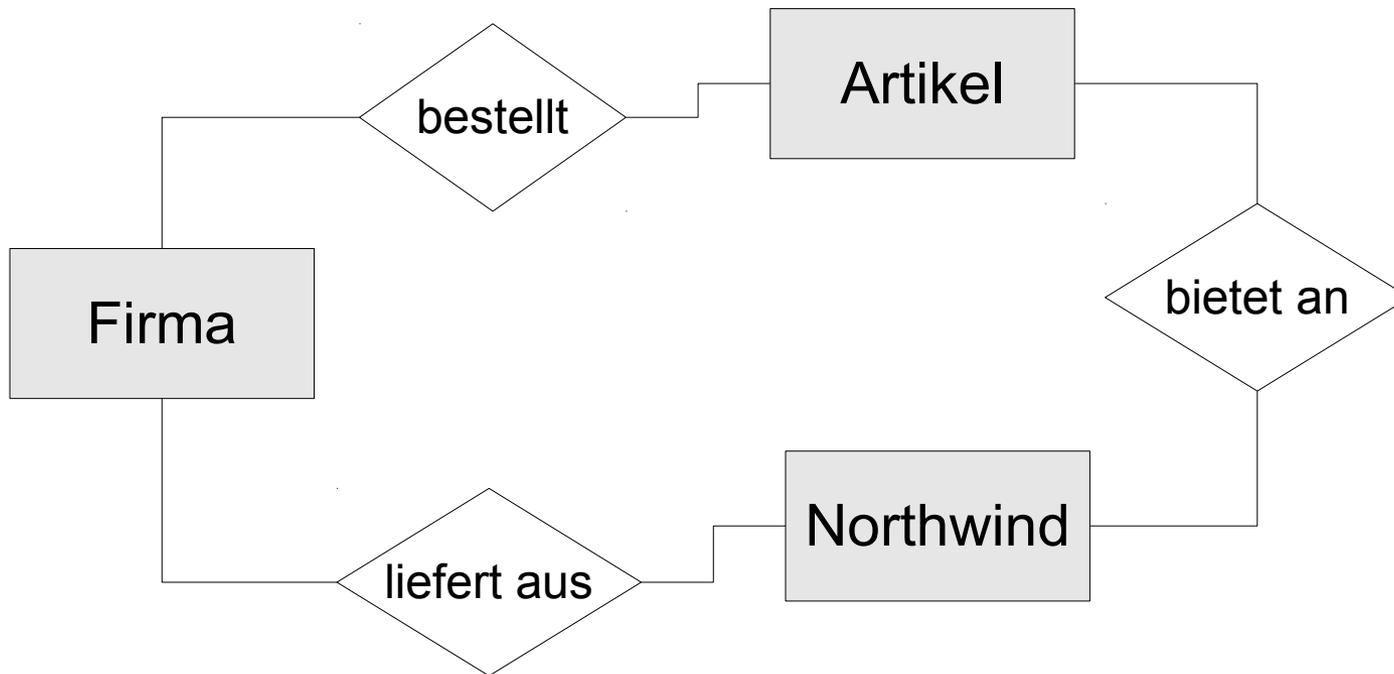


S(tructured)Q(uey)L(anguage) Entwurf einer Datenbank



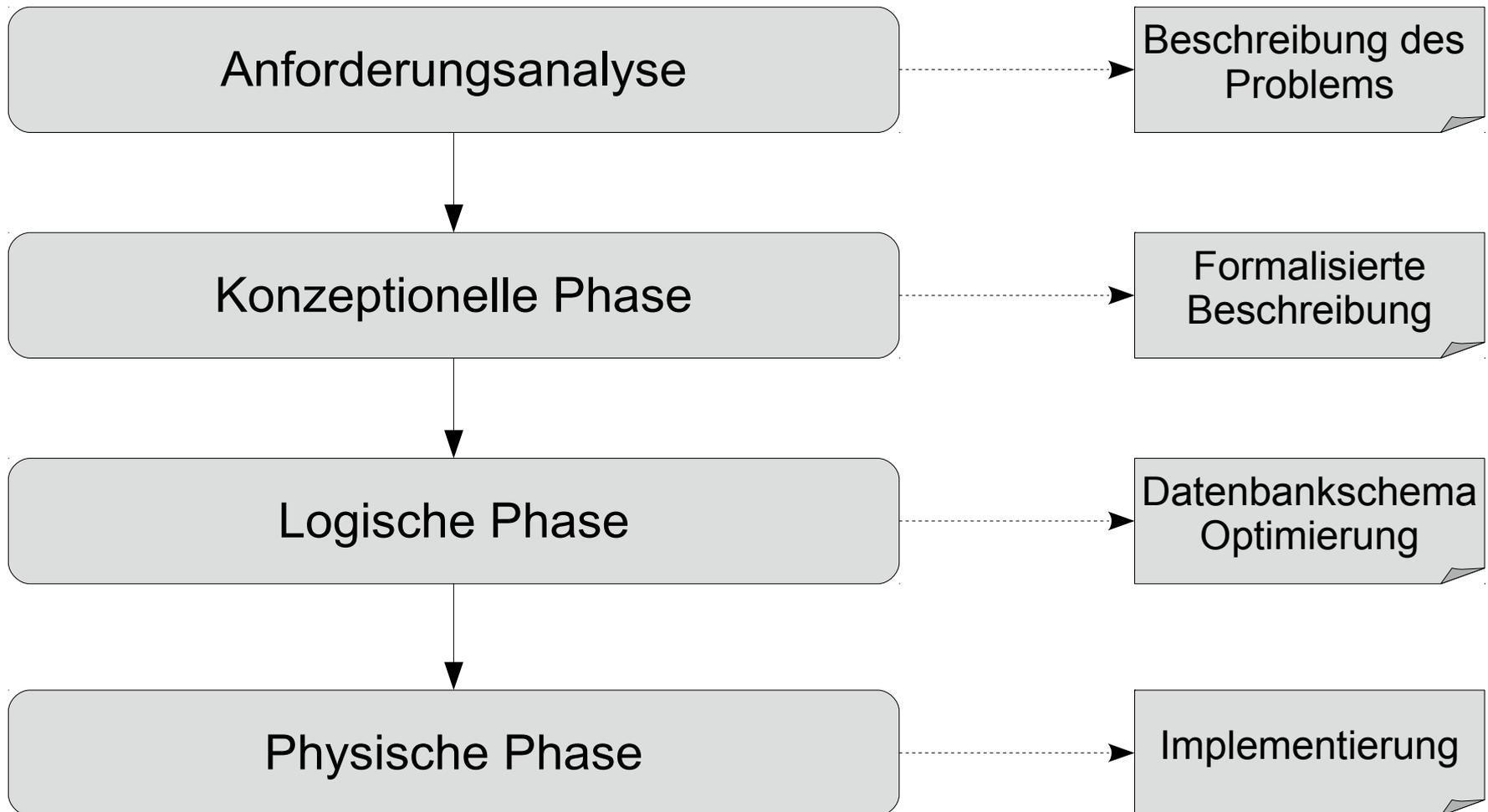
Was ist eine „Datenbank“?

- Verwaltung und Sammlung von großen Datenmengen.
- Strukturierte Ablage von Daten.
- Abbildung von Listen, bei denen in einem Tabellenkalkulationsprogramm horizontal zur Seite geblättert werden muss.

Relationales Datenbankmodell

- Entwicklung in den 70er Jahren.
- In Tabellen (Relationen) werden spaltenweise Attribute von Dingen abgelegt. Jede Zeile in einer Tabelle beschreibt ein Ding.
- Jedes Ding kann eine Beziehung zu einem anderen Ding haben. Die Beziehungen werden über Wertepaare abgebildet.
- Manipulation der Daten mit Hilfe der Abfragesprache SQL.

Entwurfsprozess einer Datenbank



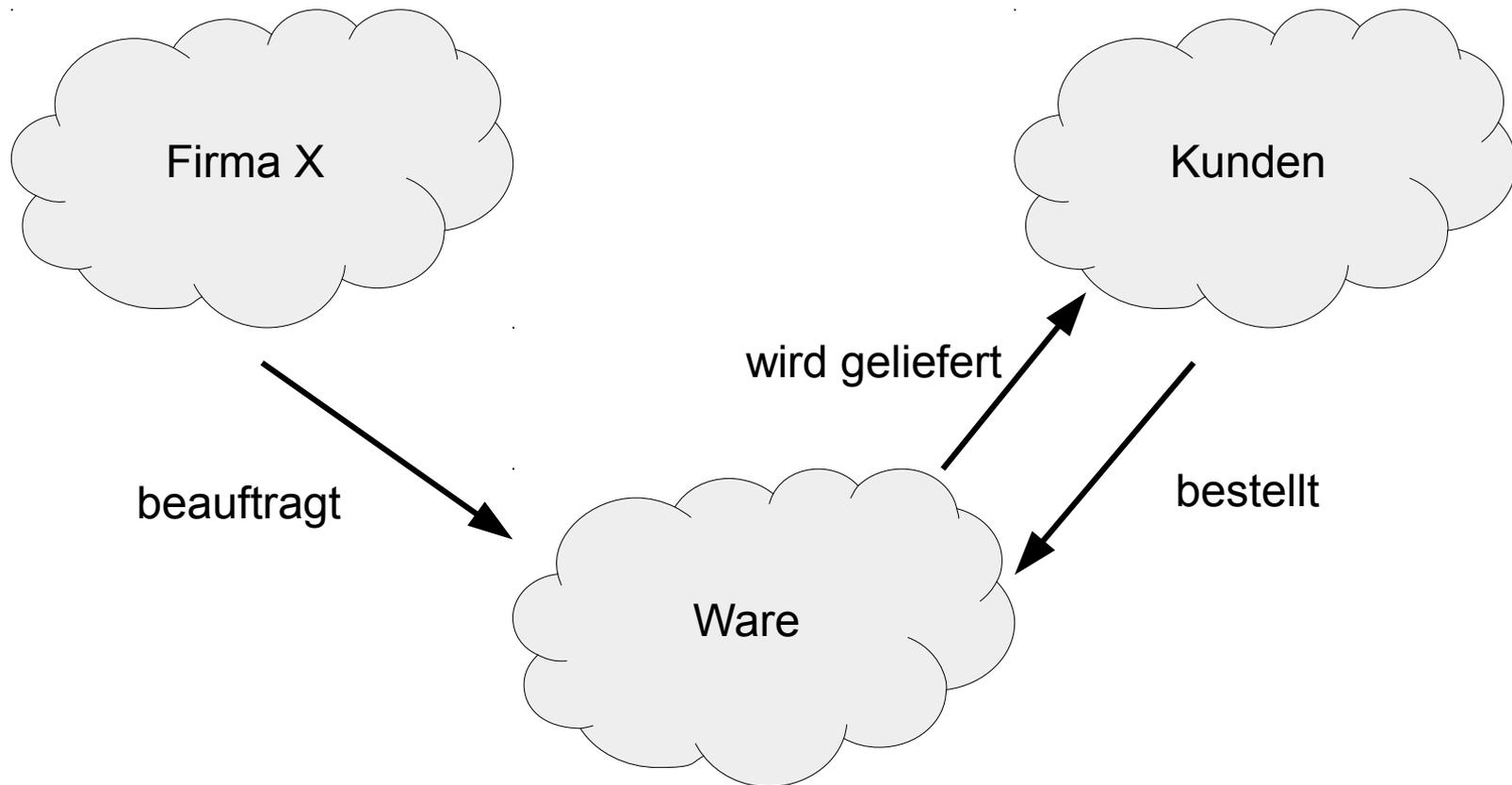
Anforderungsanalyse

- Welchen Ausschnitt aus der realen Welt soll die Datenbank abbilden?
- Welche Anforderungen muss die Datenbank erfüllen? Welche Anforderungen können aus Datenschutzgründen, technischen Voraussetzungen und so weiter nicht erfüllt werden?
- Iterativer Prozess zwischen Kunden und IT-Entwickler.
- Das Ergebnis kann mit Hilfe der Sprache UML (Unified Modeling Language) dargestellt werden. Es können Lastenhefte oder Pflichtenhefte erstellt werden.

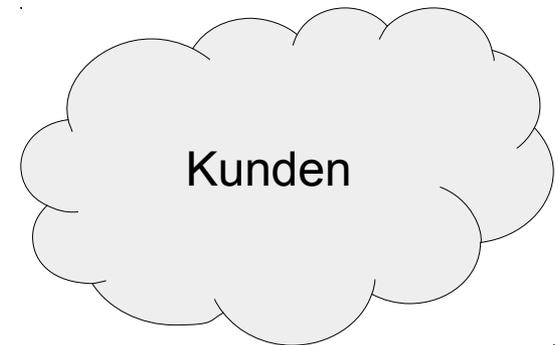
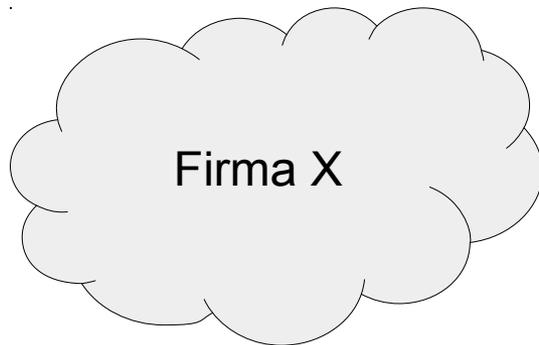
Priorisierung von Anforderungen

- **Must have.** Welche Informationen und Beziehungen müssen zwingend abgebildet werden? Informationen, die für die Funktionsfähigkeit der Datenbank unbedingt benötigt werden.
- **Should have.** Abbildung von Informationen in dem Datenbankschema, die die Akzeptanz bei den Anwendern erhöhen. Die dargestellten Beziehungen sind für die Aufgabenstellung nicht unbedingt erforderlich.
- **Could have (nice to have).** Wenn Zeit und Budget vorhanden ist, können die Informationen mit niedriger Priorität abgebildet werden.
- **Want have.** Beziehungen und Informationen, die sinnvoll sind, aber momentan aufgrund von Zeitmangel und so weiter nicht abgebildet werden.

Aufgabenstellung



Priorisierung

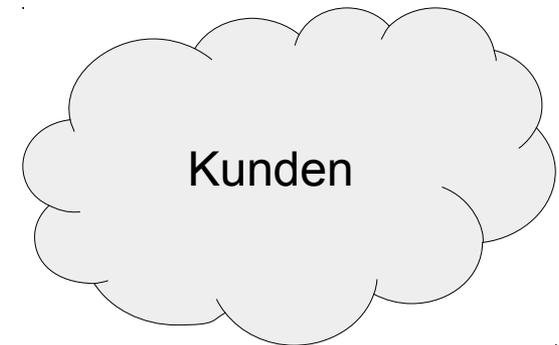
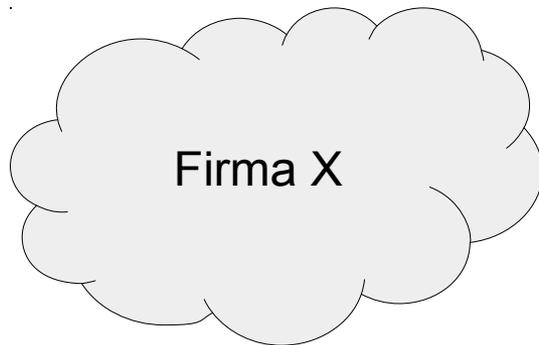


Welcher Artikel wird zu welchem Preis angeboten?	Must have	Welcher Artikel kann zu welchem Preis bestellt werden?
Automatische Bestandstransaktionen	Should have	Artikelbeschreibung zum Download
Bilder der Artikel	Could have	Online-Bestellung
Automatische Bestellung beim Lieferanten, wenn Menge x unterschritten ist	Want have	Prozess-Verfolgung vom Versand bis zur Lieferung

Konzeptionelle Phase

- Entwicklung eines Datenbankschemas unabhängig von dem später eingesetzten Datenbank-Managementsystems.
- Formalisierte Darstellung der benötigten Informationen.
- Diskussionsgrundlage zwischen Kunden und IT-Entwicklern.

Informationen zum Bestellvorgang



Wie hoch ist der Einzelpreis der Bestellposten?
Ist der bestellte Artikel in genügender Menge im Lager vorhanden?
Wohin wird die Bestellung verschickt?
Mit welcher Versandfirma kann die Bestellung verschickt werden?

Wie hoch ist der Preis des gewünschten Artikels?
Wann kann die Bestellung geliefert werden?
Wie hoch sind die Versandkosten?

Entity Relationship (ER) - Model

- Gegenstands-Beziehungsmodell nach Peter P. Chen.
- Standardmodell in dem konzeptionellen Entwurf.
- Konzept einer Datenbankstruktur.
- Visualisierung von Objekten und deren Beziehung in Bezug auf die Anforderungsanalyse.

Entity-Typ

Kunde

Artikel

Lieferanten

- Der Kunde „Müller“ bestellt den Artikel „Tomatensuppe“.
- Kategorisierende Substantive wie Kunde, Artikel etc., geben einen Hinweis auf Entity-Typen.
- Entitäten werden mit Hilfe eines Rechtecks visualisiert.

Beispiele

- Gegenstände wie zum Beispiel Buchungsrechner, Artikel, Drucker.
- Personen und Organisationen.
- Abbildung von Rollen wie zum Beispiel Besteller, Projektleiter, Ausleihender.
- Rollen von Menschen und Systemen wie zum Beispiel KFZ-Halter, Projektleiter, Ausleiher.
- Abbildung von organisatorischen Einheiten wie Bezirk, Verkaufsgebiet, Region.

Entity (Entität)

Kunde

Artikel

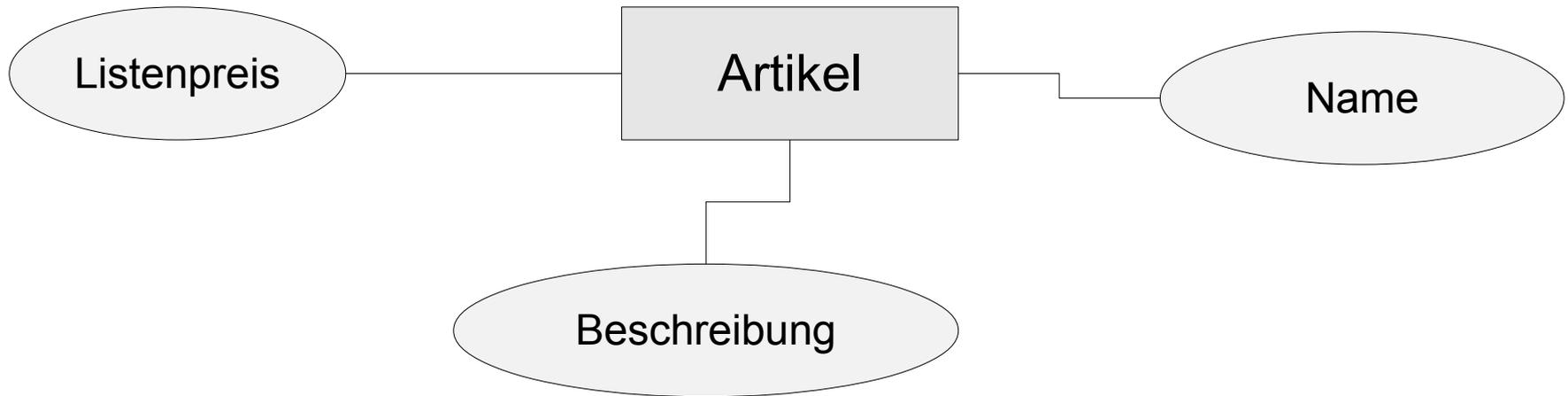
Lieferanten

- „Müller“ ist ein Kunde. „Müller“ ist eine Entität, die allgemein mit Hilfe des Entity-Typs „Kunde“ beschrieben wird.
- Entitäten beschreiben konkret ein Objekt aus der realen Welt.
- Jede Entität unterscheidet sich in mindestens einem Attribut von allen anderen Elementen des Entity-Typs.

... in einer Datenbank

- Entitäten beschreiben ein Objekt oder Gegenstand aus der realen Welt.
- Entitäten mit gleichen Attributen werden in Tabellen zusammengefasst. Jedes Rechteck in einem ER-Modell stellt eine Tabelle dar.
- In einer Tabelle wird zeilenweise ein Datensatz pro Entität abgelegt.

Attribute



- Jeder Entity-Typ hat Attribute, die eine Entität kennzeichnen.
- Mit Hilfe der Eigenschaft wird ein Objekt beschrieben,
- Darstellung als Ellipse.

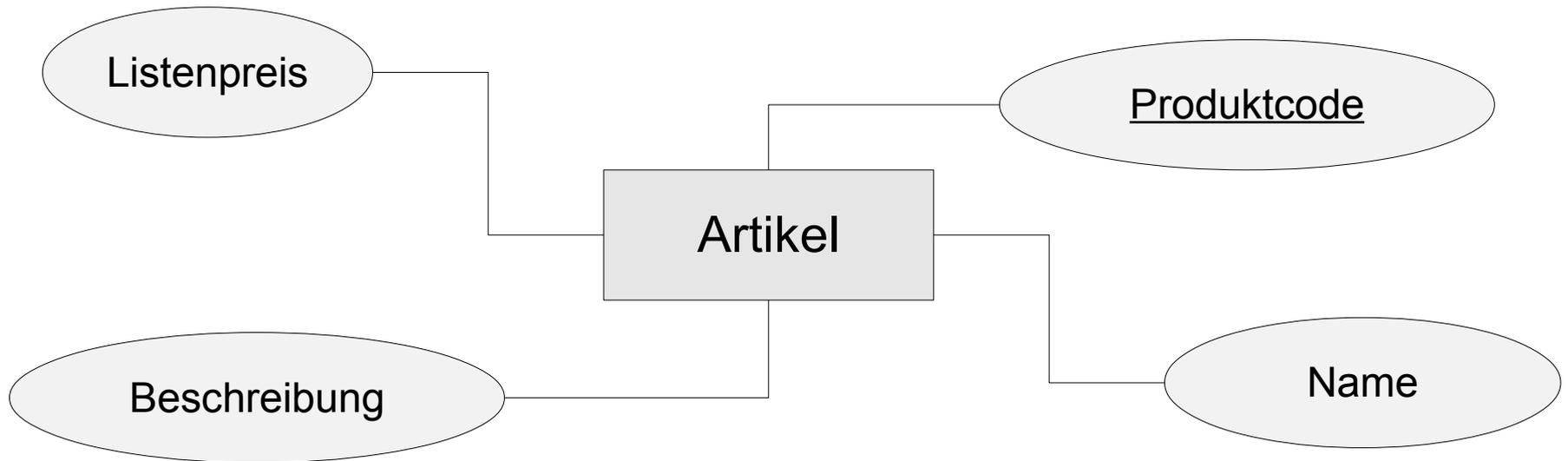
... in einer Datenbank

- Pro Spalte einer Tabelle wird ein Attribut einer Entität beschrieben.
- Datenfelder in einem Datensatz enthalten den Wert einer Eigenschaft.
- Jeder Datensatz unterscheidet sich mindestens in einem Attribut-Wert von allen anderen Datensätzen in einer Tabelle.

Identifizierende Attribute

- Ein Schlüssel kennzeichnet eine Entität von einem Typ eindeutig.
- Ein Schlüssel kann sich aus mehreren Attributen zusammensetzen.
- Identifizierende Attribute werden unterstrichen.

... in einem ER-Modell



... in der realen Welt

- Ware A wird erstmalig in den Warenkorb aufgenommen. Ware A bekommt eine Artikelnummer.
- Kunde X bestellt ... Durch das Auslösen des Bestellvorgangs wird automatisiert eine Bestellnummer erzeugt.
- A möchte erstmalig ein Buch ausleihen. A bekommt ein Bibliotheksausweis. Der Ausweis enthält eine Mitgliedsnummer.
- Die Bestellung wird an das Haus ... ausgeliefert. Das Haus wird durch die Straße, Hausnummer und Postleitzahl gekennzeichnet. Eine weitere Möglichkeit sind die GPS-Daten eines Hauses.

... finden

- Attribute, die bei der Erzeugung einer Entität oder bei Anlage einer Vorgangs einmal angelegt werden und nie wieder verändert werden.
- Künstliche Attribute oder Aktenzeichen werden als Schlüssel genutzt. Die Entitäten existieren ohne diese Kennzeichnung in der realen Welt.
- Substantive, die auf „Nummer“ enden. Zum Beispiel Auftragsnummer, Kundennummer, IBAN-Nummber, ISBN-Nummer etc.

... in einer Datenbank

- Schlüsselwerte in einer Datenbank kennzeichnen einen Datensatz in einer Tabelle eindeutig.
- Jeder Wert in der Spalte „Schlüssel“ kommt nur einmal in der Tabelle vor.
- Die Schlüsselwerte werden als Primary Key (Primärschlüssel) bezeichnet.

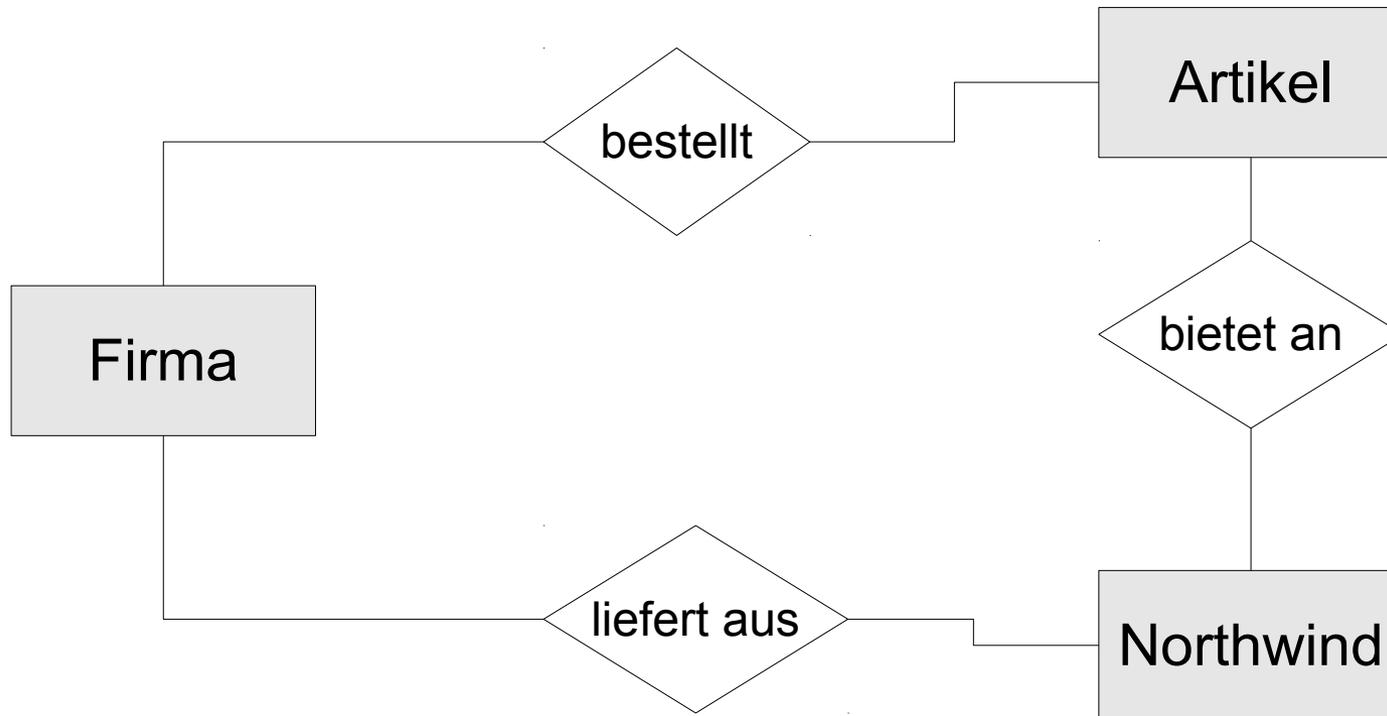
Beziehungen (Relationship)

- Verknüpfungen zwischen Entitäten.
- Abbildung von Aktionen wie zum Beispiel „Lieferant A liefert Ware X“ oder „Person A leiht Buch X aus“.
- Die Anzahl der Beteiligten an einer Beziehung kann beschrieben werden.

... in einem ER-Modell

- Raute, die zwei Entity-Typen verbindet.
- Mit Hilfe eines Verbs in der Raute kann die Beziehung beschrieben werden.
- An den Enden der Verbindungslinien kann die Anzahl der Beteiligten an der Beziehung definiert werden.
- An den Enden der Verbindungslinie kann die Rolle der Entität beschrieben werden.

Beispiel



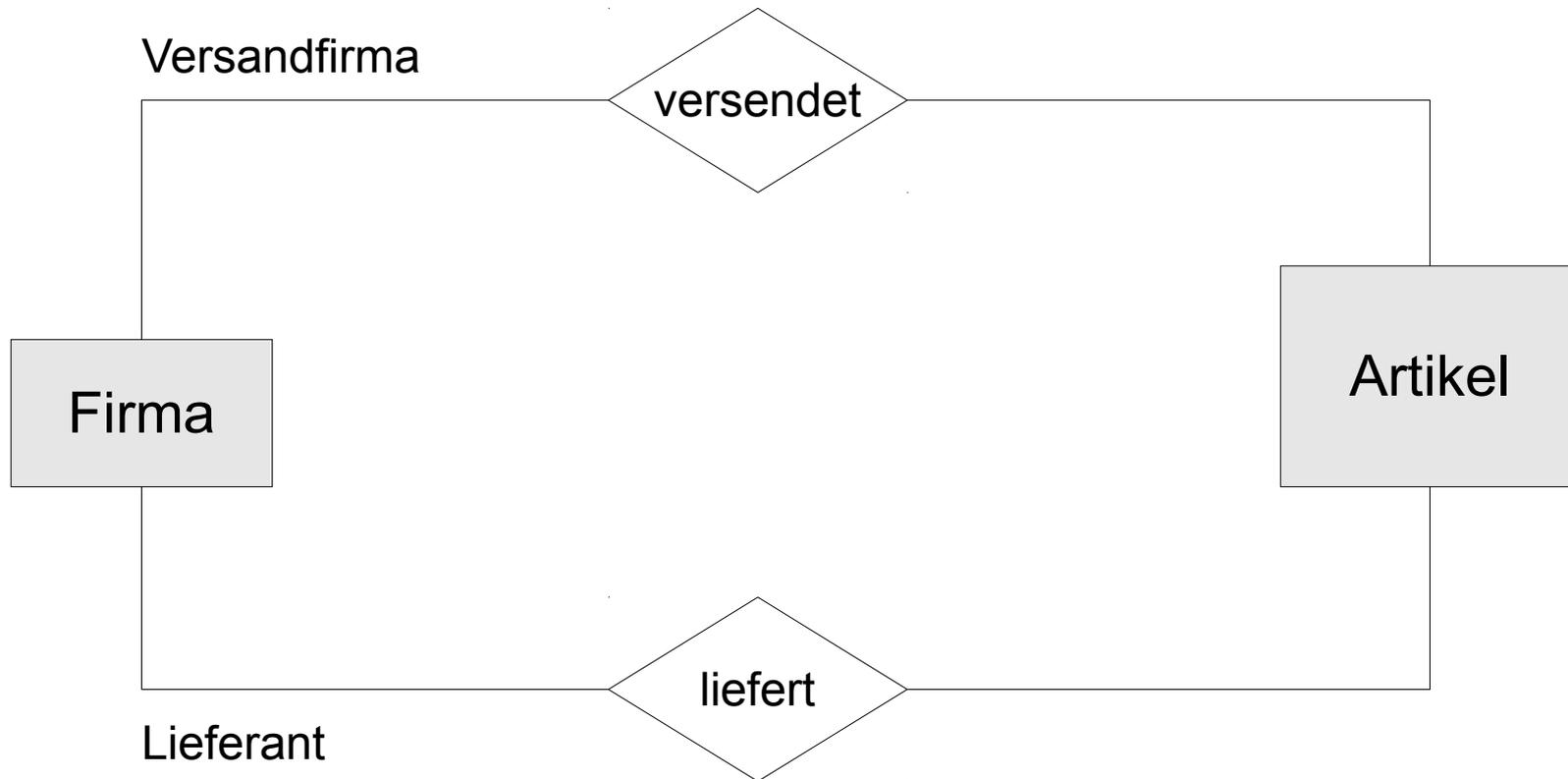
... in einer Datenbank

- Nutzung von Primärschlüssel – Fremdschlüssel in Tabellen.
- Die Tabelle Lieferant hat den Primärschlüssel Lieferanten-Nummer. In der Tabelle Auftrag wird dieser Schlüssel genutzt, um einem Auftrag eindeutig einem Lieferanten zu zuweisen. Der Auftrag selber hat den Primärschlüssel Auftragsnummer.

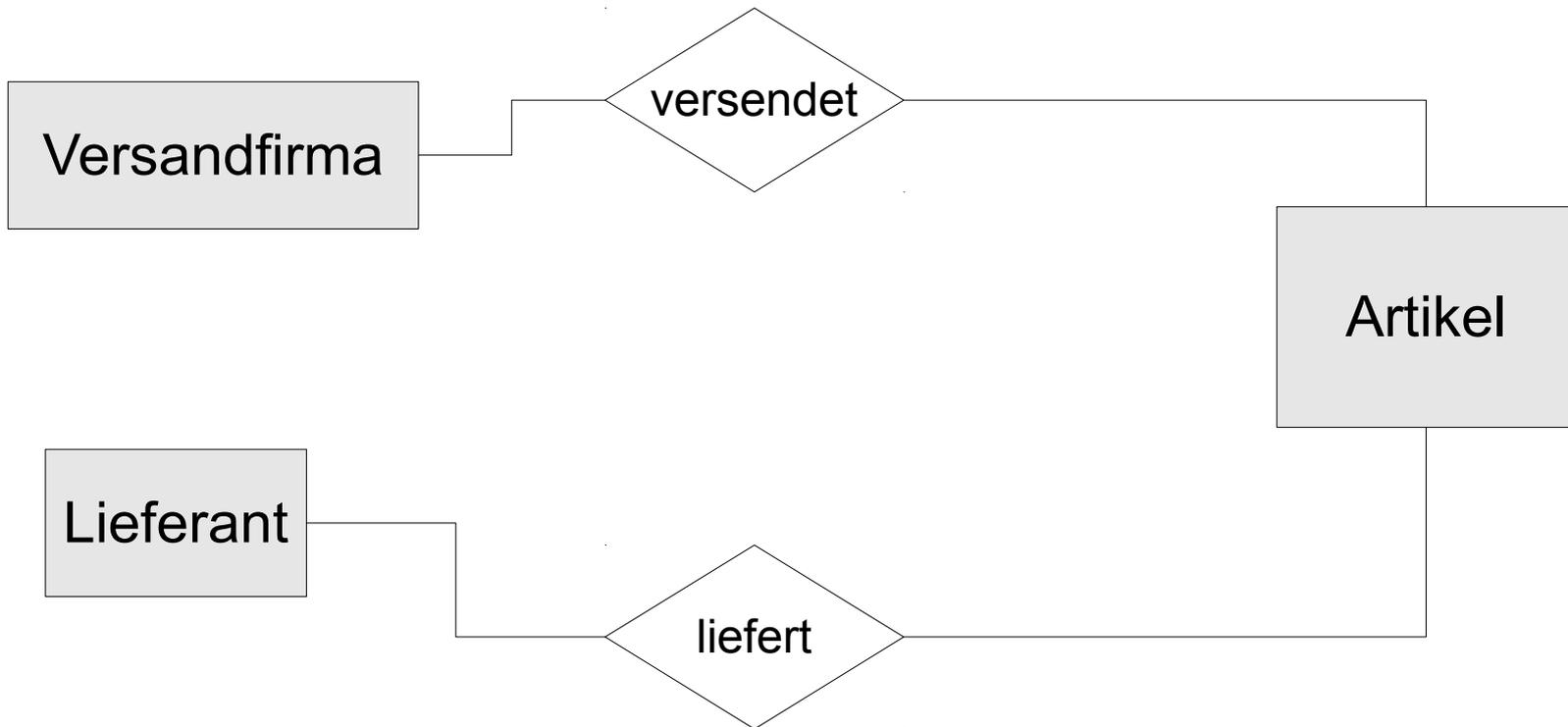
Rollen

- Beschreibung von Objektgruppen.
- Generalisierung von Objektgruppen.
- Darstellung eines Entity-Typs.

... in einem ER-Modell



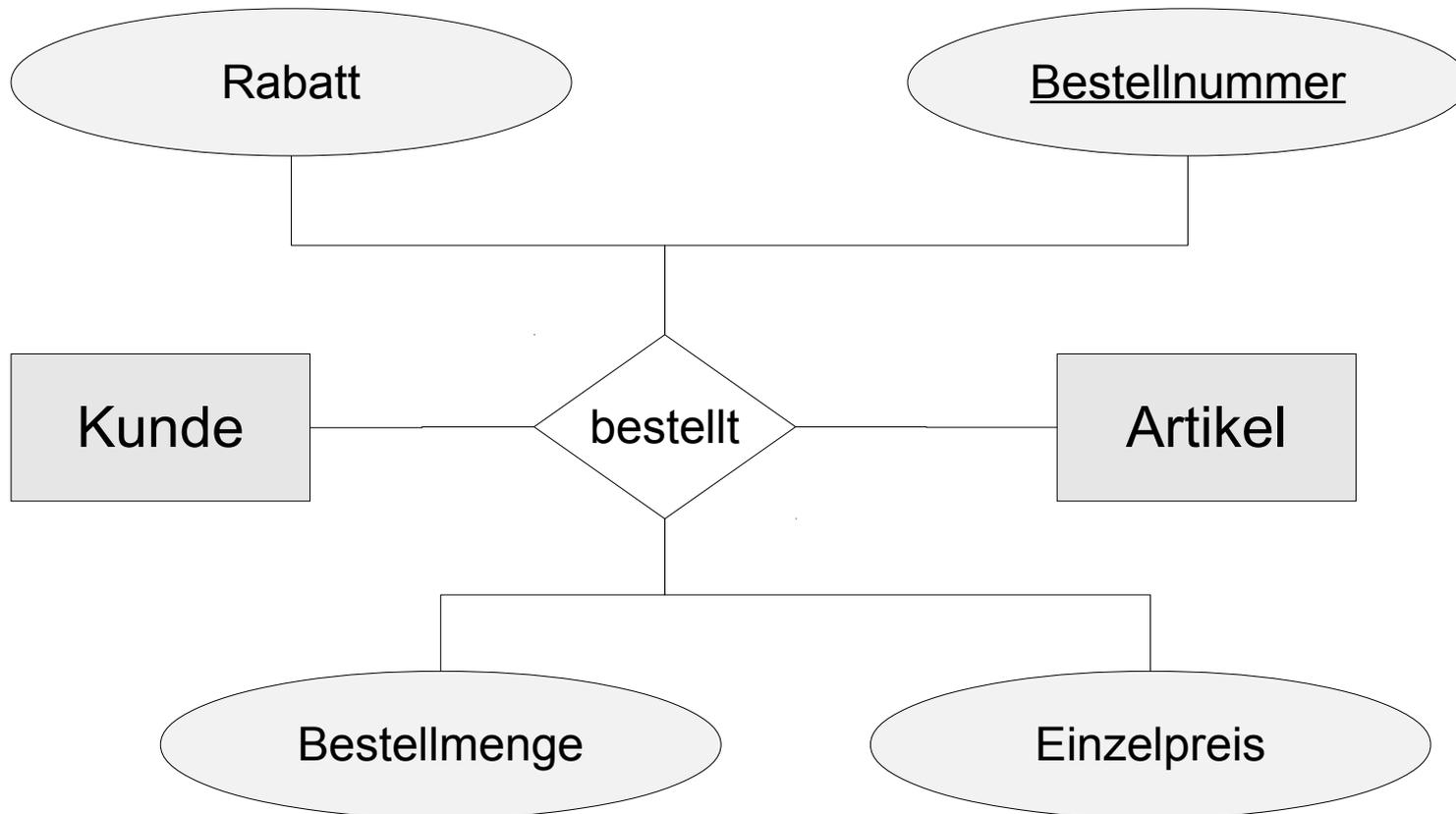
... sollte in dieses Modell umgesetzt werden



Attribute einer Beziehung

- In selten Fällen hat die Beziehung selber Attribute.
- Darstellung mit Hilfe von Ellipsen.
- Die Beziehung wird in einer gesonderten Tabelle in der Datenbank abgebildet.

... in einem ER-Modell

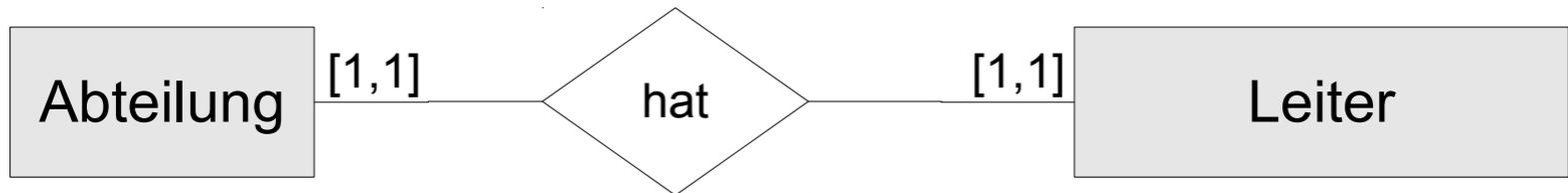


Kardinalitäten

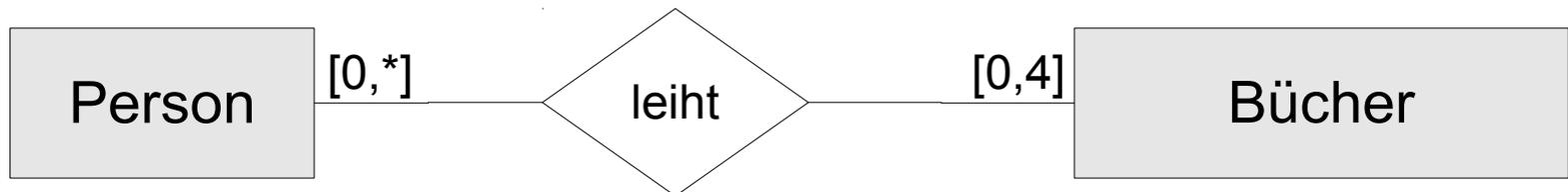
- Abbildung des Beziehungstyps aus Sicht einer Entität.
- Wie viele Entitäten sind minimal und maximal an einer Beziehung beteiligt?
- Wie oft darf der Fremdschlüssel in einer Tabelle genutzt werden?

... in einem ER-Modell

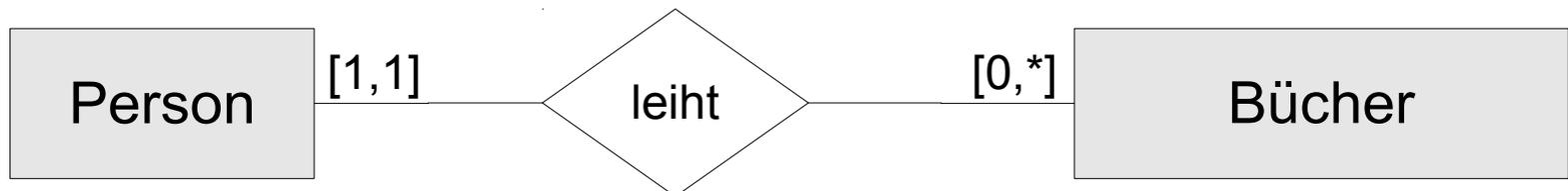
- Jede Abteilung einer Firma hat einen Leiter.



- Personen können maximal 4 Bücher ausleihen.

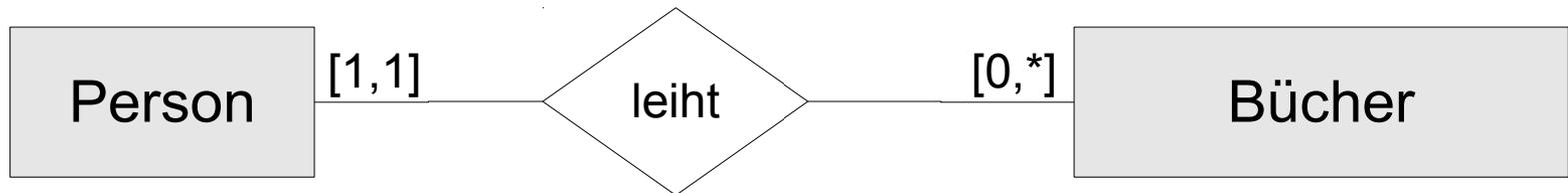


- Person A leiht beliebig viele Bücher aus.

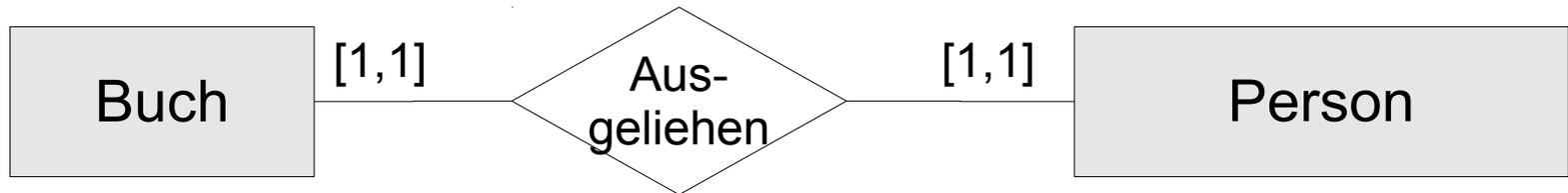


... aus Sicht einer Entität

- Person A leiht beliebig viele Bücher aus.



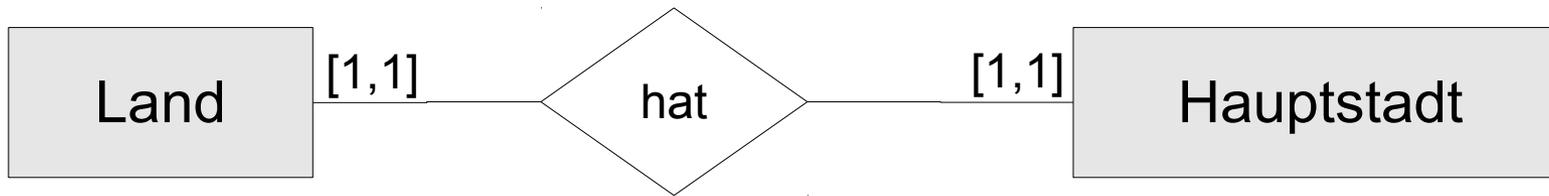
- Ein Buch wird von einer Person ausgeliehen.



Angabe von Kardinalitäten

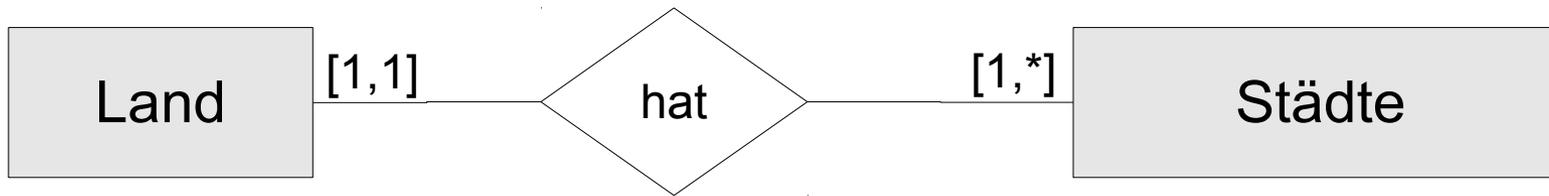
- $[\text{min}, \text{max}]$. Die minimale und maximale Anzahl der Beteiligten an eine Beziehung wird angegeben. Die Angaben in den eckigen Klammern werden durch ein Komma getrennt.
- $[0,4]$. Die häufigsten Werte für das Minimum sind 0 (keiner ist beteiligt) oder 1 (einer ist beteiligt).
- $[1,1]$. Wenn beide Werte gleich sind, wird häufig nur die maximale Anzahl angegeben.
- $[0,*]$. Das Sternchen ist ein Platzhalter für eine beliebige Anzahl von Beteiligten.

1 : 1 - Beziehung



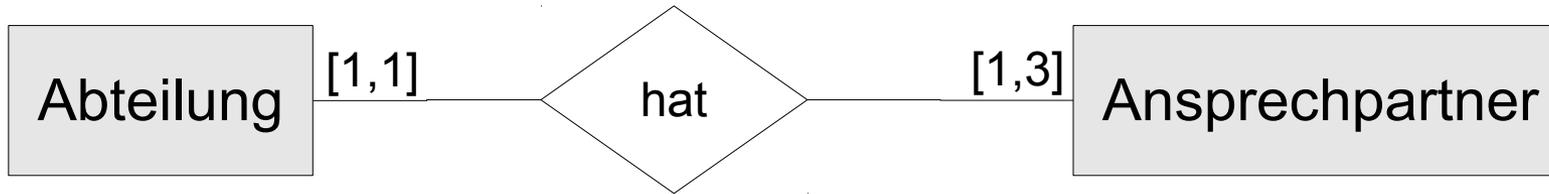
- Einer Entität kann exakt eine andere Entität zugeordnet werden.
- Jedes Land hat eine Hauptstadt. Die Hauptstadt kann exakt einem Land zugeordnet werden.
- Die Entität Hauptstadt ist ein Attribut der Entität Land.

1 : n - Beziehung



- Häufigste Beziehung.
- Ein Land hat beliebig viele Städte. Die Städte Berlin, Hannover, Hamburg, Celle liegen in Deutschland. Aber eine Stadt liegt in einem Land. Die Stadt Berlin liegt in Deutschland. Städte wie Frankfurt / Oder oder Görlitz, die in zwei Ländern liegen können nicht abgebildet werden.

Angabe des maximalen Wertes



- Eine Abteilung hat einen, maximal drei Abteilungsleiter.
- Einem Abteilungsleiter kann eine Abteilung zugeordnet werden.

m : n - Beziehung

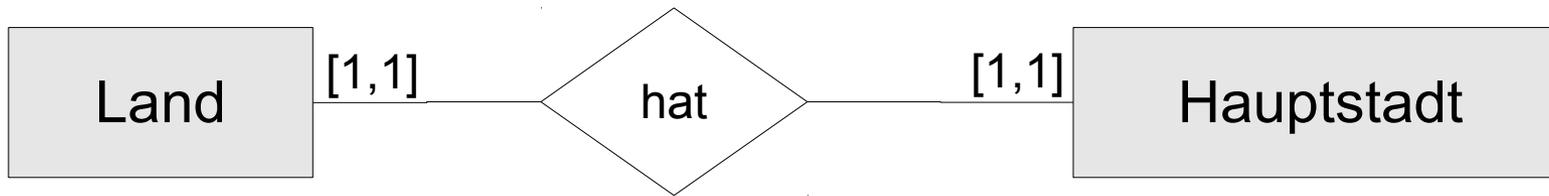


- Eine oder beliebig viele Personen können an keinem Kurs oder beliebig vielen Kursen teilnehmen.
- Ein Kurs hat keinen oder beliebig viele Teilnehmer.
- Jeder Entität aus der Gruppe A sind beliebig viele Datensätze aus der Gruppe B zugeordnet. Aber jeder Entität aus der Gruppe B sind beliebig viele Entitäten aus der Gruppe A zugeordnet.

Logische Phase

- Transformation des ER-Modells in ein relationales Datenbankmodell.
- Auswahl des Datenbank-Managementsystems. Entsprechend des System werden die Typen der Datenfelder festgelegt.
- Optimierung des Entwurfs mit Hilfe der Normalisierung.

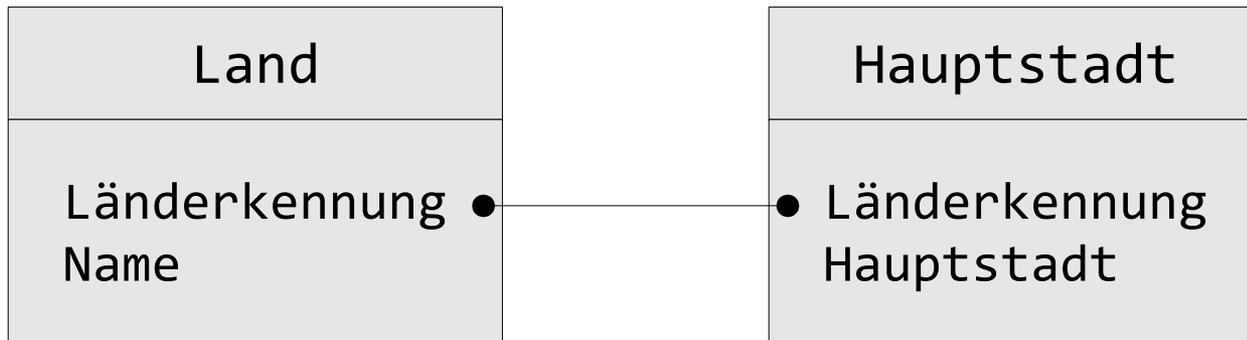
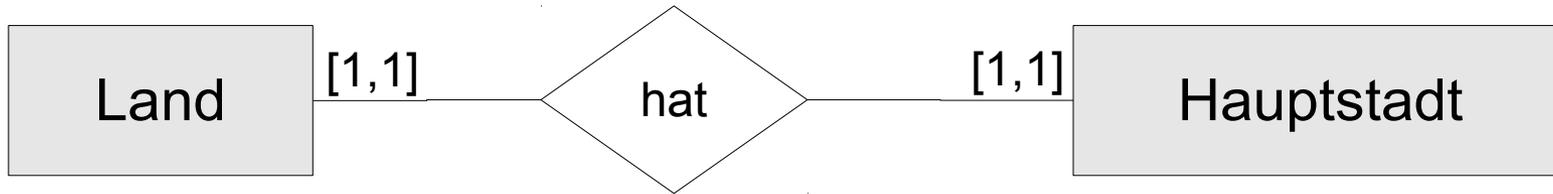
1 : 1 - Beziehung



Hinweise zur Umsetzung

- Zu einem Datensatz existiert exakt ein Datensatz in einer anderen Tabelle.
- Die beiden Tabellen können in eine zusammengeführt werden.

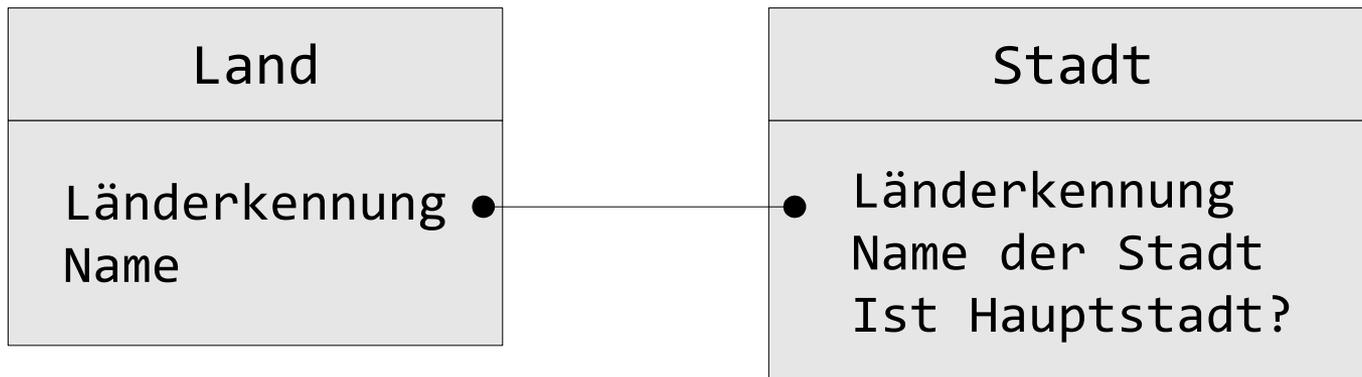
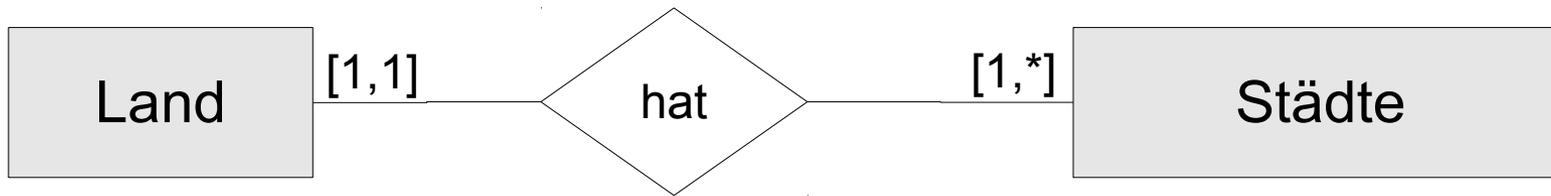
... trennen



Gründe für die Trennung der Tabellen

- Die Anzahl der Datenfelder kann nicht auf einer Bildschirmseite dargestellt werden.
- Einige Attribute werden nur von wenigen Entitäten genutzt. Die von allen Entitäten am häufigsten Attribute werden in einer Tabelle zusammengefasst. Alle anderen Attribute werden ausgelagert.
- Daten werden vor den Zugriff einer bestimmten Personengruppe geschützt. Beispiel: Alle Mitarbeiter dürfen die Kontaktdaten einsehen. Aber nur die Personalabteilung sieht die Gehaltsstufe und -höhe.

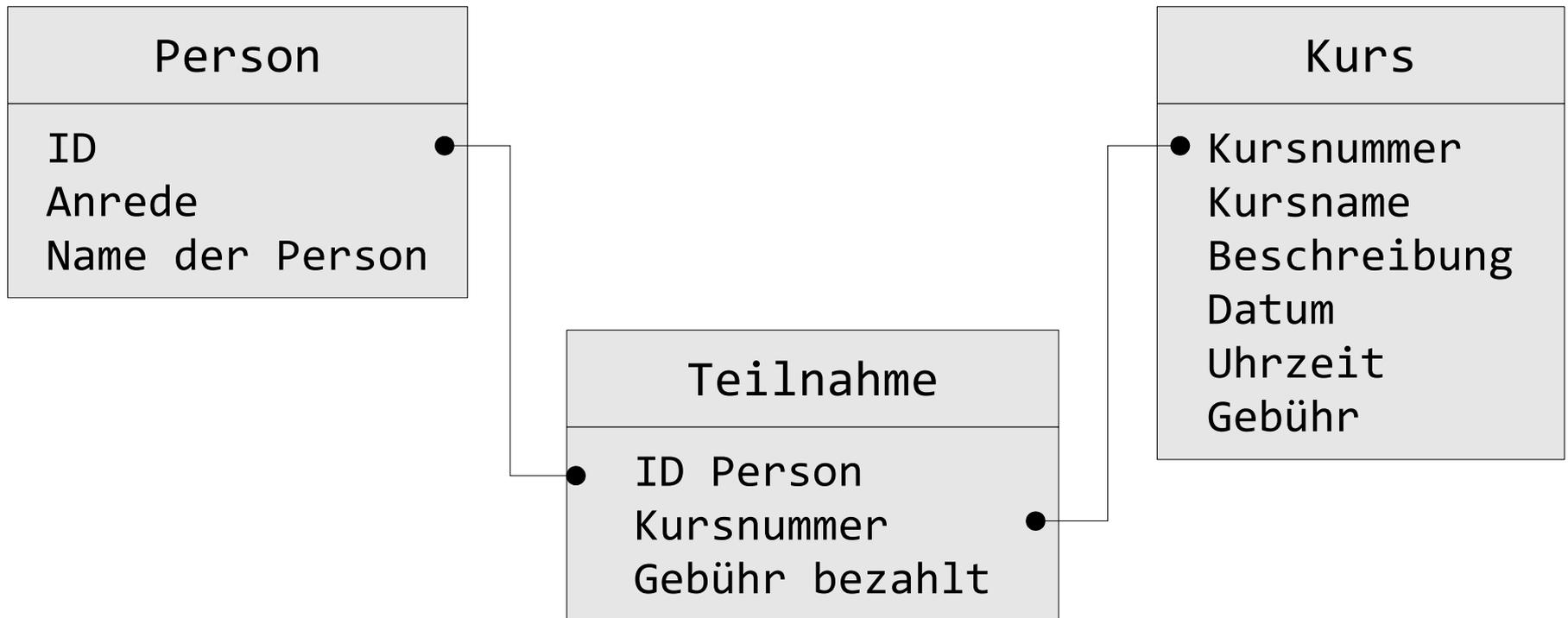
1 : n - Beziehung



Hinweise zur Umsetzung

- Zu einem Datensatz in einer Tabelle existieren beliebig viele Datensätze in einer anderen Tabelle.
- Der Primärschlüssel in der Tabelle der „1-Seite“ wird als Fremdschlüssel in der Tabelle der „n-Seite“ übernommen.
- Eine Einschränkung von „beliebig viele“ auf einen maximalen Wert kann nur mit Hilfe von Code etc. überprüft werden.

m : n - Beziehung



Hinweise zur Umsetzung

- Für jeden Entity-Typ wird eine Tabelle angelegt.
- Die Beziehung selber wird in einer Pseudo-Tabelle dargestellt. In dieser Tabelle wird jeder Beteiligte an der Beziehung durch ein Schlüsselwert symbolisiert. Jeder Schlüsselwert in der Pseudo-Tabelle hat ein Gegenstück in einer der Tabellen, die eine Entity darstellt.
- In der Pseudo-Tabelle werden Eigenschaften der Beziehungen spaltenweise definiert.

Datentypen von Feldern

- Datentypen legen den Speicherbedarf und den Wertebereich des Attribut-Wertes fest.
- Datentypen beeinflussen die Weiterverarbeitung der gespeicherten Attribut-Werte.
- Datentypen können in Abhängigkeit des Datenbank-Managementsystems einem Datenfeld zugeordnet werden.

... angeben

Person

```
ID : integer  
Anrede : varchar  
Name der Person : varchar
```

Teilnahme

```
ID Person : integer  
Kursnummer : varchar  
Gebühr bezahlt : boolean
```

Kurs

```
Kursnummer : varchar  
Kursname : varchar  
Beschreibung : varchar  
Datum : date  
Uhrzeit : time  
Gebühr : numeric
```

Möglichkeiten

- Zeichenketten aus numerischen und alphanumerischen Zeichen: `varchar` **oder** `character`.
- Ganzzahlen: `integer`, `smallint` **oder** `bigint`.
- Gleitkommazahlen: `float`, `real` **oder** `double precision`.
- Zahlen, die in kaufmännischen Berechnungen genutzt werden: `numeric` **oder** `decimal`.
- Datums- und Zeitangaben: `time`, `date` **und** `timestamp`.
- Ja / Nein: `boolean`.

Hinweise

- Einige Hersteller bieten Datentypen an, die nicht im Standard-SQL definiert sind.
- Die Nutzung der Daten hilft häufig bei der Auswahl des Datentyps.

Normalisierung von Daten

- Optimierung von Tabellen durch Zerlegung.
- Vermeidung von Fehlern durch widersprüchlich gespeicherte Daten.
- Daten werden nicht redundant in der Datenbank abgelegt.
- In einer Tabelle existieren keine Spaltengruppen.

Spaltengruppen in Tabellen

- Beispiel: Zu jedem Kunden kann keine oder maximal drei Telefonnummern abgelegt. Die Spaltengruppe Telefon, besteht aus drei Datenfeldern. In der Tabelle Kunde werden in drei verschiedenen Datenfeldern Telefonnummer abgelegt.
- Nachteil von Spaltengruppen: Alle drei Datenfelder enthalten den gleichen Typ von Information. Die Tabelle wächst in die Breite.

Vermeidung von Anomalien bei der Eingabe

- Beispiel: Bei jedem Auftrag werden die Angaben zu dem Lieferanten vollständig gespeichert.
- Die Firma Johansen KG wird nochmals beauftragt. Als Lieferant wird die Firma Johansen eingetragen. Die Angabe KG wird vergessen.
- Aus der Sicht der Datenbank ist der Auftrag an zwei verschiedene Lieferanten heraus gegangen.

Vermeidung von Anomalien bei Änderung

- Beispiel: Bei jedem Auftrag werden die Angaben zu dem Lieferanten vollständig gespeichert.
- Die Firma Johansen KG ist von Hannover nach Braunschweig gezogen. Bei der Eingabe der Änderung wird die Adressänderung nicht bei allen Aufträgen durchgeführt.
- Aus der Sicht der Datenbank ist der Auftrag an zwei verschiedene Lieferanten heraus gegangen.

Vermeidung von Anomalien bei Löschung

- Beispiel: Bei jedem Auftrag werden die Angaben zu dem Lieferanten vollständig gespeichert.
- Aufträge, die älter als fünf Jahre sind, werden gelöscht. Wenn ein Lieferant, die letzten fünf Jahre keinen Auftrag erhalten hat, werden alle Informationen zu diesem Lieferanten gelöscht.
- Die Firma Johansen KG stellt den Betrieb ein. Alle Aufträge der Firma werden gelöscht. Das gesamte Auftragsvolumen des Auftraggebers wird nicht korrekt berechnet werden.

Auflösen von funktionalen Abhängigkeiten

- Länderkennung → Ländername. Die Länderkennung definiert eindeutig ein Land. Über die Länderkennung kann der Name eines Landes eindeutig ermittelt werden.
- Lieferanten-Nummer → Lieferanten-Name. Sobald die Lieferanten-Nummer bekannt ist, kann der Lieferanten-Name mit Hilfe einer SQL-Anweisung ermittelt werden.
- Straße, Hausnummer, Ort → Postleitzahl (in Deutschland). Bei der Auflösung dieser funktionalen Abhängigkeit muss der Entwickler Aufwand und Nutzen abwägen.

Nullte Normalform (Rohdaten)

Aufgenommen von	Janet Leverling	Robert King
Bestelldatum	07.04.2006	07.04.2006
Versanddatum	07.04.2006	08.04.2006
Lieferung zu	Cactus Comidas Cerrito 333 Buenos Aires	Cactus Comidas Cerrito 333 Buenos Aires
Bestellter Artikel	50 x Craeb Meat	50 x Clam Chowder

Erste Normalform

- Alle Attribute sind atomar. Jedes Attribut hat einen Attributwert.
- Eine Liste von Werten in einem Datenfeld wird durch Hinzufügung von Spalten aufgelöst.
- Kein Datenfeld ist von einem anderem Datenfeld funktional abhängig.

Beispiel

Aufgenommen von	Janet Leverling	Robert King
Bestelldatum	07.04.2006	07.04.2006
Versanddatum	07.04.2006	08.04.2006
Lieferung zu	Cactus Comidas	Cactus Comidas
Lieferung Straße	Cerrito 333	Cerrito 333
Lieferung Stadt	Buenos Aires	Buenos Aires
Bestellter Artikel	Craeb Meat	Clam Chowder
Bestellmenge	50	50

2. Normalform

- Die Tabelle befindet sich in der ersten Normalform.
- Jedes Datenfeld, welches kein Schlüsselfeld ist, ist nur vom Schlüsselfeld voll funktional abhängig.
- Zusammenfassung von Daten, die ein Objekt beschreiben. Für jede Objektgruppe wird eine Tabelle angelegt.

Beispiel „Aufgenommen von“

Aufgenommen von	Janet Leverling	Robert King
-----------------	-----------------	-------------

Mitarbeiter
Mitarbeiter-Nr. Vorname Nachname

Beispiel „Ware“

Bestellter Artikel	Craeb Meat	Clam Chowder
--------------------	------------	--------------

Ware
ArtikeInr. Name

Beispiel „Lieferung zu“

Lieferung zu	Cactus Comidas	Cactus Comidas
Lieferung Straße	Cerrito 333	Cerrito 333
Lieferung Stadt	Buenos Aires	Buenos Aires

Kunde
Kundenr. Name Straße Stadt

Beispiel „Bestellung“

Aufgenommen von	Janet Leverling
Bestelldatum	07.04.2006
Versanddatum	07.04.2006
Lieferung zu	Cactus Comidas
Lieferung Straße	Cerrito 333
Lieferung Stadt	Buenos Aires
Bestellter Artikel	Craeb Meat
Bestellmenge	50

Bestellung
Bestellnr.
Kundenr.
Artikelnr.
Menge
Bestellt am
Versandt am
ID Mitarbeiter

Dritte Normalform

- Die zweite Normalform ist erfüllt.
- Jedes Datenfeld, welches kein Schlüsselfeld ist, ist nicht transitiv vom Primärschlüssel abhängig.
- Aus keinem Nichtschlüsselattribut folgt kein anderes Nichtschlüsselattribut.

Beispiel „Lieferung zu“

Lieferung zu	Cactus Comidas	Cactus Comidas
Lieferung Straße	Cerrito 333	Cerrito 333
Lieferung Stadt	Buenos Aires	Buenos Aires

Kunde
Kundenr. Name Straße Hausnummer Stadt

- Die Straße muss nicht geändert werden, wenn die Hausnummer geändert wird und umgekehrt.

Beispiel „Mehrere Lieferadressen“

Lieferung zu	Cactus Comidas	Cactus Comidas
Lieferung Straße	Cerrito 333	Cerrito 333
Lieferung Stadt	Buenos Aires	Buenos Aires



Physische Phase

- Implementierung einer leeren Datenbank mit Hilfe eines Datenbank-Managementsystems.
- Umsetzung des Modells aus der logischen Phase mit Hilfe der Datendefinitionssprache SQL.
- Festlegung von Zugriffsrechten auf die Datenbank.