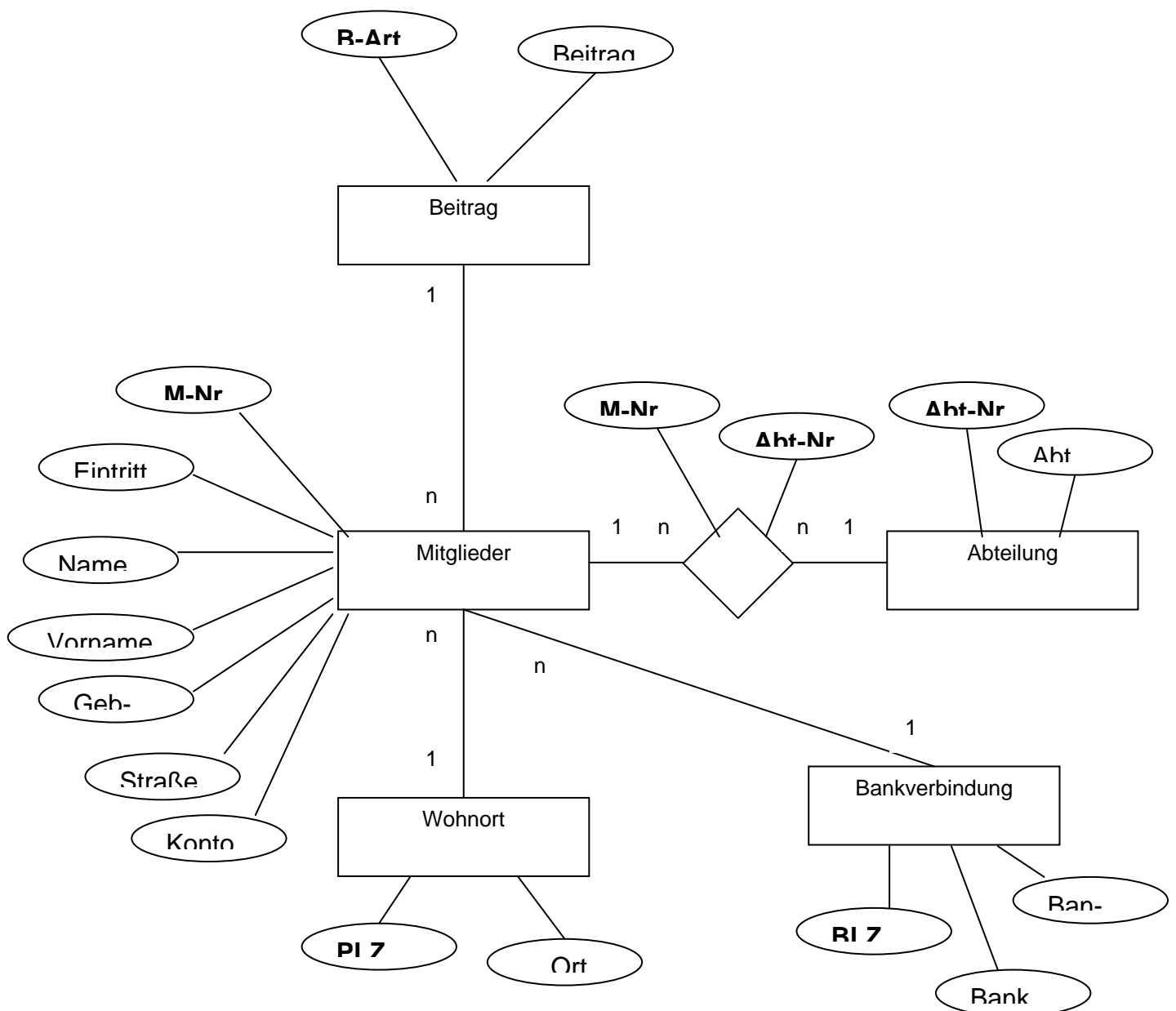
Name und Adresse	Ge- burtsda- tum	Job	Fremd- sprachen	Stun- denlohn	Arbeits- zeit	Stun- denan- zahl	Zula- genstun- den
Willi Wahrberg, Weißdornring 20, 48301 Darup, Tel.: 02549-35388	21.11.59	Mechaniker		18,20 €	325,00 €-Basis	10	3
Gudrun Kauer, Holtkamp 13a, 48301 Darup, Tel.: 02549-28764	20.07.59	Mechaniker	Englisch, Französisch	18,20 €	Teilzeit	25	4
Traugott Taugenicht, Holtkamp 52, 48301 Darup, Tel.: 02549-29052	14.07.59	Koch	Englisch	15,90 €	Vollzeit	150	18
Leonie Graf, Holtkamp 25, 48301 Darup, Tel.: 02549-29628	14.04.55	Haus-haltshilfe	Englisch, Spanisch	13,80 €	Vollzeit	120	14
Viktor Nolte, Holtkamp 22c, 48301 Darup, Tel.: 02549-29916	27.05.69	Fliesenleger		17,80 €	325,00 €-Basis	8	8
Veronika Tönnies, Weißdornring 8, 48301 Darup, Tel.: 02549-31068	07.09.67	Haus-haltshilfe		13,80 €	Vollzeit	145	15
Ralph Deckers, Londoner Weg 68, 48301 Darup, Tel.: 02549-31932	02.06.25	Dachdecker	Russisch	20,00 €	Teilzeit	64	8
Trudi Zwergstein, Londoner Weg 2, 48301 Darup, Tel.: 02549-34524	30.12.58	Dachdecker		20,00 €	Vollzeit	200	16
Tino Wuttke, Reddagstrasse 52, 48301 Darup, Tel.: 02549-21852	26.01.62	Gärtner	Englisch	14,50 €	325,00 €-Basis	12	0
Beate Winzger, Birminghamstrasse 22, 48301 Darup, Tel.: 02549-35100	03.03.58	Putzfrau	Italienisch, Spanisch	12,30 €	Teilzeit	55	21
Simone Wittrup, Holtkamp 14, 48301 Darup, Tel.: 02549-30492	17.07.68	Putzfrau	Englisch	12,30 €	Vollzeit	195	52
Cordula Wacker, Dublinweg 2, 48301 Darup, Tel.: 02549-34746	30.09.60	Haus-haltshilfe	Spanisch	13,80 €	Vollzeit	186	14

Falls Sie die Stundanzahl und Zulagenstunden nicht in einer eigenen Tabelle platziert haben: Welche Gründe sprechen für eine Trennung dieser beiden Attribute aus der Mitarbeitertabelle und in welcher Beziehung stünde die neu geschaffene Tabelle zur Mitarbeitertabelle?

2) Führen Sie eine Normalisierung der „Vereinsverwaltung“ durch, in dem Sie:

- Entnehmen Sie der Beispieltabelle alle Attribute und erstellen Sie damit eine unnormalisierte Tabelle.
- Begründen Sie anhand der unnormalisierten Tabelle die Notwendigkeit und Vorteile der Normalisierung.
- Normalisieren Sie das Datenbankmodell bis zur 3. Normalform. Begründen Sie kurz die einzelnen Normalisierungsschritte, indem Sie angeben, welche Normalform warum verletzt ist. Geben Sie die so entstandenen Tabellen in Kurzschreibweise an. Aus der Kurzschreibweise sollen Relationsname, Attribute und Schlüsselattribute ersichtlich sein.

Musterlösung ERM:



Musterlösung Normalisierung:

