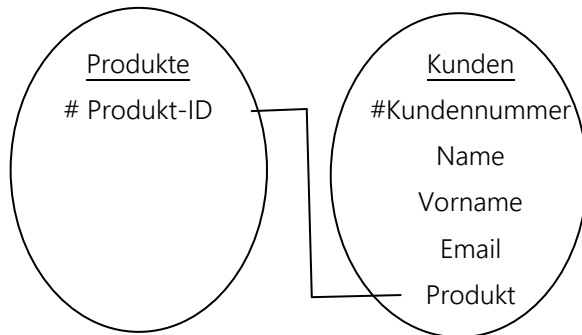


Lösungsvorschlag Modellfabrik



1. Aufgabe :

Erster Ansatz :



(Auf weitere, in der Realität sicher nötige, Attribute der Produkte, wie Preis, Gewicht, usw. wird verzichtet)

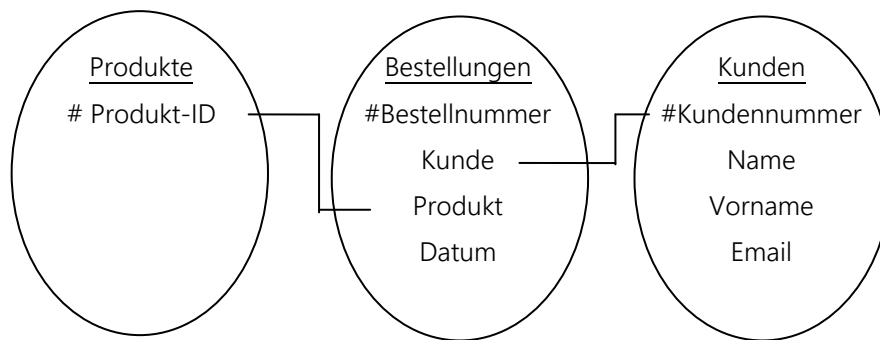
Der ER-Test ist schnell gemacht :

Produkt->Kunde : Ein Produkt kommt in der Kundenliste öfter vor -> n

Kunde->Produkt : Ein Kunde kann mehrere Produkte kaufen -> m

n:m – Kardinalität, also eine Zwischentable !

Modifizierter Ansatz :



Wieder ist der ER-Test schnell gemacht :

Produkte->Aufträge : (nochmal „umständlich“, ausführlich formuliert)

Kommt ein Element aus Produkte in der table Aufträge einmal oder öfter vor ?

Ein Produkt ist Bestandteil in mehreren Aufträge ! -> n !

Aufträge->Produkte :

Kommt ein Element aus Aufträge in Produkte einmal oder öfter vor ?

Eine Zeile in Aufträge („ein Element“) beinhaltet nur jeweils 1 Produkt ! -> 1 !

Bei den beiden anderen Tests kommt genau das Gleiche raus.

Um das ein wenig anschaulicher zu machen, schadet es nicht, auch mal die wirklichen Tabellen anzuschauen :

Produkte

| |
|------------|
| Produkt-ID |
| P1 |
| P2 |
| P3 |
| |
| (usw.) |

Bestellungen

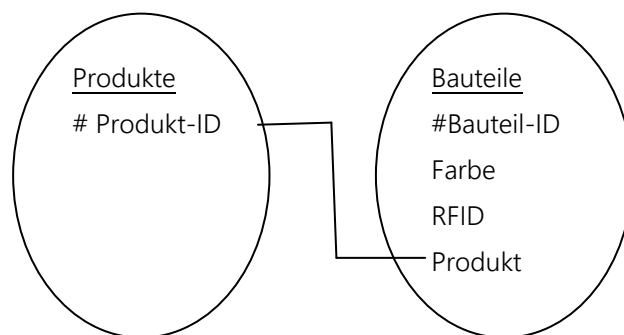
| Bestellnummer | Kunde | Produkt | Datum |
|---------------|-------|---------|---------|
| 1 | 7 | P3 | 27.3.20 |
| 2 | 12 | P3 | 12.3.20 |
| 3 | 12 | P9 | 12.3.20 |
| | | | |
| (usw..) | | | |

Kunden

| Kundennummer | Vorname | Name | e-mail |
|--------------|----------|--------|------------|
| 1 | Donald | Trump | dt@usa.gov |
| 2 | Wladimir | Putin | vp@rus.gov |
| 3 | Angela | Merkel | am@brd.gov |
| | | | |
| (usw.) | | | |

2. Aufgabe :

nochmal simpler Ansatz :



Wieder ist der ER-Test schnell gemacht :

Produkt->Bauteil : Ein Produkt besteht aus mehreren Bauteilen -> n

Bauteil->Produkt : Ein Bauteiltyp ist in verschiedenen Produkten drin -> m

n:m – Kardinalität, also eine Zwischentable !

Es gibt viele verschiedene Möglichkeiten, das zu strukturieren.

Ich schlage hier eine Lösung vor, die sich enger auf die Modellfabrik bezieht, ich hoffe, der Zusammenhang zur Realität bleibt trotzdem sichtbar.

Zunächst vereinfache ich die Fabrikfunktion, und lasse den Zulieferprozess weg.

Modul 2 kann dann entweder aus dem supply-Lager 1 (rot) oder dem Notlager 2 (schwarz) bestücken, die Zuliefersequenz entfällt.

Zum Verständnis zuerst die Tabellendarstellung :

Produkte :

| |
|------------|
| Produkt-ID |
| P1 |
| P2 |
| (usw..) |

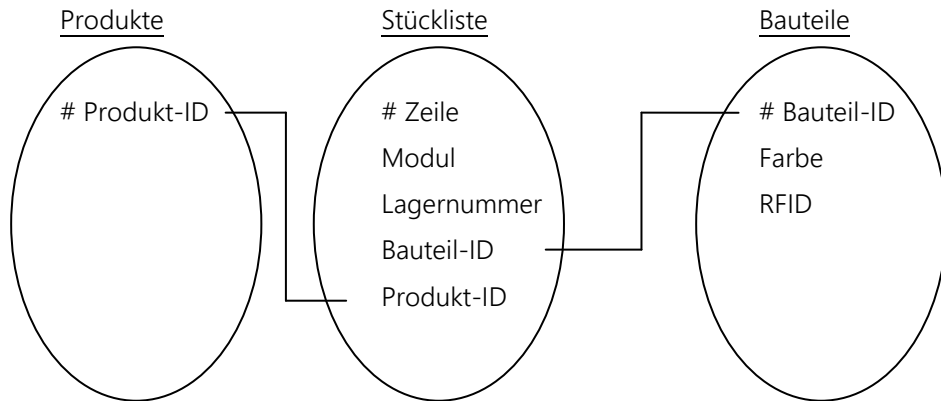
Bauteile :

| Bauteil-ID | Farbe | RFID |
|------------|-------|------|
| 1 | blau | ja |
| 2 | rot | ja |
| 3 | blau | nein |
| (usw..) | | |

Stückliste :

| Zeile | Modul | Lagernummer | Bauteil-ID | Produkt-ID |
|---------|-------|-------------|------------|------------|
| 1 | M1 | 1 | 1 | P1 |
| 2 | M1 | 2 | 2 | P2 |
| 3 | M2 | 1 | 4 | P1 |
| (usw..) | | | | |

Nun als ER-Modell dargestellt :



Ein Produkt kommt in der Stückliste in mehreren Zeilen vor -> n

In einer Zeile der Stückliste kommt nur ein Produkt vor -> 1

In einer Zeile der Stückliste kommt nur ein Bauteil vor -> 1

Ein Bauteil kommt in der Stückliste in mehreren Zeilen vor -> n

Wenn man über den Sinn dieses Manövers nachdenkt, kann man erkennen, daß im Prinzip sichergestellt wird, daß in einer Zelle der Datenbank nur genau 1 Element eingetragen wird. „Atomar“ nennt man das hier. Rein datentechnisch ist das natürlich selbstverständlich. Nehmen sie an, der Datentyp einer Zelle sei Integer, wie wollen sie denn da zwei verschiedene Werte reinschreiben ?

Weiterführende Aufgabe :

Schreiben sie die vollständigen Tabellen für die Produkte und die Bauteile sowie einen ausführlichen Ausschnitt (mindestens 10..15 Zeilen) aus der Stückliste für die oben beschriebene Speicherstruktur der Modellfabrik, also ohne Zuliefersystem mit nur zwei Lagern an Modul 2.