



UNIT 0x03

MODELLIERUNG UND

ENTITY-RELATIONSHIP-MODELL I

Motivation

Q&A

- Wozu Modellierung?
- Welche Rollen sind beteiligt? Wer sieht was?
- Wie läuft der Prozess der Modellierung ab?

Motivation

Wozu Modellierung?

- Übersetzt ein reales Problem (Szenario/Mini-Welt, z.B. die lesenswerten Artikel) in eine 'bearbeitbare' Version, das Modell, um final ein Ergebnis (Datenbankentwurf) systematisch zu entwickeln.



Ein Zentrales Ziel der Datenbankvorlesung.

Was wird benötigt?

- Grundverständnis der Vorgehensweise und Definitionen.
- Abstraktionskonzepte und Tools, z.B. Entity-Relationship-Diagramme (ER-Diagramme) und Implementationskonzepte.

Nebenbedingungen an das Modell

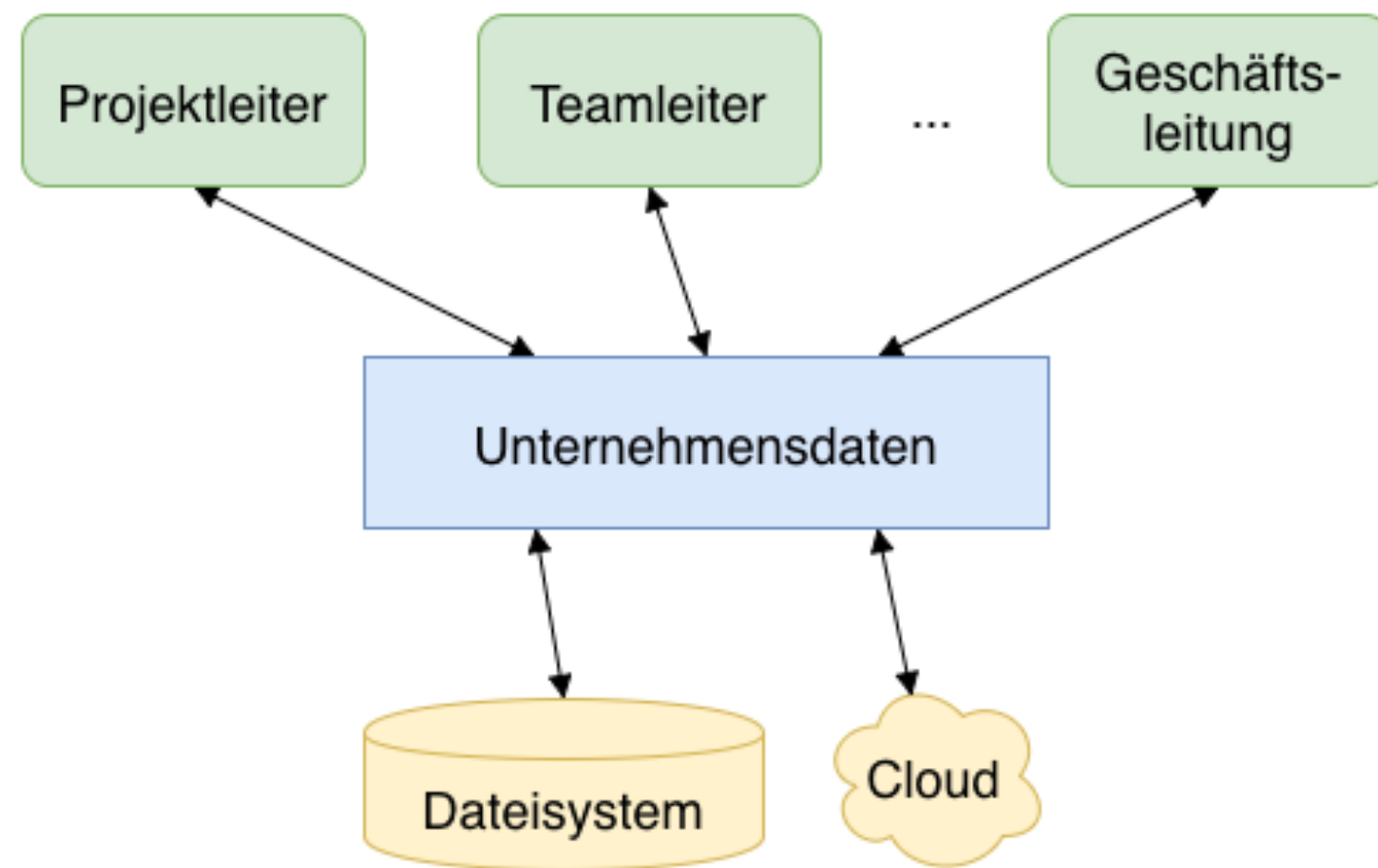
- Vollständig, korrekt, minimal, verständlich, erweiterbar.

Q&A

Woran erinnert Sie das?

Abstraktionsebenen

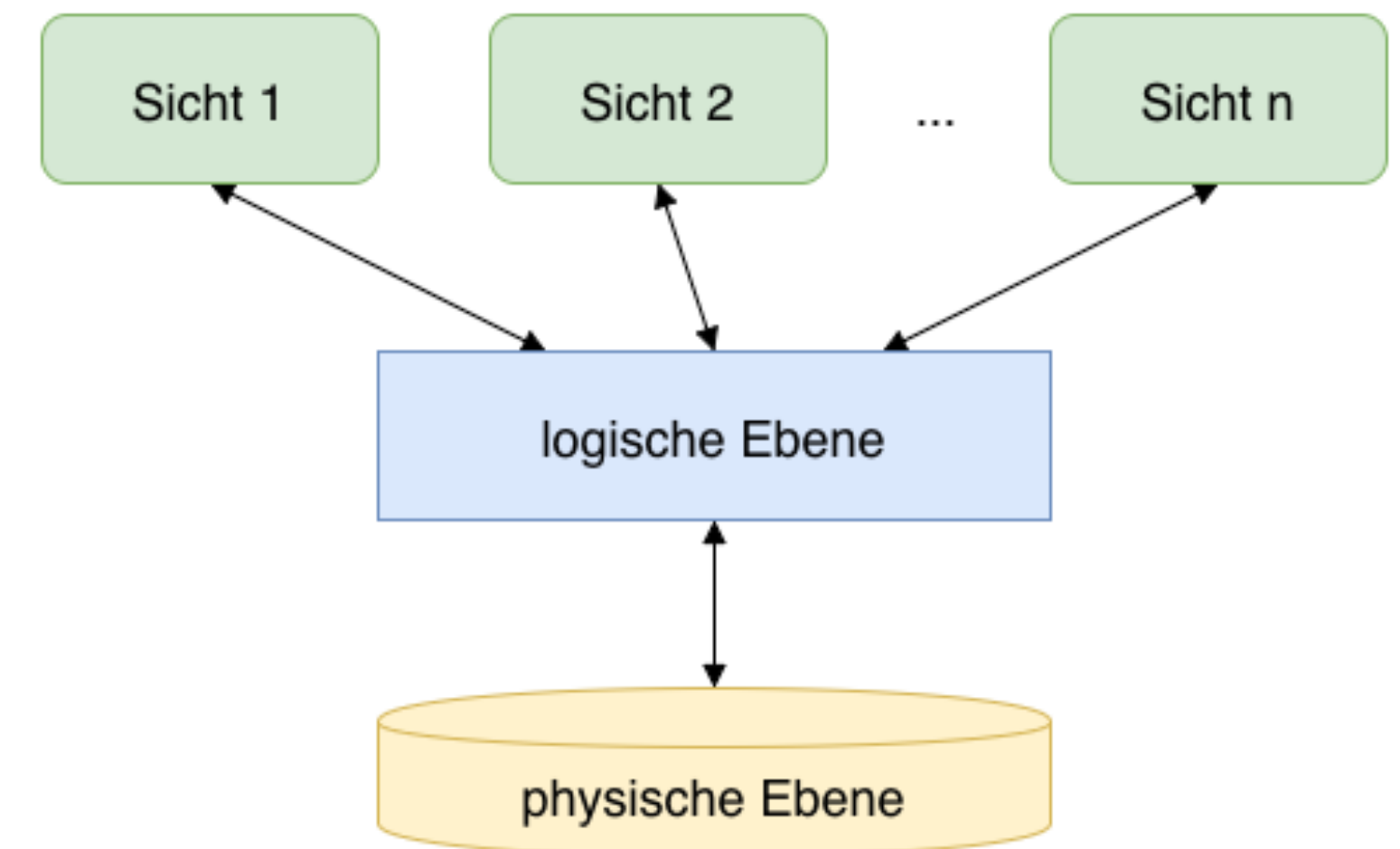
Ein erster Ansatz...



Je Rolle unterschiedliche Sicht auf Datenteilmenge.

Logische Ebene der Daten, d.h. Strukturierung, Schema.

Physische Ebene der Daten, d.h. Storage Engine, Cloud.



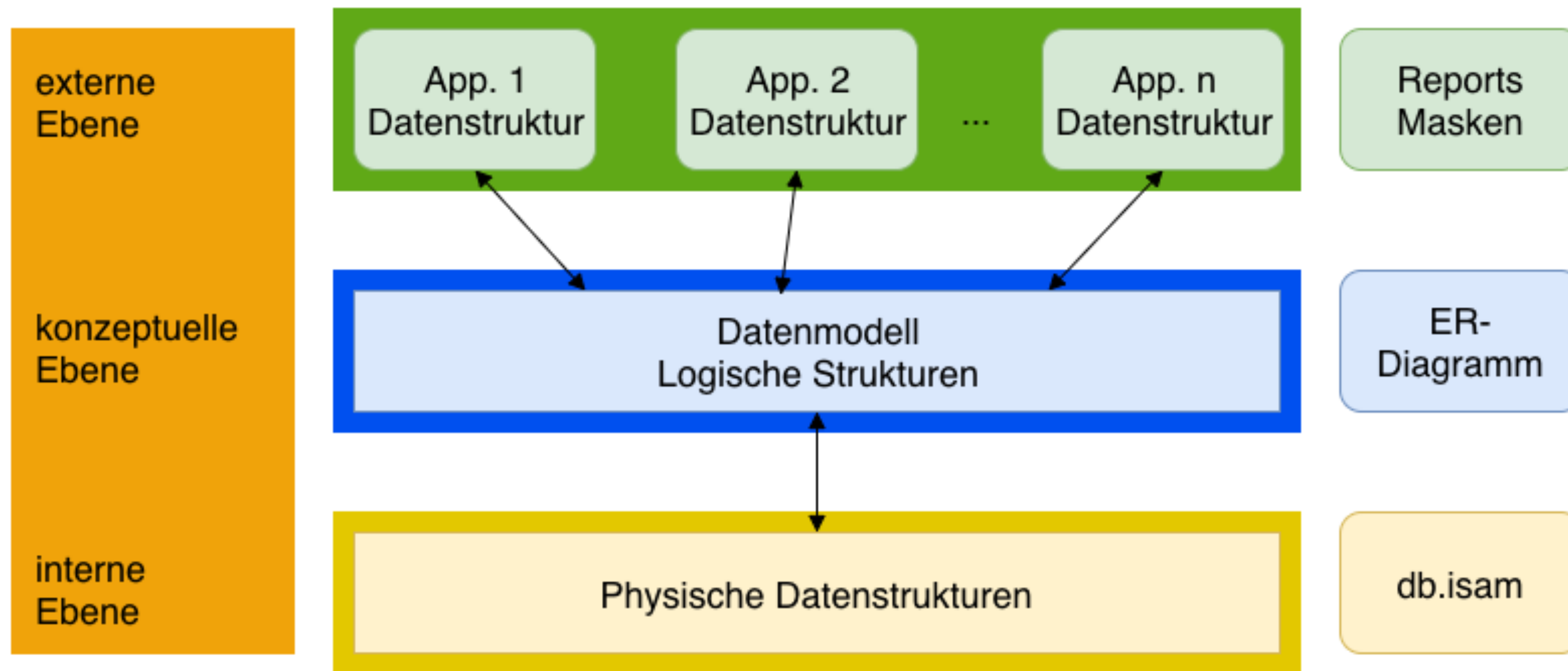
Idee: Aufbau in Schichten, so dass Austausch oder Modifikation darunter liegender Schichten problemlos möglich ist.



DBMS erfüllen zumeist nur physische Unabhängigkeit.

Abstraktionsebenen

ANSI/SPARC-Architektur

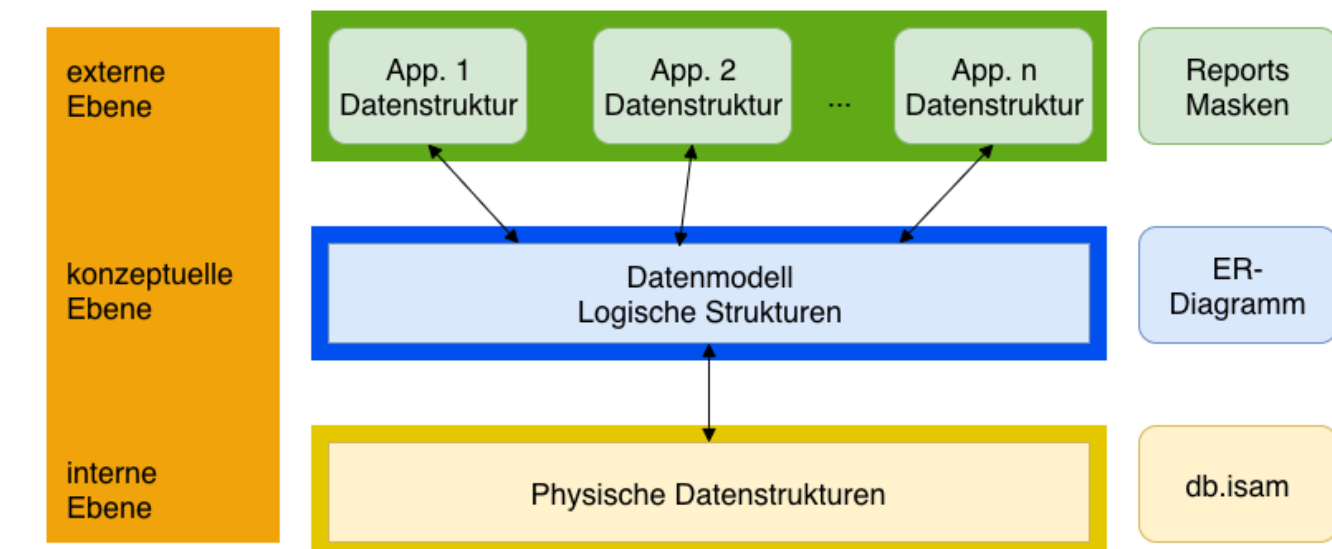


Abstraktionsebenen

ANSI/SPARC-Architektur

Drei getrennte Ebenen

- *Externe Ebene*: Individuelle Benutzersichten.
- *Konzeptuelle Ebene*: Vollständige und redundanzfreie Darstellung aller Informationen, Konzeptuelles (ER-Modell) und Datenbankschema mit Daten und Relationen.
- *Interne/physische Ebene*: Physische Sicht der Datenbank im Computer.



Vorteile

- *Logische Datenunabhängigkeit*: Änderungen auf der konzeptuellen Ebene haben keine Auswirkungen auf die externe Ebene.
- *Physische Datenunabhängigkeit*: Änderungen auf der internen Ebene, z.B. Wechsel des DBS, wirken sich nicht auf die konzeptuelle oder externe Ebene aus.

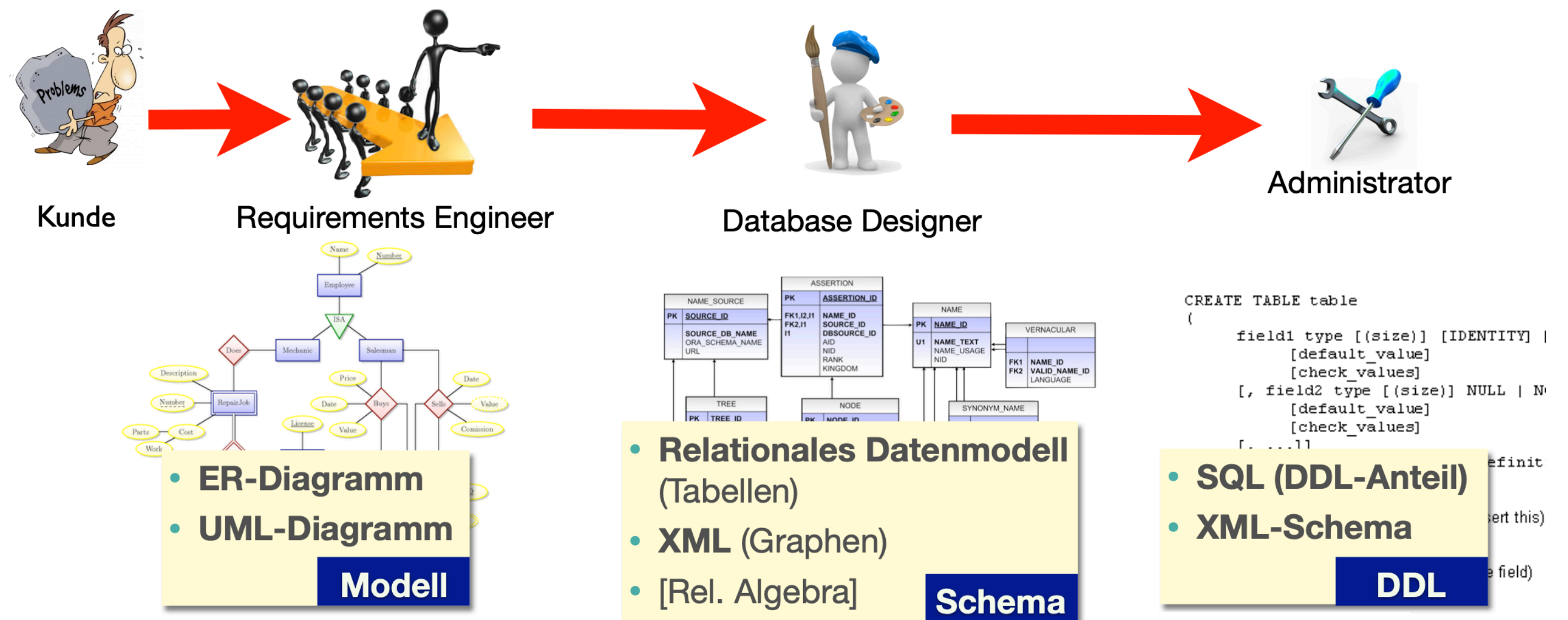


Soweit die Theorie...

Modellierungsprozess

Wie läuft die Entstehung einer Datenbank ab?

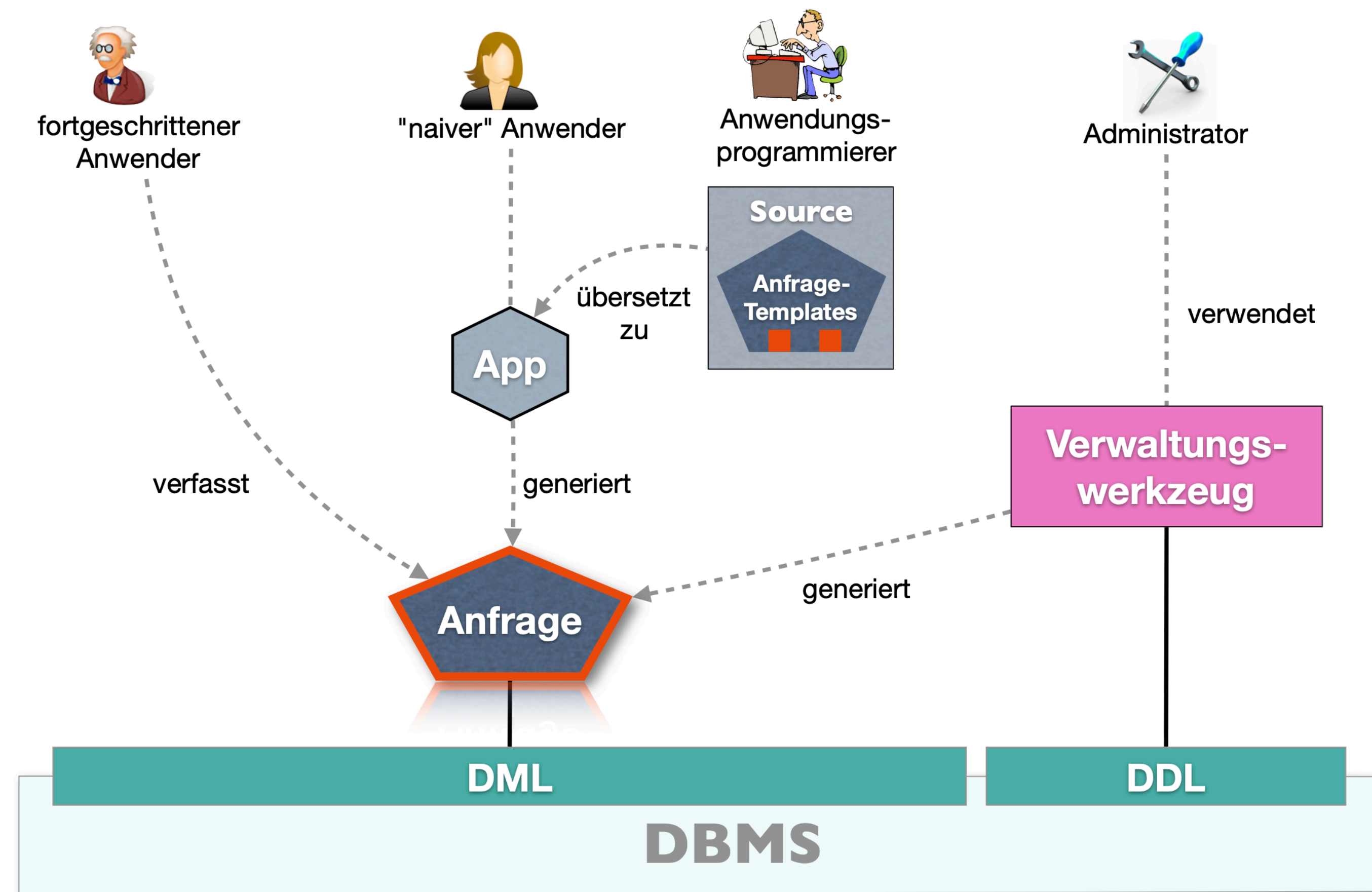
- **Rollen:** Wer übernimmt welche Aufgaben?
- **Anforderungsanalyse:** Was soll erreicht werden?
- **Entwürfe:** Konzeptuelle, Implementations-, Physische



Unterlagen Prof. Striegnitz

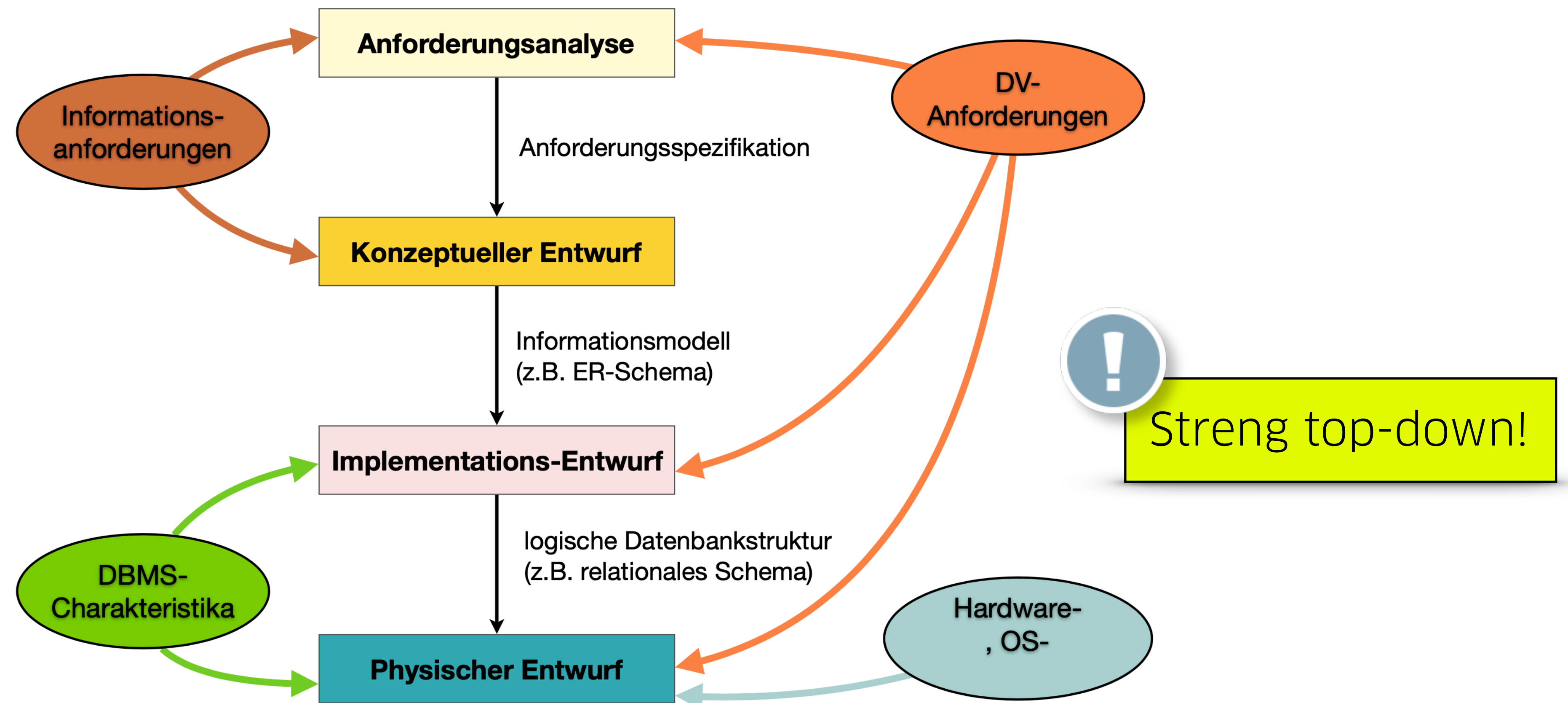
Modellierungsprozess

Architektur und Anwender



Modellierungsprozess

Phasen des Datenbankentwurfs

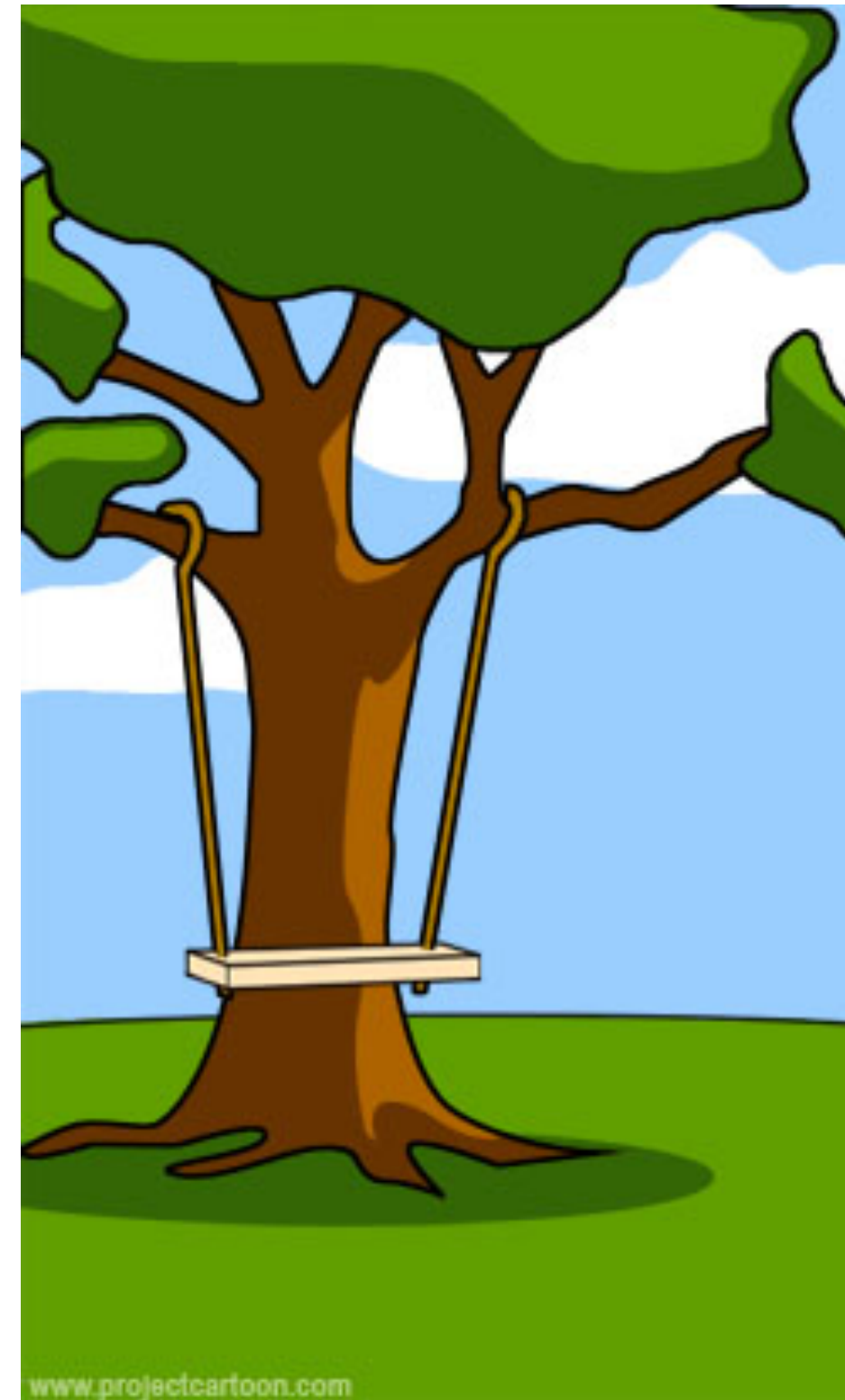


Unterlagen Prof. Striegnitz

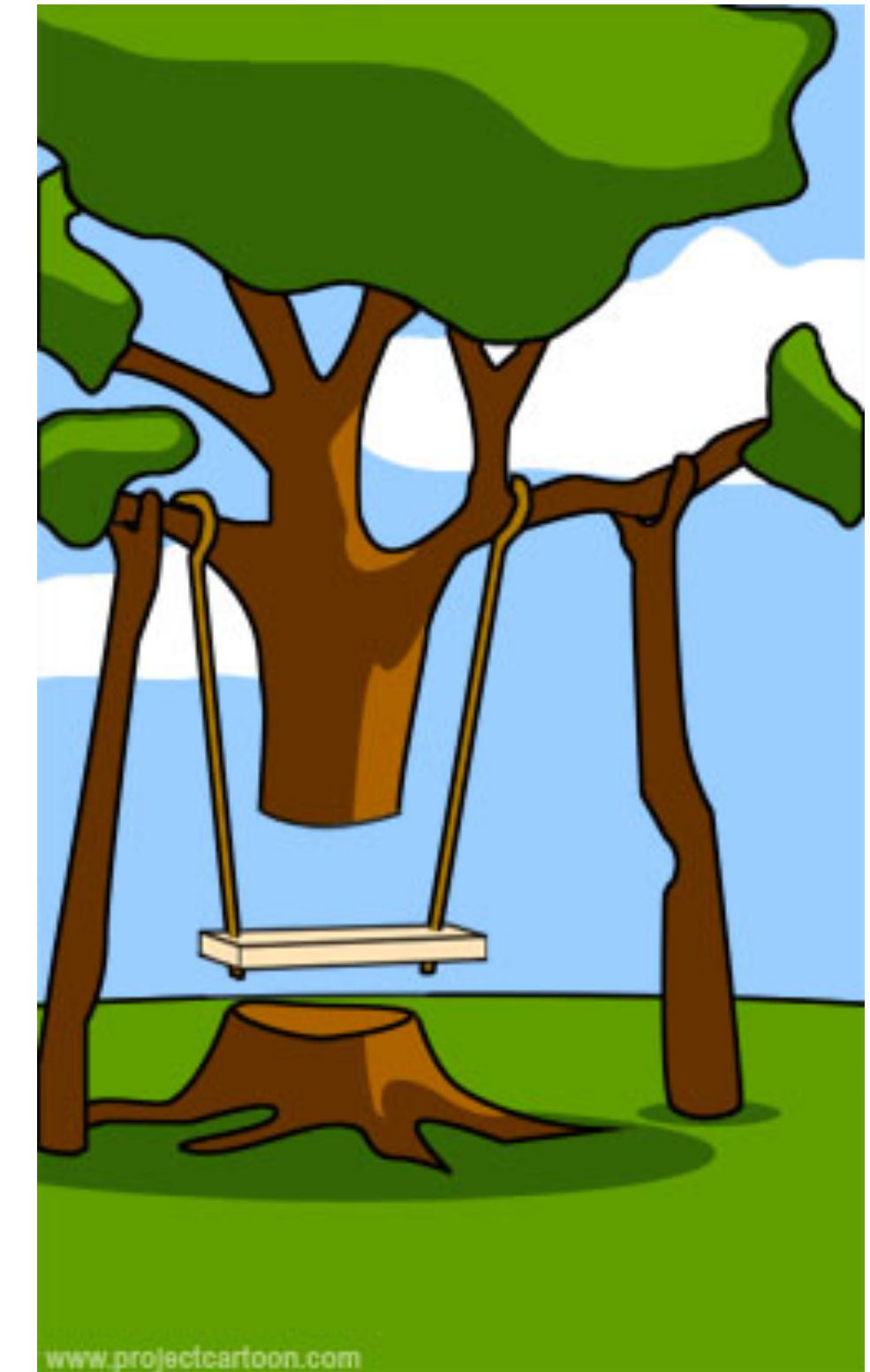
Anforderungsanalyse



Wie der Kunde
es erklärt hat



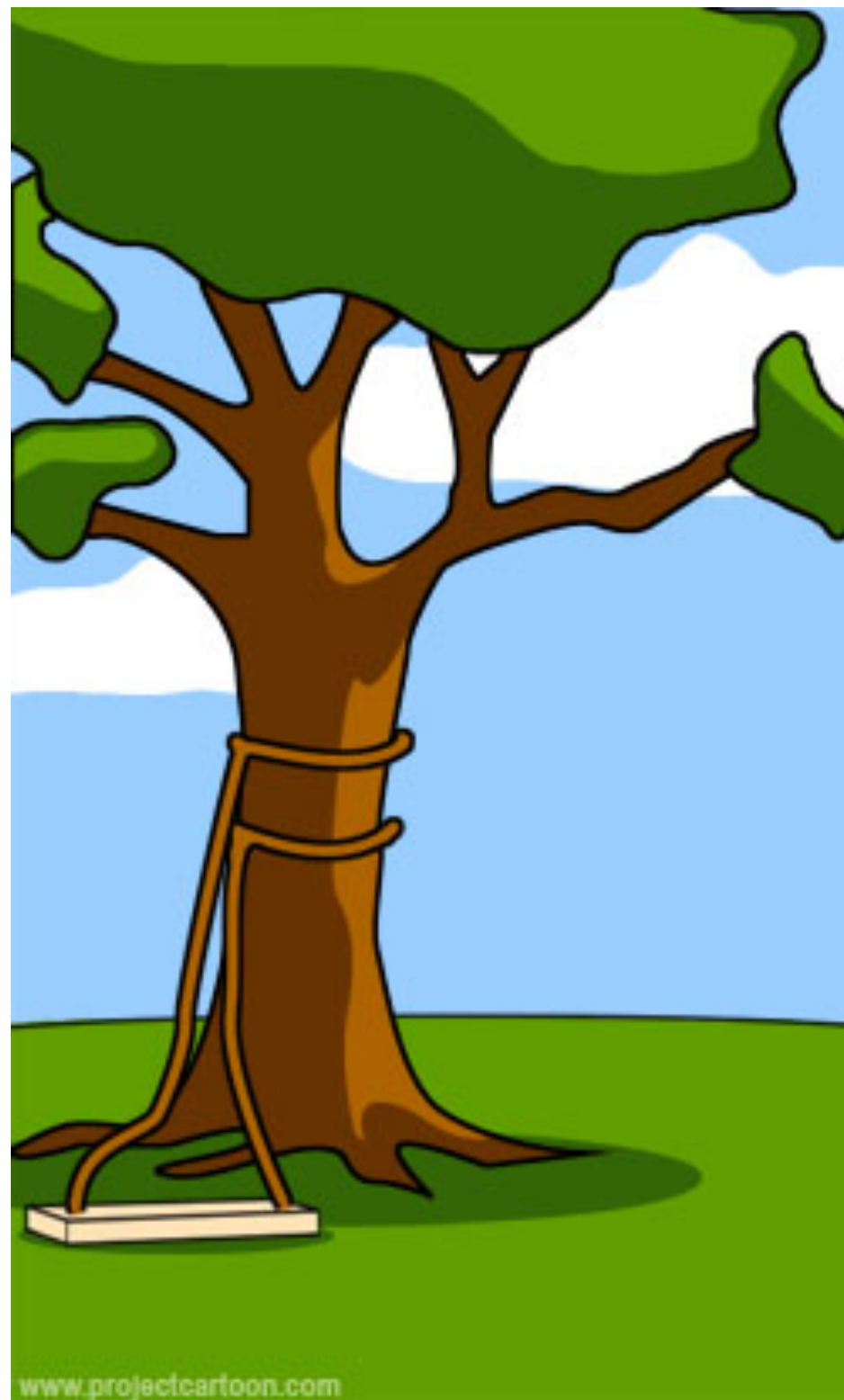
Was der Projekt-
leiter versteht



Wie der Analyst
es auffasst

Projectcartoon

Anforderungsanalyse



Was der Programmierer
geschrieben hat



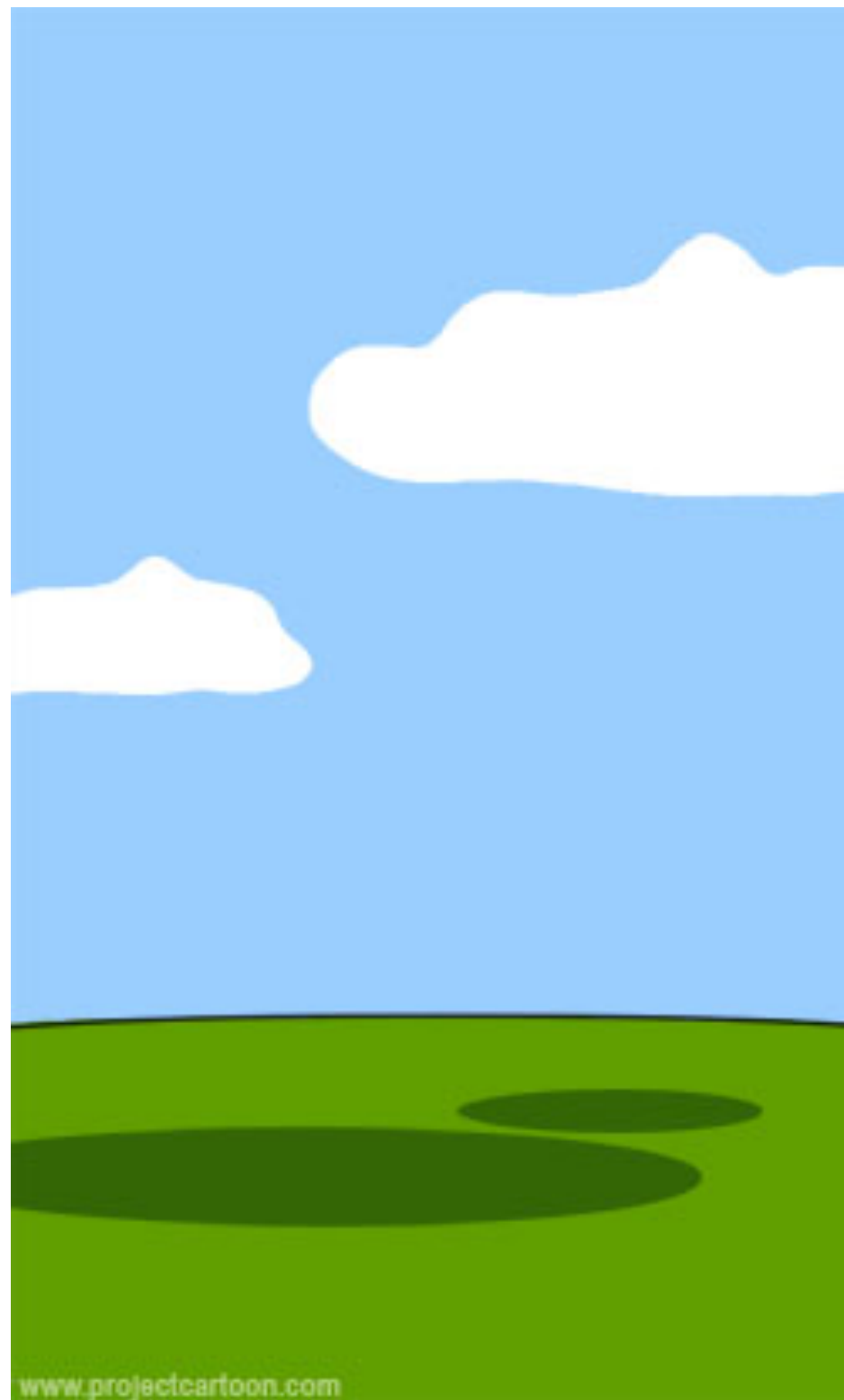
Was die Beta-
Tester erhalten



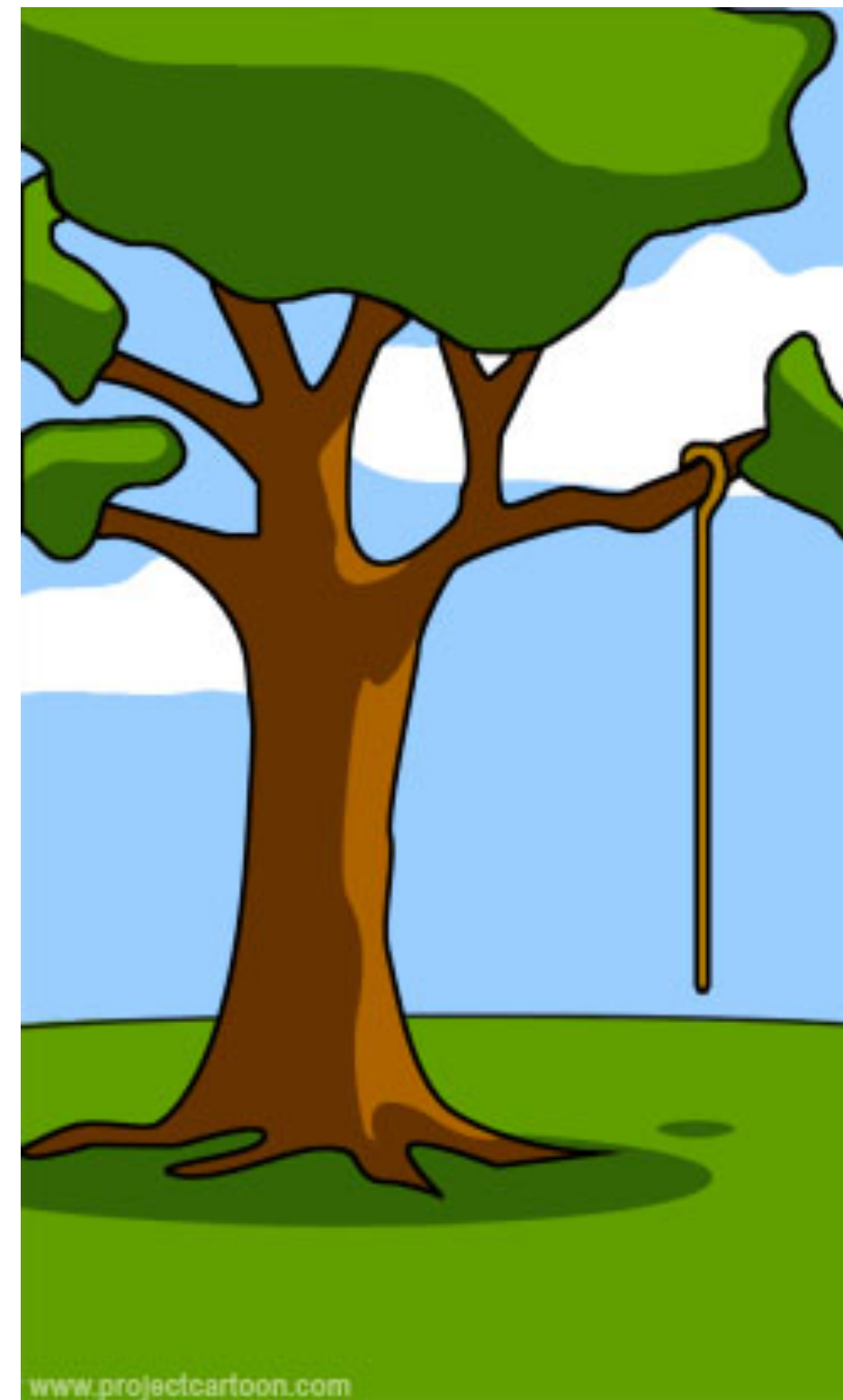
Wie der Vertrieb
es verkauft

Projectcartoon

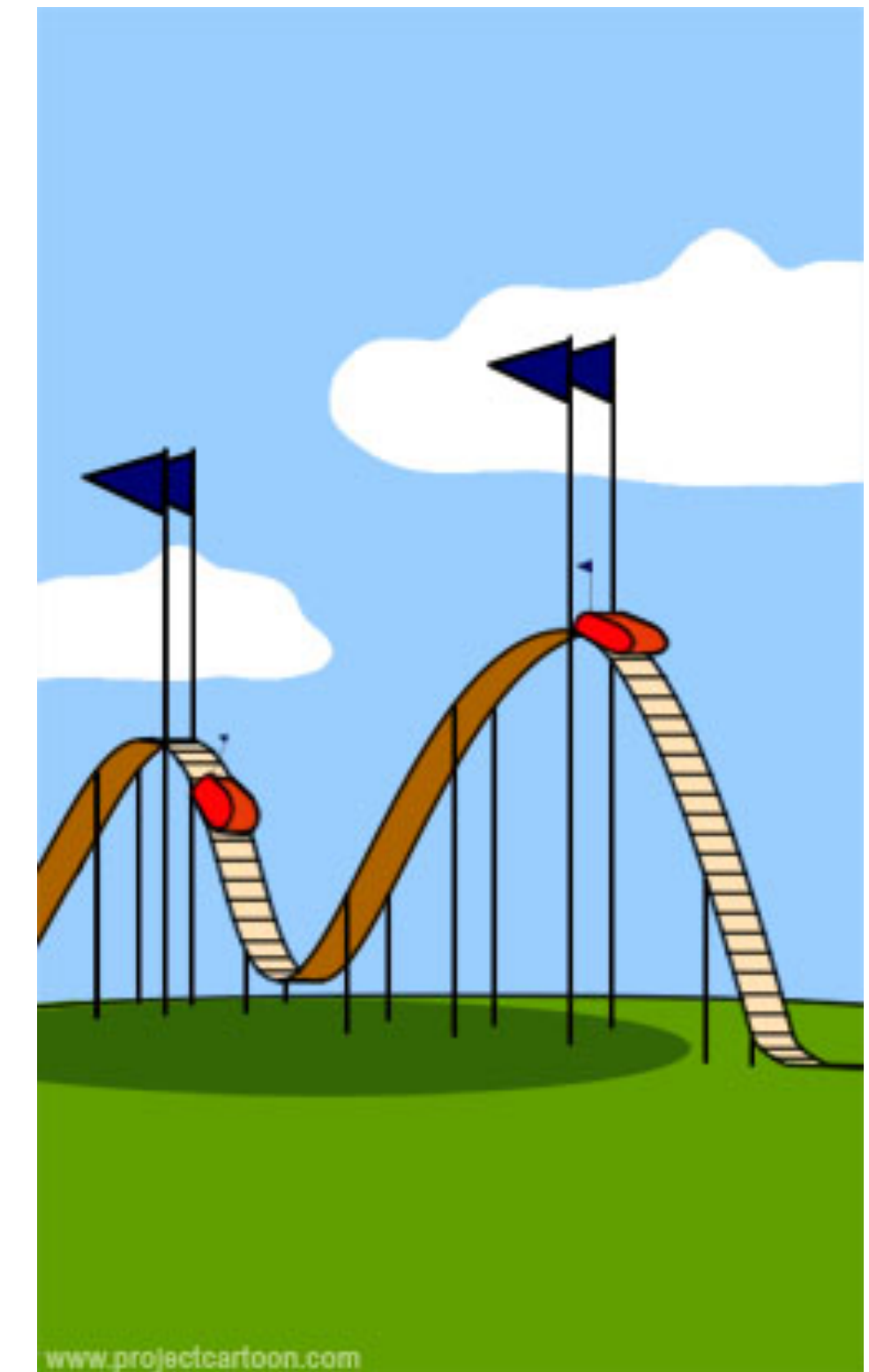
Anforderungsanalyse



Wie die Doku.
aussieht



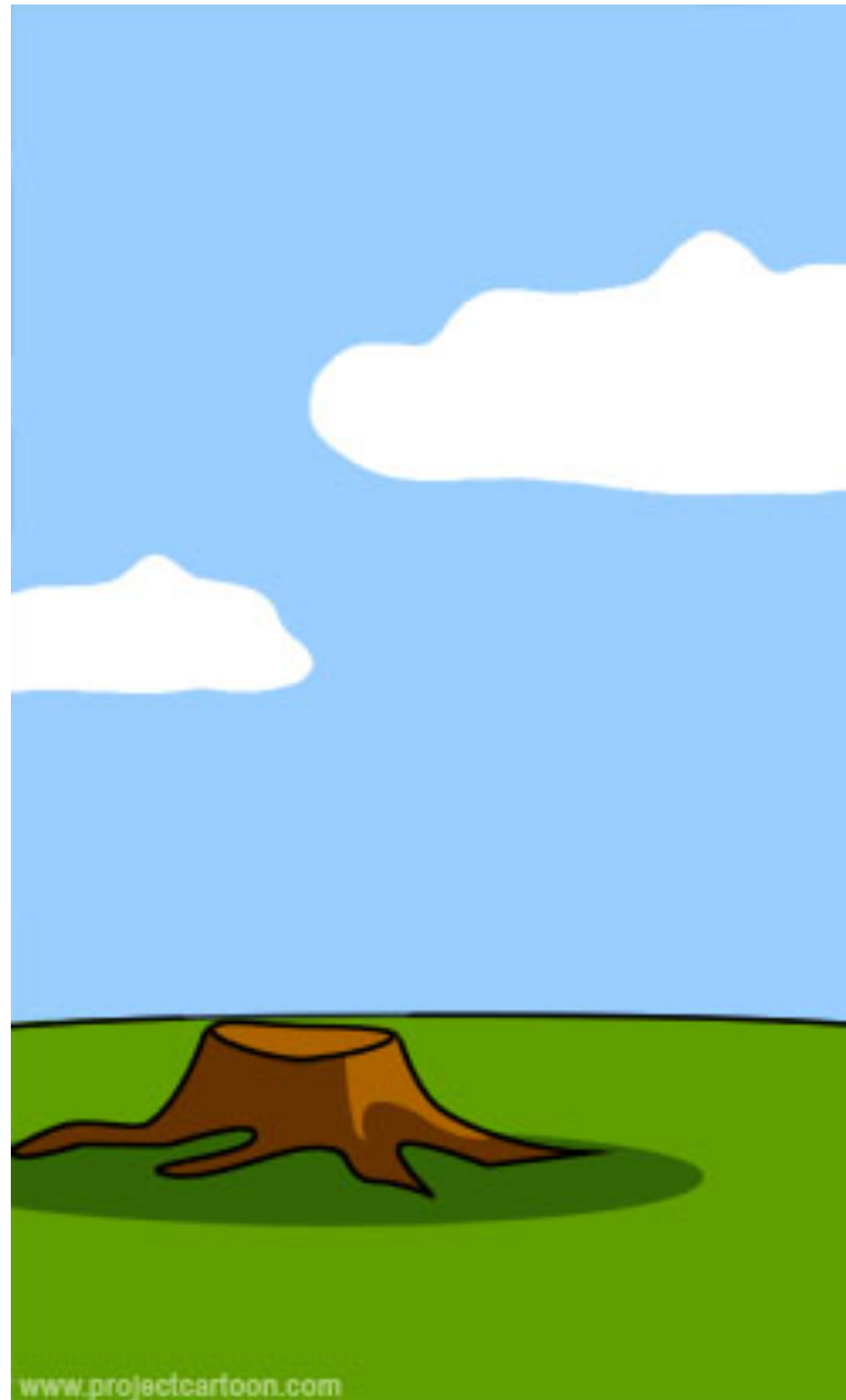
Was installiert
wurde



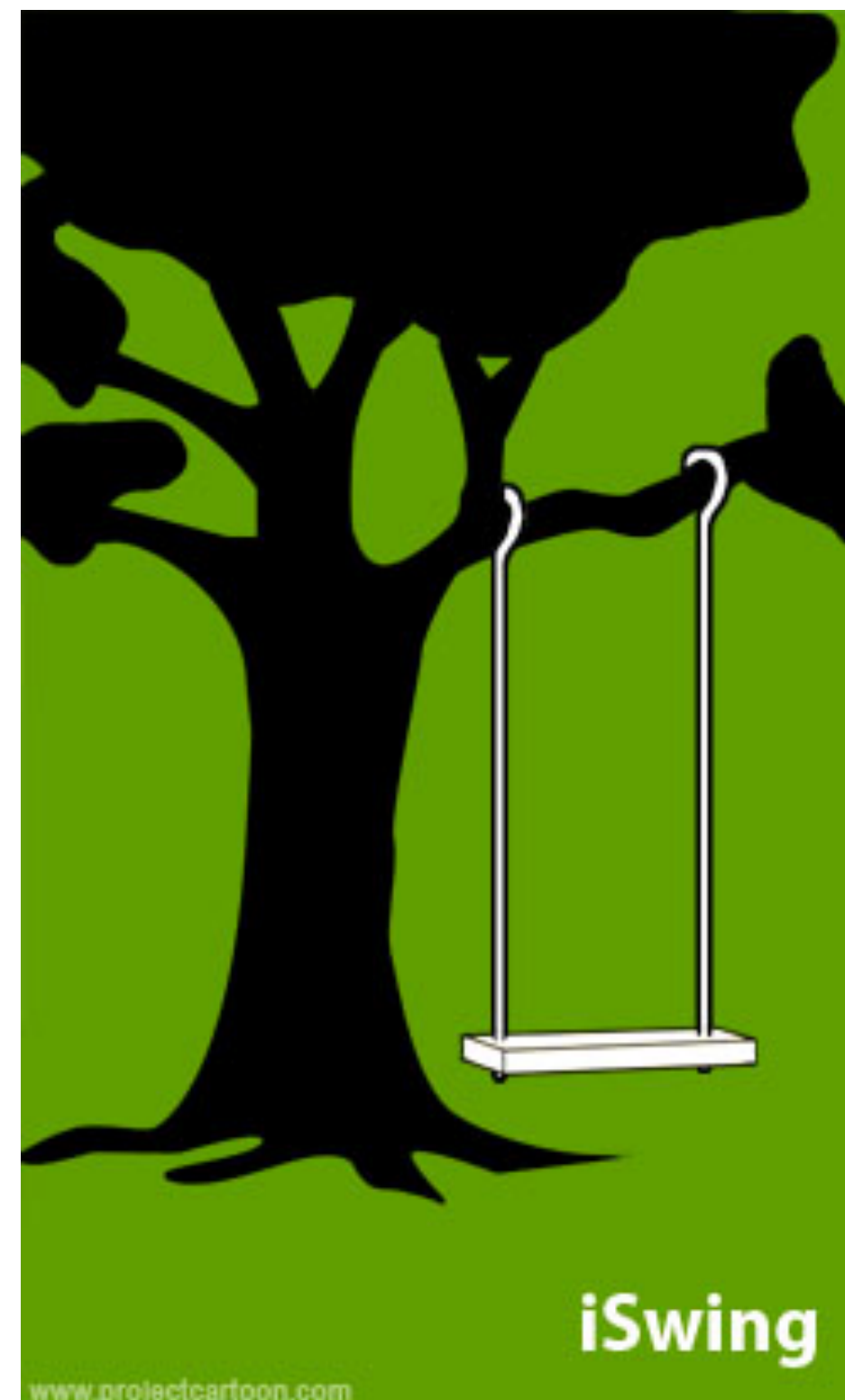
Wie es dem Kunden
berechnet wurde

Projectcartoon

Anforderungsanalyse



Wie der Support aussieht



Wie das Marketing damit wirbt



Was der Kunde gebraucht hätte

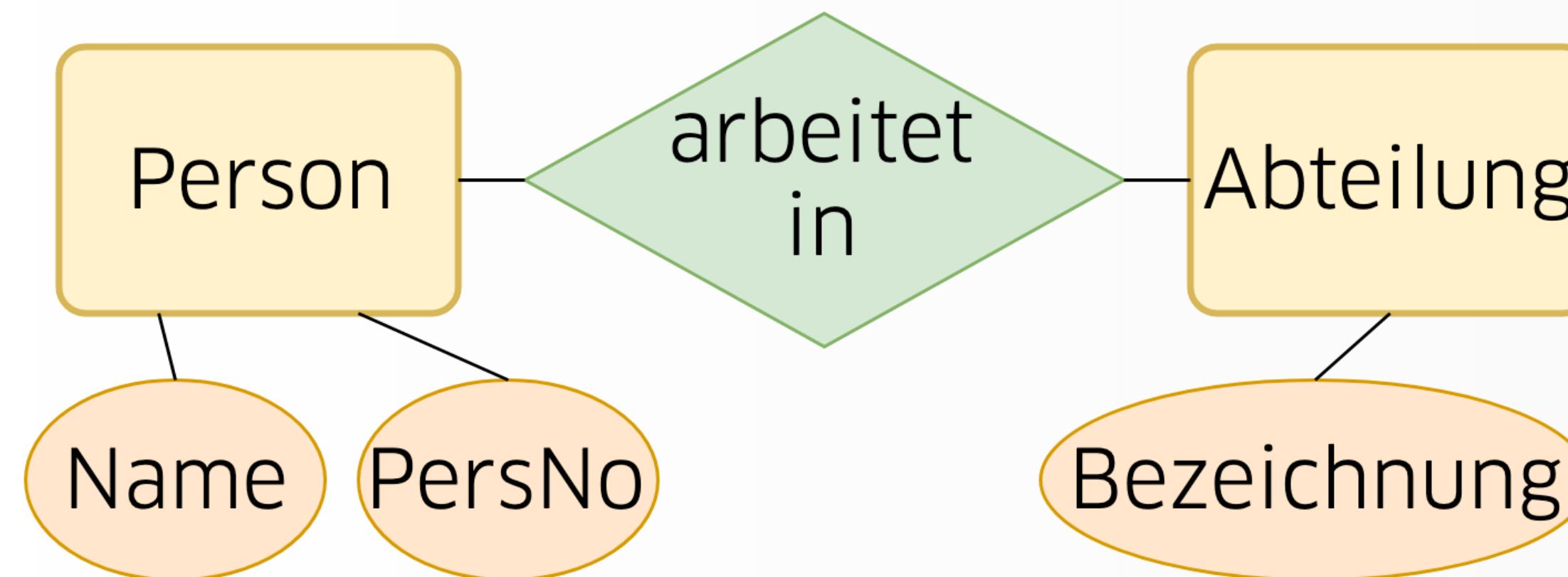
Projectcartoon

Konzeptueller Entwurf: Entity-Relationship-Modell / -Diagramm

Mini-Welt: Person mit Name und Personalnummer arbeitet in Abteilung.

Anforderungen: ...

Konzeptueller Entwurf:



ER-Diagramm



Q&A

- Wie sieht die Datenbank, bzw. das Schema, dazu aus?

Konzeptueller Entwurf: Datenbankentwurf

Welche Möglichkeiten des Datenbankentwurfs gibt es hier?

- Klar ist eine Tabelle zu Person und eine zu Abteilung mit den jeweiligen Attributen.
- Aber wie die Relation `arbeitet_in` genau abzubilden ist, hängt an den zuvor ermittelten Anforderungen!
 - Wenn dort festgelegt ist, dass jede Person *in genau einer* Abteilung arbeitet, dann könnte man die Information über die Abteilung bei der Person ablegen.
 - Wenn aber eine Person *in mehreren* Abteilungen arbeitet, dann kann man entweder mehrere Abteilungen bei der Person ablegen - führt zu anderen Schwierigkeiten - oder aber man muss diese Zuordnung, also die Relation, gesondert speichern.

Wie beim
Softwareentwurf!



Q&A

- Wie genau?
- Achtung:
Redundanzen!



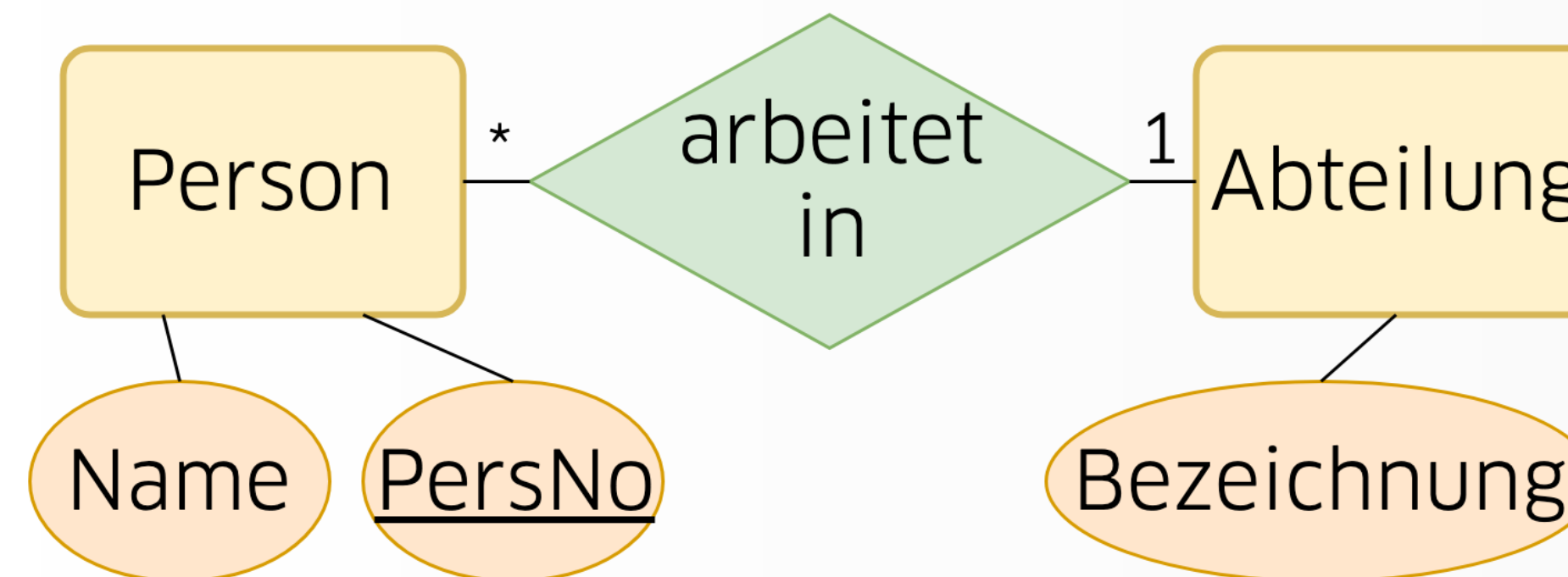
Der Datenbankentwurf ist nicht gleich dem ER-Diagramm!

Entity-Relationship-Modell / -Diagramm

Entwickelt 1976 von Peter Pin-Shan Chen



- Entity-Mengen, Attribute, Schlüssel, Relationen, Kardinalitäten, Wertebereiche;
- unterstützt Klassifikation und Aggregation;
- grafische Darstellung durch (einheitliche) Diagramme.



ER-Diagramm

- **Entity-Mengen**
- **Attribute**
Schlüssel
- ◆ **Relation**
- 1, * **Kardinalität**

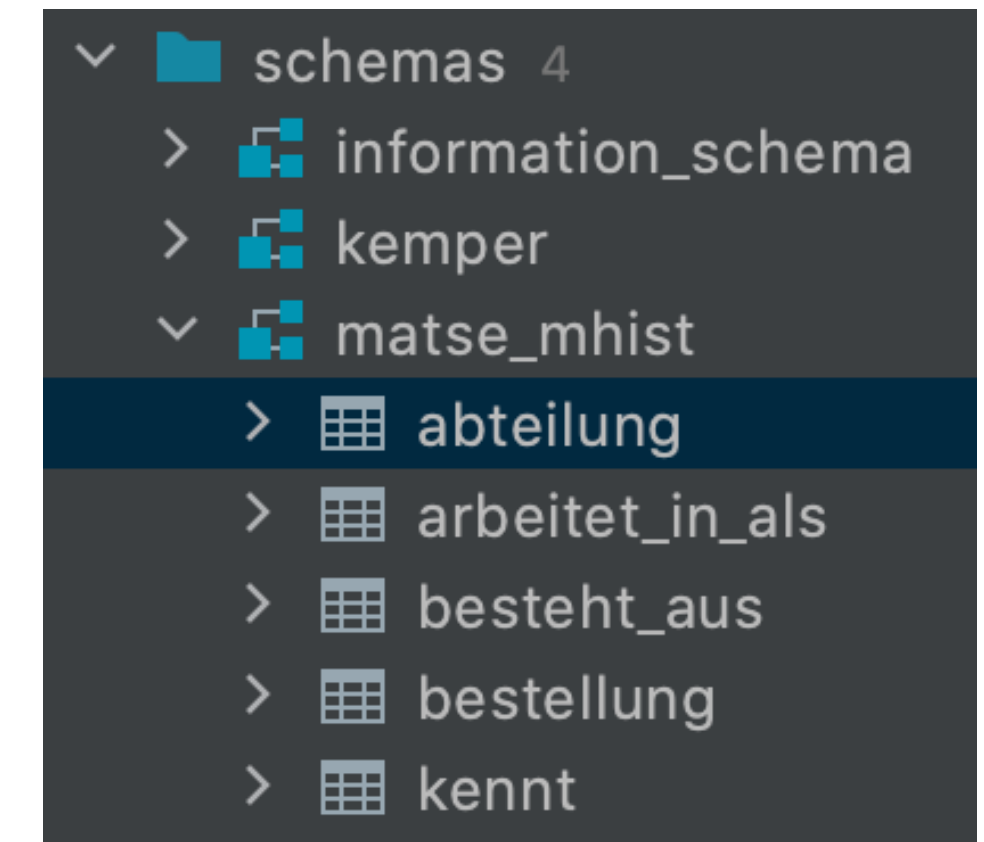


Gemeinsames Verständnis!

Definitionen

Schemata (Schemas)

- Allgemein formale Beschreibung der Struktur von Daten, z.B. XML-Schemata oder Datenbankschemata.
- Vergleichbar einem Namensraum aus der Programmierung.
- In einer relationalen Datenbank sind die Tabellen oft einem Schemata zugeordnet. Dem gegenüber werden manchmal z.B. sog. NoSQL-DBS als 'schema-less' bezeichnet, aber es gibt dort ebenfalls 'collections' oder 'namespaces' und Strukturen in den Daten – Stichwort 'Schema validation'.



Schema in MariaDB

Datenbankschemata

- Konkrete Festlegung, welche Daten in welcher Form gespeichert werden und welche Beziehungen dazwischen bestehen.

Definitionen

Entity/Entität

- Repräsentiert abstraktes oder physisches, aber konkretes Objekt der realen Welt, z.B. Person 'Max', Stadt 'Aachen', Lied 'Hello' etc..
- Besitzt Werte zu Attributen bzw. Eigenschaften jeweils mit Typ, z.B. 'Name' 'Max' vom Typ 'Text'.
- Vergleichbar einer Objektinstanz.
- Entspricht einem einzelnen Datensatz in einer Datenbank, z.B. der Zeile in einer Tabelle oder einem 'Dokument'.



ein 'reales' Objekt...

id	bezeichnung	stueckpreis	einheit
1	Spinat	1.99	PK
2	Vier Käse Pizza	2.39	ST
3	Spinatpizza	2.29	ST
4	Fischstäbchen	1.99	PK
5	Nudelpfanne	3.29	PK

Entitäten der Tabelle 'produkt'

Definitionen

Entity-Set/Entity-Menge oder Entitätstyp

- Menge E von Entitäten $e \in E$ mit gleichen Eigenschaften (Klassifikation), z.B. 'Person'. Hier liegt eine Betrachtung als *Menge von Objekten* zugrunde.
- Oder, mit Schwerpunkt auf der Beschreibung der Eigenschaften, ist E ein Entitätstyp mit in der Regel typisierten Attributen, z.B. 'Name' vom Typ 'Text' (varchar).
- Entitätstyp ist vergleichbar einer Klassenbeschreibung.
- Entspricht häufig einer Tabelle in einer relationalen Datenbank, hierbei sind die Spalten die Attribute.

Definition etwas unscharf – ob Menge oder Typ gemeint ist, hängt am Kontext.



Symbol eines Entitätstypen
im ER-Diagramm

Table:

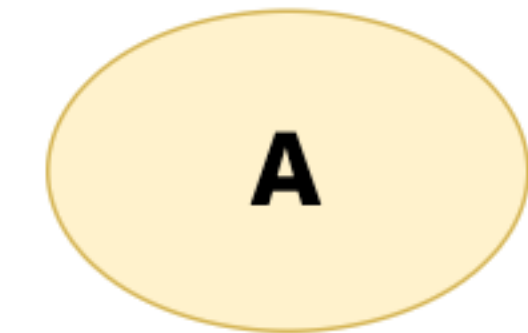
produkt	
Columns (6)	Keys (1) Indices (1)
id	int(11) -- part of primary key
bezeichnung	varchar(100)
warengruppe_id	int(11)
einheit	varchar(20)
stueckpreis	decimal(12,2)
umsatzsteuer	decimal(6,2)

Tabelle 'produkt' mit
Attributen und Datentypen

Definitionen

Attribute und Wertebereiche

- Eine Entität e ist durch die Werte ihrer Attribute charakterisiert und ein Entitätstyp E durch die Attribute selber. Jedem Attribut A ist ein Datentyp T und damit implizit oder auch explizit ein Wertebereich D (Domain) zugeordnet, der festlegt, welche Attributwerte zulässig sind: $A \in D$ oder $A:D$ bzw. $A:T$.
- *Achtung Besonderheit:* Nullwert (`NULL`). Spezieller Attributwert, dessen Bedeutung variiert, d.h. z.B. ist der Wert unbekannt oder noch nicht festgelegt oder nicht möglich. `NULL` kann, muss aber nicht, im Wertebereich liegen.
- Wir notieren $E [A_1:D_1, A_2:D_2, \dots A_n:D_n]$ für einen Entitätstyp E mit den Eigenschaften A_1, \dots, A_n . Wertebereiche bzw. Typen werden, wenn nicht relevant, auch ausgelassen.
- $W(D)$ bezeichnet alle real existierenden Werte der Domain D in einer Datenbank.

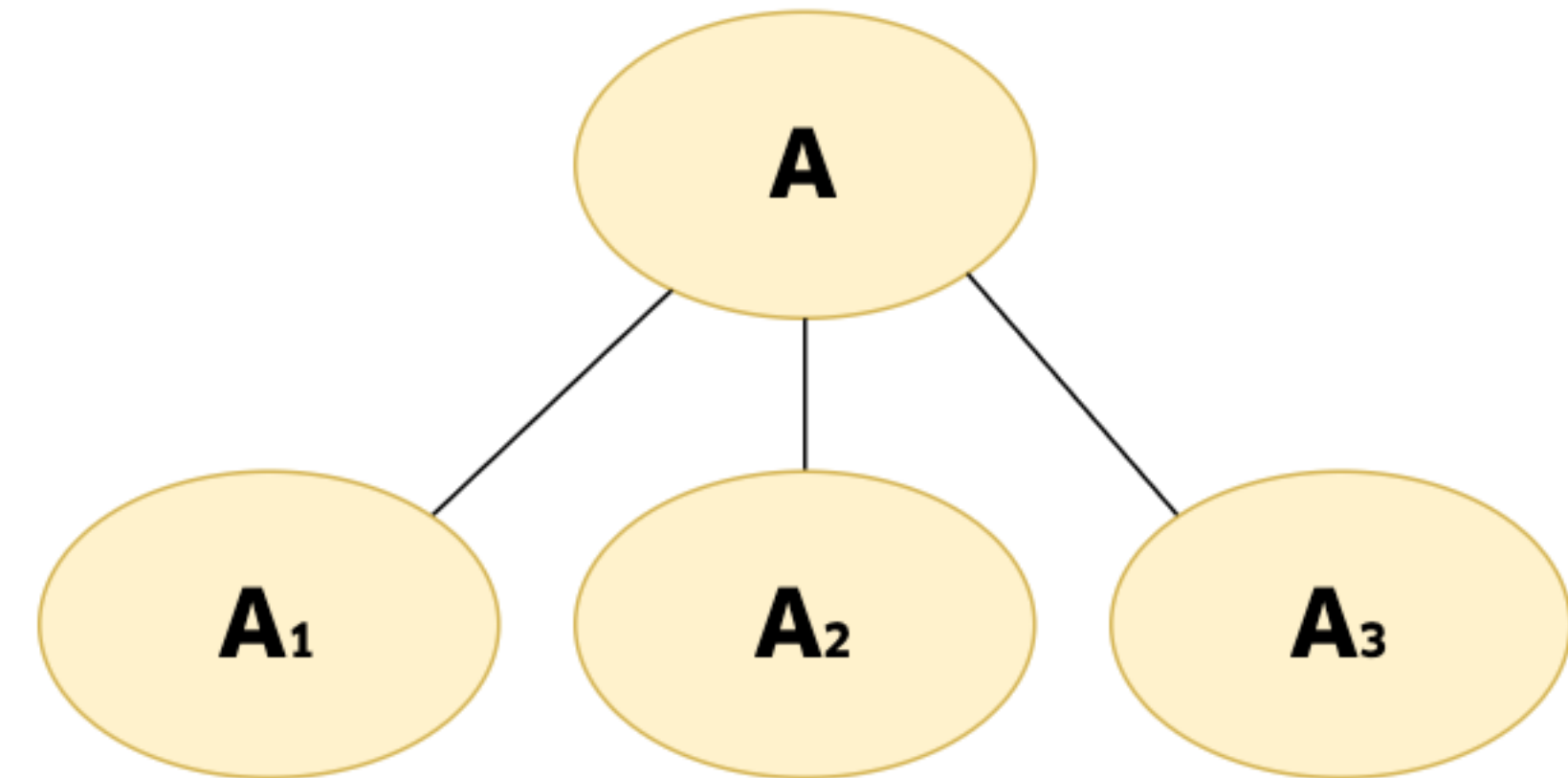


Symbol eines Attributs
im ER-Diagramm

Definitionen

Zusammengesetzte Attribute

- Zusammengesetzte Attribute haben selbst Attribute, z.B.
 - NAME: [Vorname: char(30),
Nachname: char(30)]
 - ANSCHRIFT: [Strasse: char(30),
Ort: char(30),
PLZ: char(5)]
- Domain für ein zusammengesetztes Attribut A mit Unterattributen A_1, \dots, A_n :
 $A: [A_1:D_1, \dots, A_n:D_n]: D_1 \times \dots \times D_n$.



Symbol eines zusammengesetzten
Attributs im ER-Diagramm

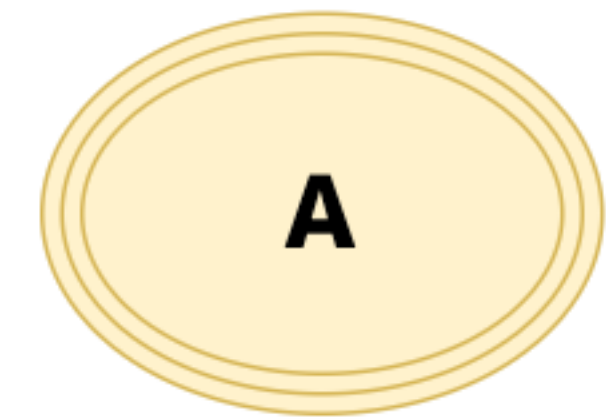
Q&A

- Was für ein Problem ist mit Blick auf eine Datenbank bzw. Tabelle zu erwarten?

Definitionen

Mehrwertige Attribute

- Ein mehrwertiges Attribut kann mehrere Ausprägungen (Werte) haben, z.B.
 - AUTOFARBE: {char (20)} (ein Auto hat mehrere Farben)
 - TELEFONNR: {char (30)} (eine Person hat mehrere Tel.nr.)
- Domain für ein mehrwertiges Attribut E: { A }



Symbol eines mehrwertigen Attributs im ER-Diagramm

Q&A

- Was für ein Problem ist mit Blick auf eine Datenbank bzw. Tabelle zu erwarten?

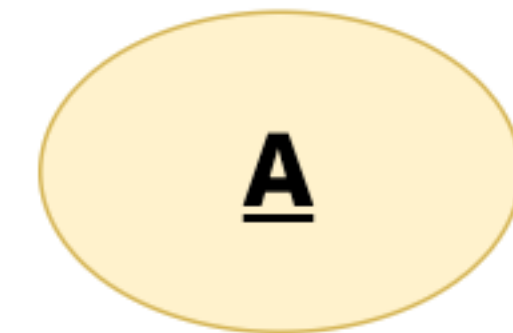
Definitionen

Schlüsselkandidaten

- Ein Schlüsselkandidat, oder kurz Schlüssel (key), ist ein einwertiges Attribut oder eine Attributkombination, die jede Entität eindeutig identifiziert.
- Es ist möglich, dass mehrere Schlüsselkandidaten existieren. Der sog. Primärschlüssel ist ein ausgewählter Schlüsselkandidat. Seine Primärschlüsselattribute werden im ER-Diagramm durch Unterstreichungen gekennzeichnet.
- Häufig wird eine künstliche Identifikationsnummer (ID) als Primärschlüssel gewählt, aber eine Postleitzahl in einem Städteverzeichnis wäre auch möglich.

Q&A

- Vor- oder Nachteile künstlicher Primärschlüssel?



Symbol eines Primärschlüsselattributs im ER-Diagramm

Table:

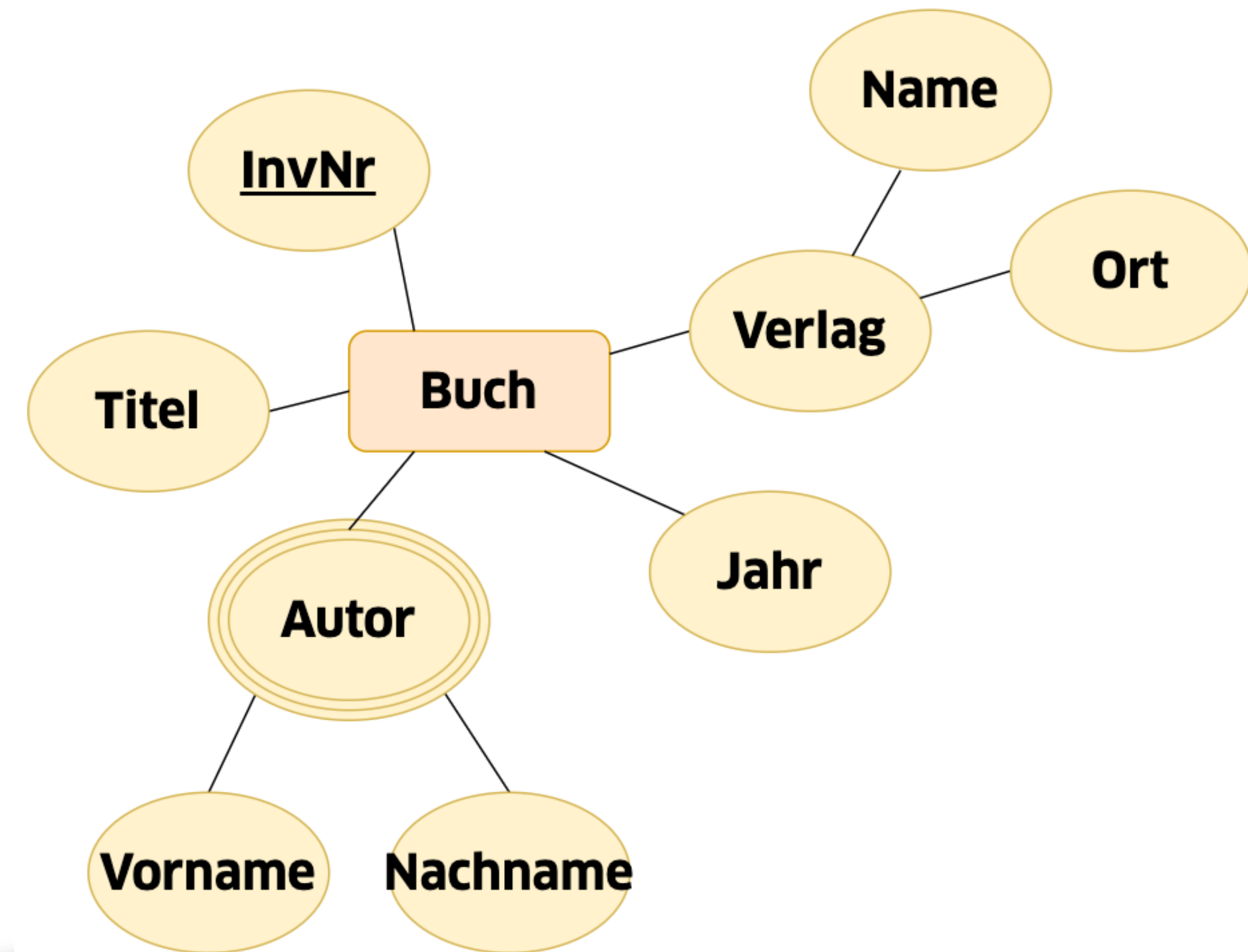
Columns (6)		Keys (1)	Indices (1)
id	int(11)	-- part of primary key	
bezeichnung	varchar(100)		
warengruppe_id	int(11)		

Primärschlüssel 'id' in der Tabelle 'produkt'

Beispiel ER-Diagramm

Beispiel Buch


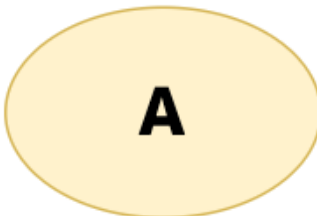
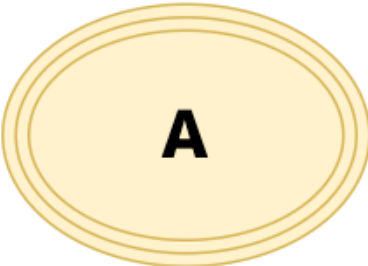
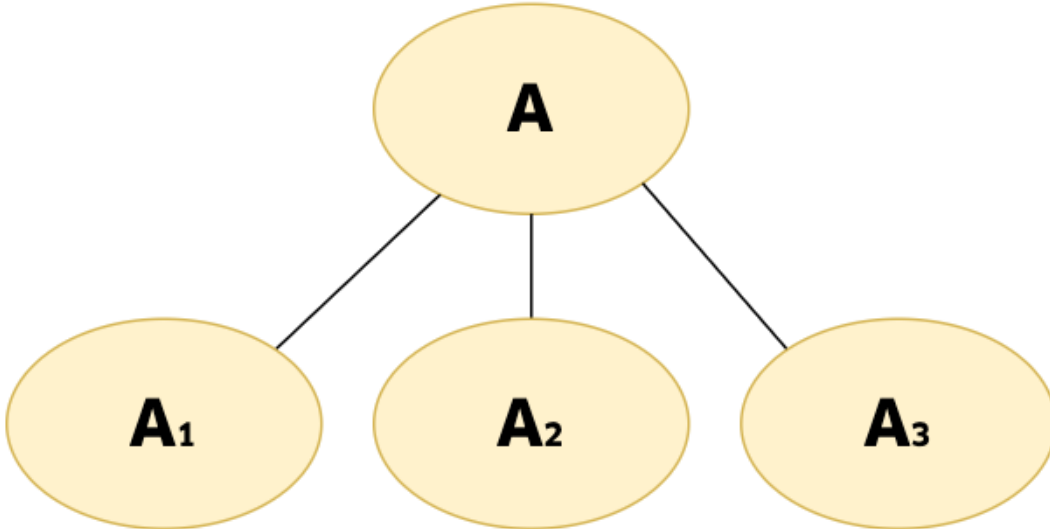
- Angabe des Entitätstypes in der Form
E: <[Attribute],{Primärschlüssel}>, d.h.



Buch: < [InvNr, Titel, Jahr, Verlag:[Name,Ort], {Autor:[Vorname,Nachname]}], {InvNr} >

ER-Diagramm Übersicht

Grafische vs. formale Darstellung

	ER-Diagramm	Formal
Entity-Set		$E: \langle [S], \{K\} \rangle$ $S = [A_1, \dots, A_n]$ $K = \{A_{i1}, \dots, A_{im}\}$
Attribut		$A:D$
Mehrwertiges Attribut		$A:\{D\}$
Zusammengesetztes Attribut		$A:[A_1:D_1, \dots, A_n:D_n]$

Definitionen

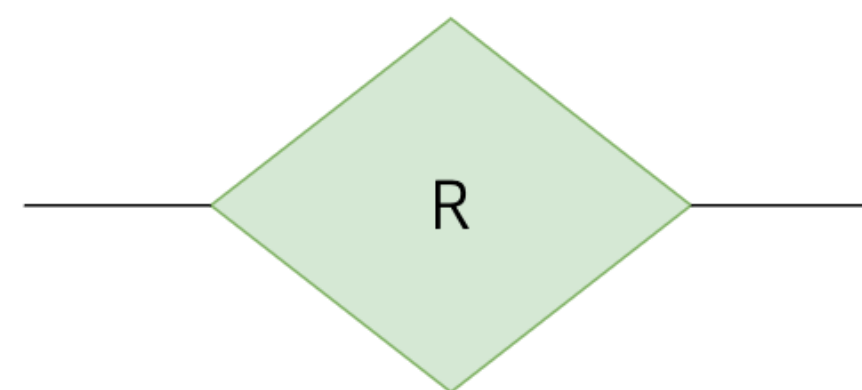
Relations/Relationships/Relationen/Beziehungen

- Eine Relationship-Menge R entspricht einer mathematischen Relation zwischen n Entity-Mengen E_i :

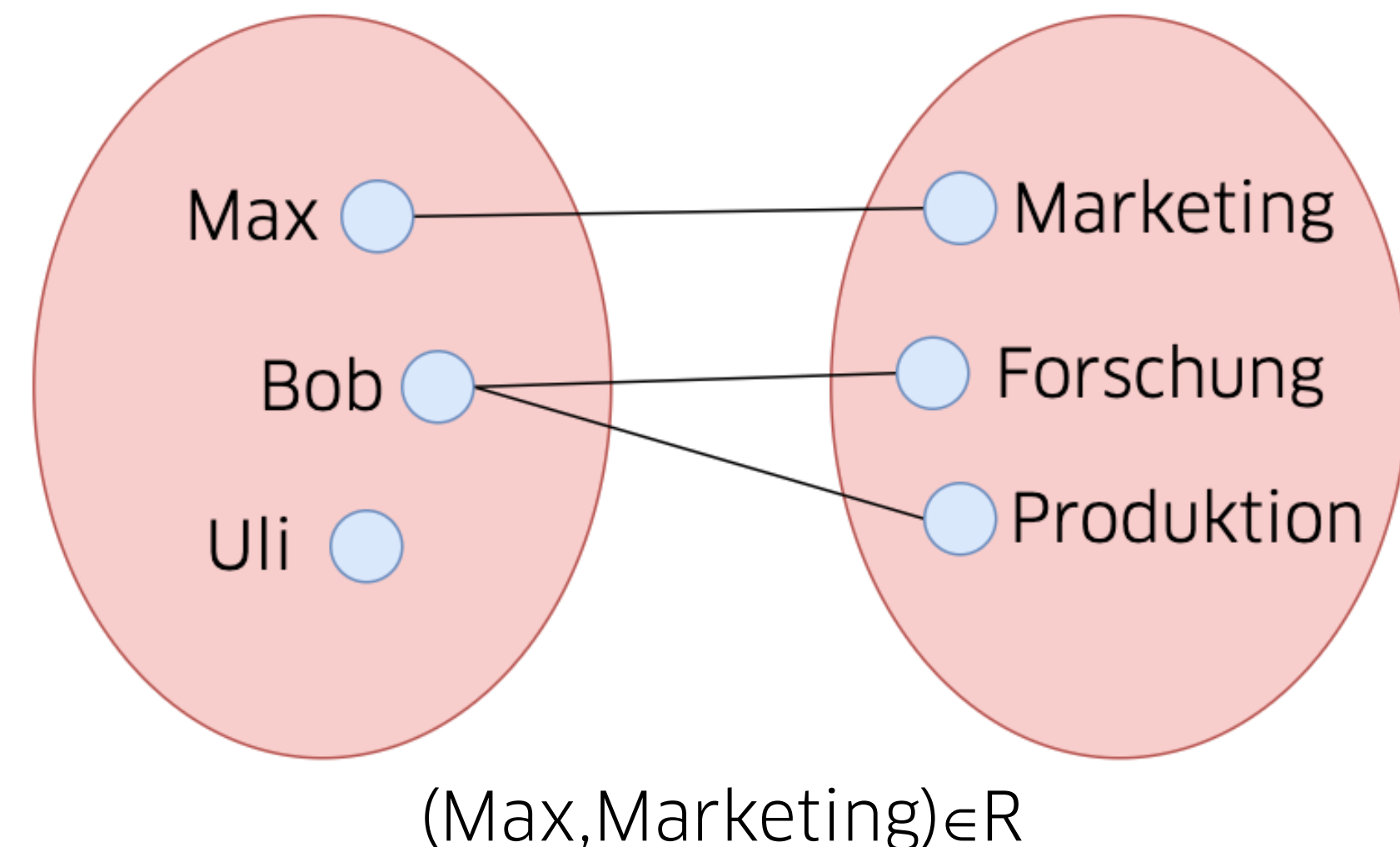
$R \subseteq E_1 \times \dots \times E_n$, häufig $n=2$ oder $n=3$.



- Anders ausgedrückt: Eine Relation gibt an, ob eine Entität e_1 einer Menge E_1 , also $e_1 \in E_1$, zu einer anderen Entität einer zweiten Menge, $e_2 \in E_2$, in Beziehung steht oder nicht, d.h. $(e_1, e_2) \in R$ oder $(e_1, e_2) \notin R$.
- Die Relation ist beschreibend: arbeitet in



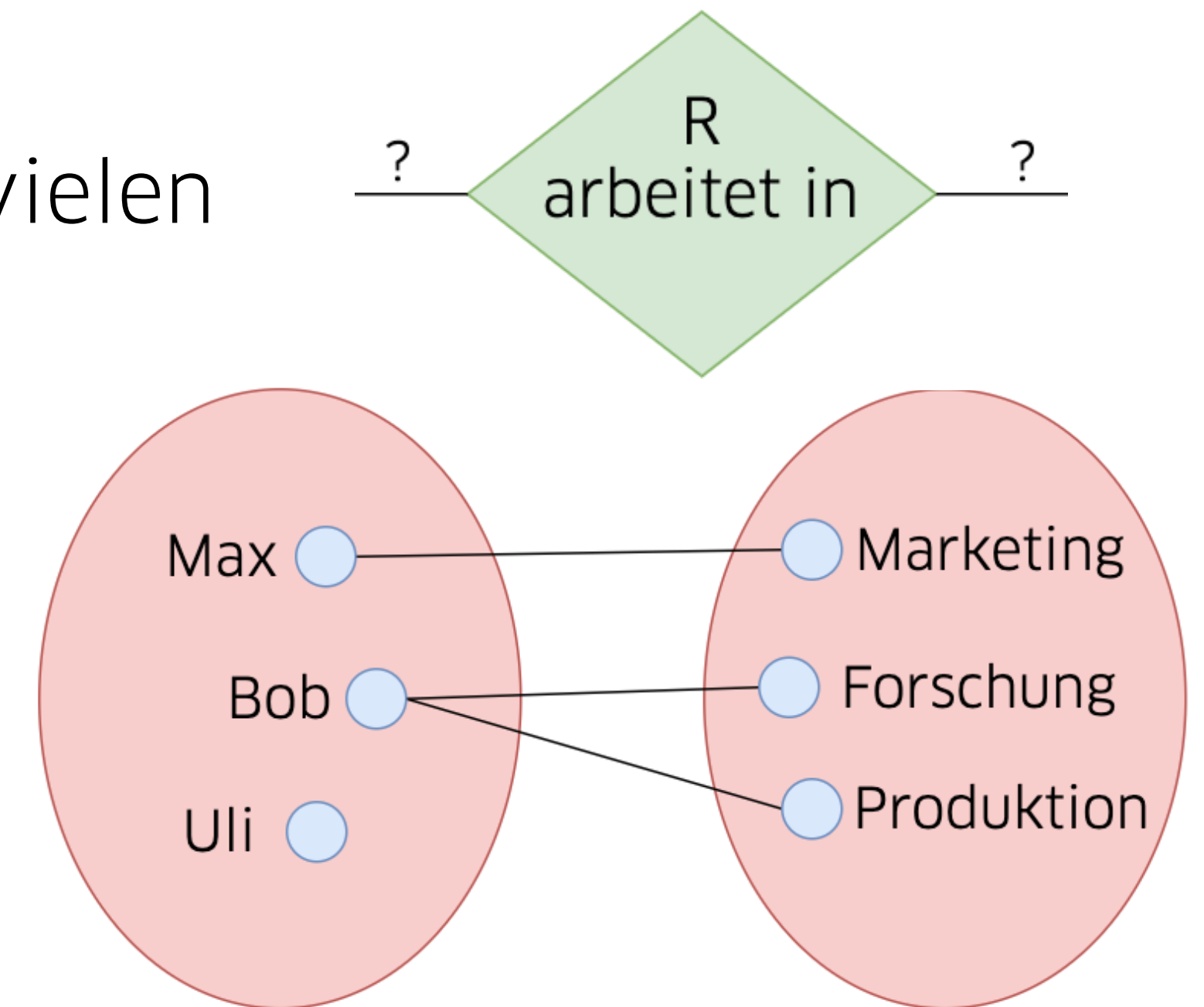
Symbol einer
Relation im
ER-Diagramm



Definitionen

Kardinalität

- Die Kardinalität (die '?' an der Relation) gibt an, zu wie vielen Elementen ein Element in Beziehung steht.
- Es gibt grundsätzlich folgende Abbildungsbeziehungen zwischen Entity-Mengen in der *Chen-Notation*: **1:1, n:1, 1:n, n:m**. (Details und Beispiele folgen).
- Die Kardinalität ergibt sich im Wesentlichen aus den Anforderungen und bestimmt ganz massgeblich den Datenbankentwurf.
- Die Abbildung des konzeptuellen Entwurfs, also des ER-Diagramms, auf den relationalen Implementationsentwurf, d.h. z.B. die Abbildung in die Datenbankstrukturen/Tabellen, folgt im Kapitel 'Relationales Modell'. Mögliche Abbildungen in Nicht-relationalen Datenbanken betrachten wir gegen Ende.



Relationen und Kardinalität

Mini-Welt Fotoshooting I

Hintergrund: Auf einer Familienfeier werden Fotos aufgenommen. Diese Fotos sollen in einer Datenbank abgelegt werden. Dabei besitzen Fotos u.a. Datum und Zeit und eine fortlaufende Nummer und Personen mind. einen Namen. Es gilt, dass auf einem Foto max. eine Person zu sehen sein soll und dass es max. ein Bild von einer Person geben darf.

- **Ziel Konzeptueller Entwurf:**
 - zunächst ER-Model
 - später: Implementationsentwurf (DB-Schema)

Q&A

Was können Sie identifizieren?

- Entitätstypen?
- Attribute?
- Relationen?
- Kardinalitäten?

Relationen und Kardinalität

Mini-Welt Fotoshooting I

Hintergrund: Auf einer Familienfeier werden **Fotos** aufgenommen. Diese Fotos sollen in einer Datenbank abgelegt werden. Dabei besitzen Fotos u.a. *Datum* und *Zeit* und eine fortlaufende *Nummer* und **Personen** mind. einen *Namen*. Es gilt, dass auf einem Foto max. eine Person zu sehen sein soll und dass es max. ein Bild von einer Person geben darf.

Mögliches Vorgehen

- Sie identifizieren zunächst die massgeblichen **Entitätstypen**,
- zusammen mit deren *Eigenschaft/Attributen*, und
- den Beziehungen und Kardinalitäten untereinander.

Kardinalitäten sind ggf. implizit gegeben. Die Aussage max. eine Person zu max. einem Bild beschreibt eine 1:1-Beziehung zwischen Foto und Person.

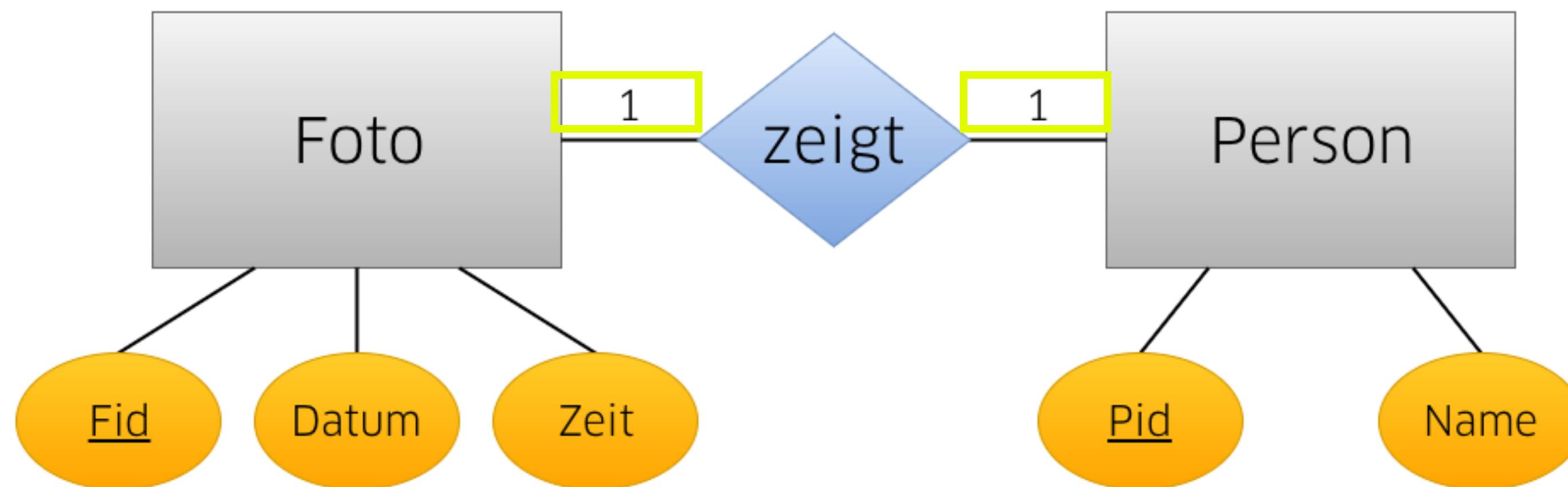


Markieren Sie
die Begriffe

Relationen und Kardinalität

ER-Diagramm mit Chen-Notation

- Die sog. **Chen-Notation** der Kardinalität läßt eine '1', ein 'n' oder alternativ '*' zu.
- Achtung: '1' meint allerdings 0 oder 1, d.h. hier dürfen Elemente auch keine Partner haben, d.h. es darf ein Foto auch keine Person zeigen oder keine Person auf einem Foto abgebildet sein.



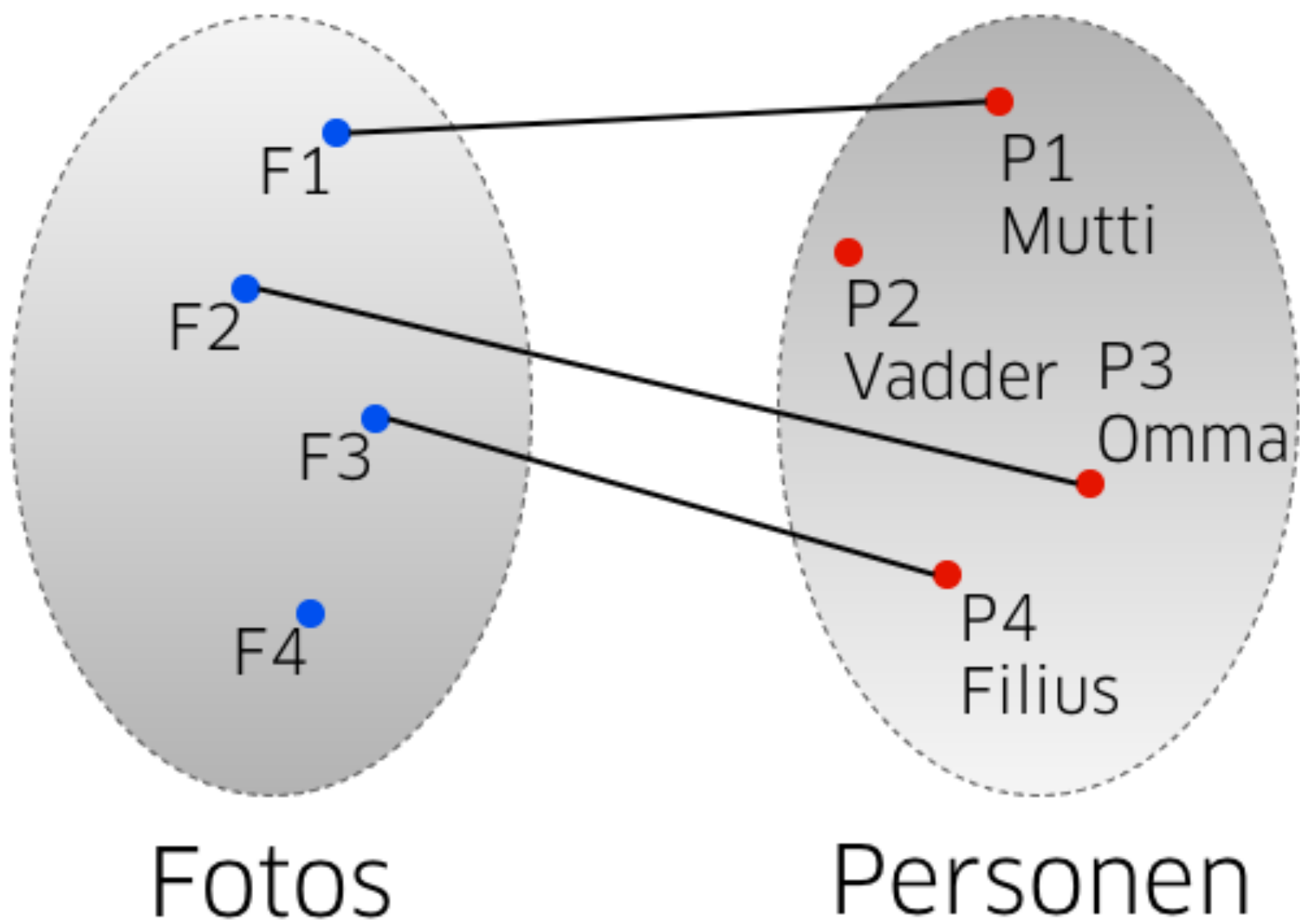
Relationen und Kardinalität

1:1 Relation

Die Mengendiagramme zeigen eine mögliche Zuordnung bei einer 1:1 Relation. Insbesondere gibt es Personen (P2) und Fotos (F4) ohne Partner.

Die Tabellen zeigen eine mögliche Variante (es gibt weitere), die Relation (letzte Tabelle Fid zu Pid) über sog. Fremdschlüssel abzubilden.

Man achte insbes. auf die Anzahl der Paare je Element. Bei einer 1:1 Relation kommt ein Element in max. einer Kombination vor.



Ein Foto zeigt max. eine Person und eine Person ist auf max. einem Foto.

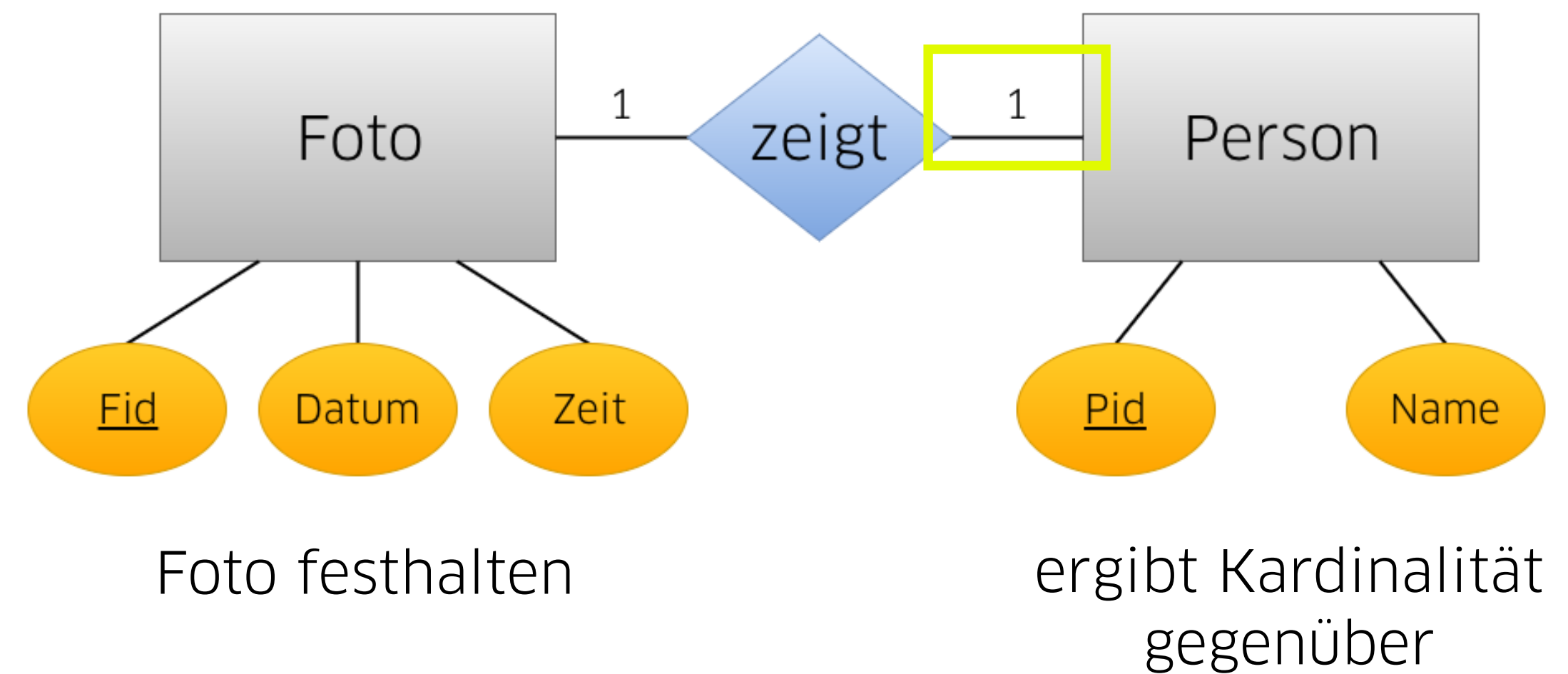
Fid	Datum	Zeit	Pid	Name	Fid	Pid
F1	24.12.18	09:00	P1	Mutti	F1	P1
F2	24.12.18	09:45	P2	Vadder	F2	P3
F3	25.12.18	00:05	P3	Omma	F3	P4
F4	25.12.18	04:15	P4	Filius		

Relationen und Kardinalität

Kardinalität

Die Bestimmung der Chen-Kardinalität kann man sich wie eine Abbildung der einen Seite (Foto) auf die andere Seite (Person) vorstellen:

- Wählen Sie ein Foto aus (wie in der Analysis, wählen Sie ein beliebiges aber festes x) und bestimmen Sie, wieviele Personen diesem Foto max. zugeordnet werden können (hier 1)
→ Kardinalität für Personen.
- Achtung: Foto festhalten links der Relation ergibt Kardinalität auf der rechten Seite.
- Die Kardinalität auf der linken Seite ergibt sich analog.



Relationen und Kardinalität

Mini-Welt Fotoshooting II

Hintergrund: Auf einer Familienfeier [...]

Am zweiten Weihnachtstag gilt jedoch: Ein Foto zeigt beliebig viele Personen, aber eine Person ist weiter nur auf max. einem Foto zu sehen.

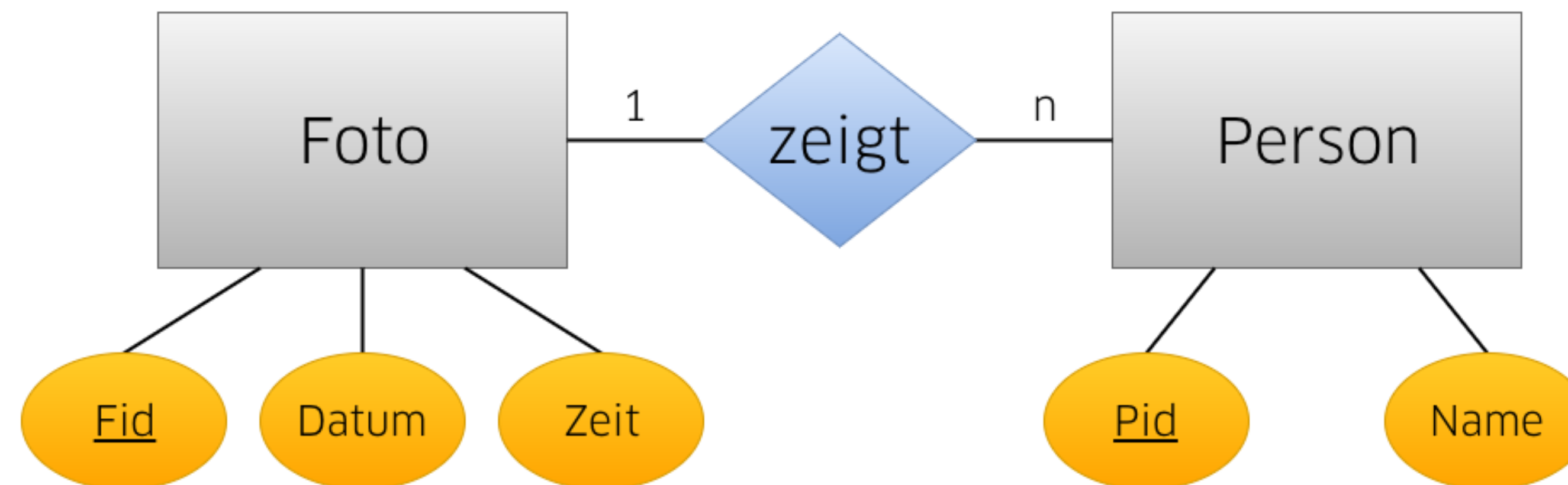
Q&A

- Kardinalitäten?
- Mengensituation?
- Tabellen?

Relationen und Kardinalität

1:n Relation

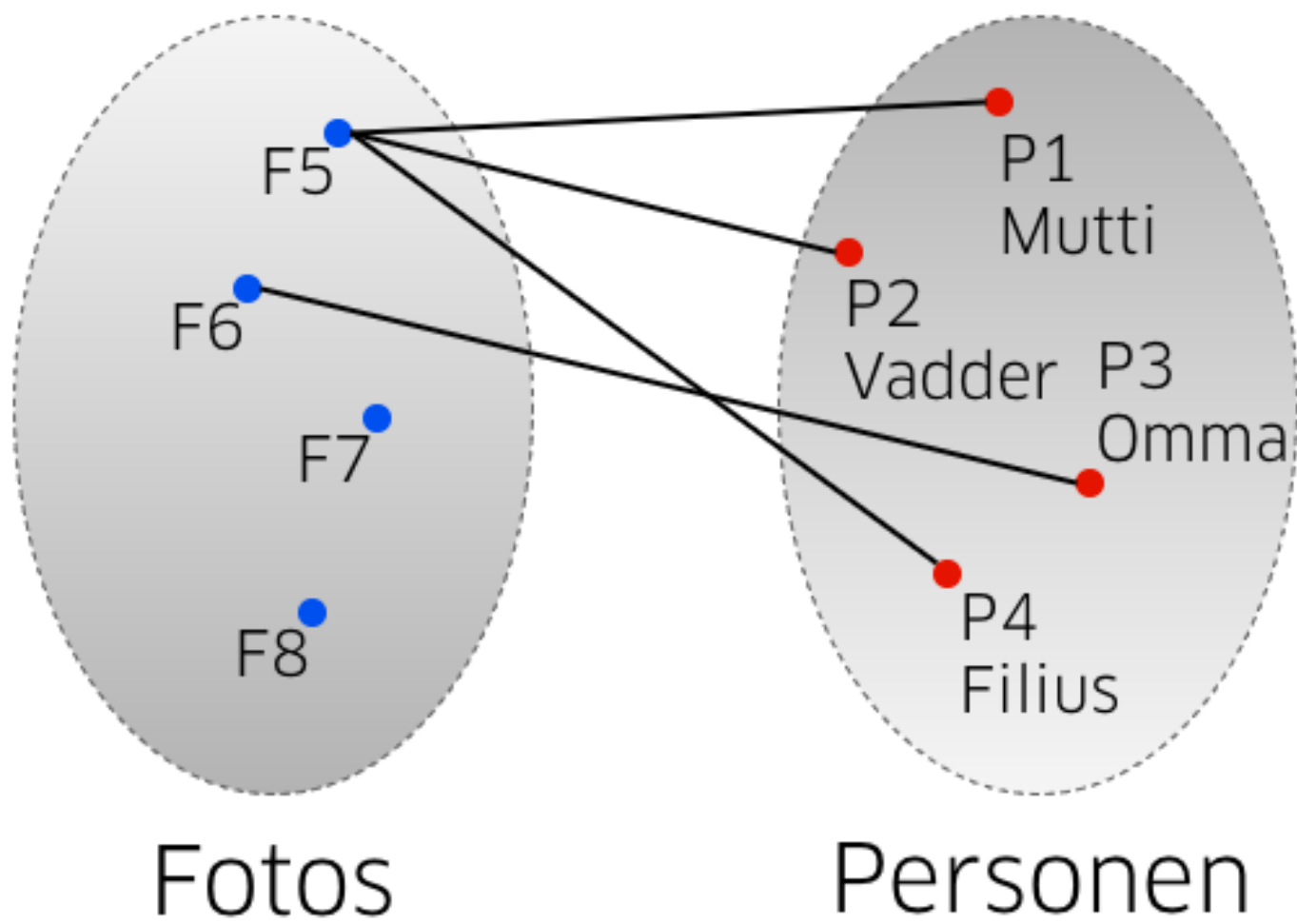
- Ein (fest gewähltes) Foto zeigt mehrere ('n' oder '*') Personen.
- ← Eine (fest gewählte) Person ist nur auf einem ('1') Foto zu sehen.



Relationen und Kardinalität

1:n Relation

- Achtung:** Ein Foto, was beliebig viele Personen zeigt kommt natürlich beliebig häufig in der Relationen-Tabelle vor (z.B. F5). Umgekehrt kommt eine Person hier nur max. einmal vor. Diese Sichtweise ist genau umgekehrt zur Notation.



Ein Foto zeigt beliebig viele Personen aber eine Person ist auf max. einem Foto zu sehen.

Fid	Datum	Zeit	Pid	Name	Fid	Pid
F5	26.12.18	15:00	P1	Mutti	F5	P1
F6	26.12.18	15:05	P2	Vadder	F5	P2
F7	26.12.18	15:10	P3	Omma	F5	P4
F8	26.12.18	15:15	P4	Filius	F6	P3

Relationen und Kardinalität

Mini-Welt Fotoshooting III

Hintergrund: Auf einer Familienfeier [...]

Am Sylvester wird die Regel umgedreht. Das bedeutet, ein Foto zeigt max. eine Person, aber diese kann auf beliebig vielen Fotos zu sehen sein.

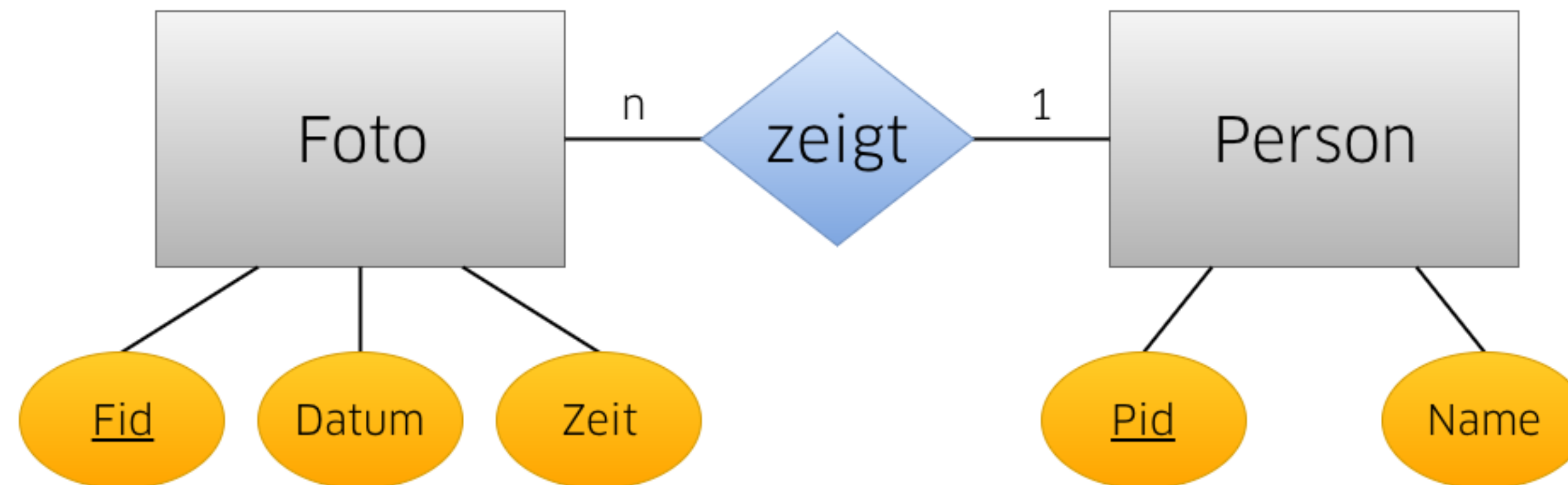
Q&A

- Kardinalitäten?
- Mengensituation?
- Tabellen?

Relationen und Kardinalität

n:1 Relation

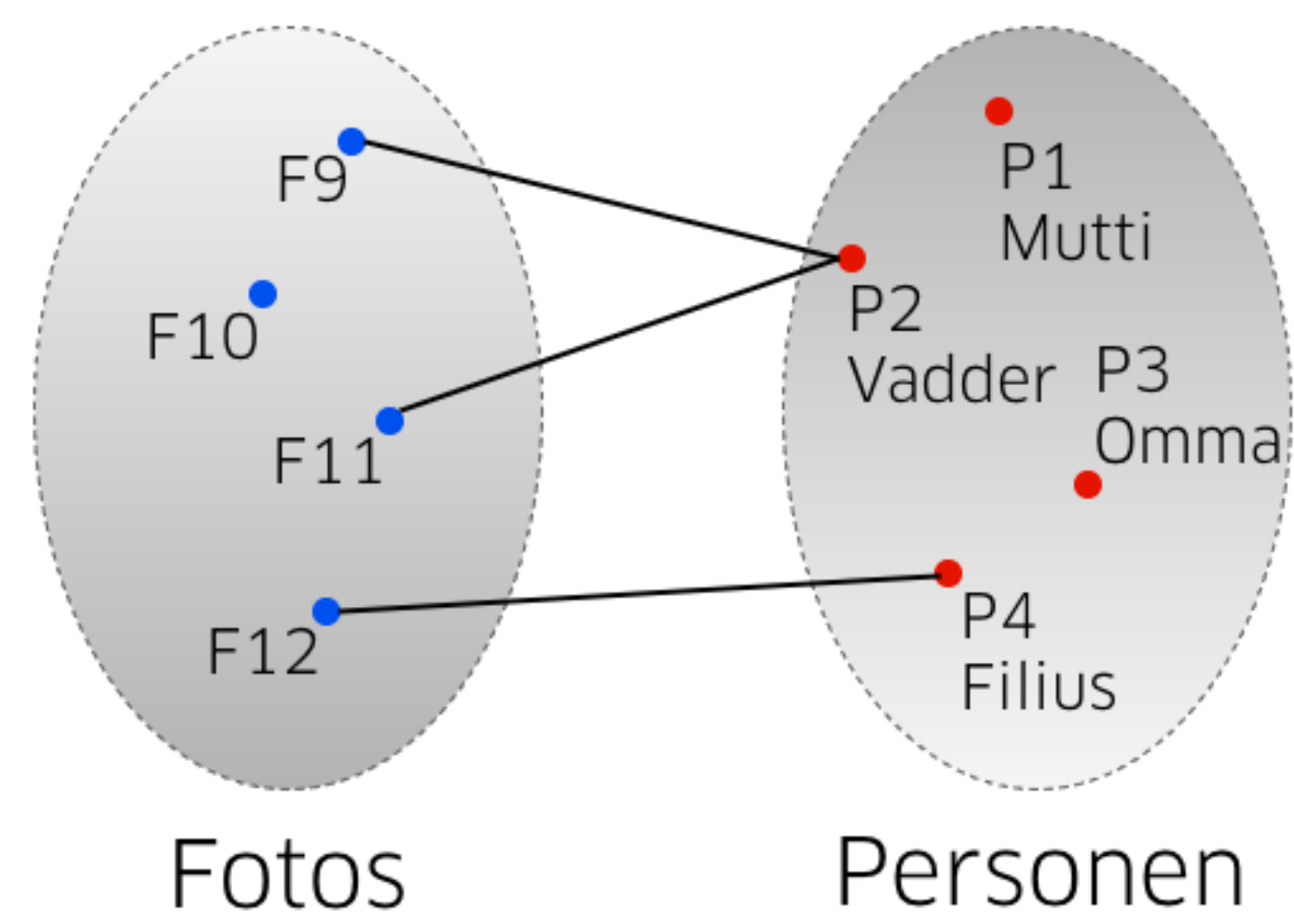
- Ein (fest gewähltes) Foto zeigt genau eine ('1') Person.
- ← Eine (fest gewählte) Person ist auf mehreren ('n' oder '*') Fotos zu sehen.



Relationen und Kardinalität

n:1 Relation

- **Achtung:** Ein Foto kommt in der Relationen-Tabelle max. einmal vor, wogegen die Personen beliebig oft vorkommen (z.B. F2).



Ein Foto zeigt max. eine Person
aber diese kann auf beliebig
vielen Fotos zu sehen sein.

Fid	Datum	Zeit	Pid	Name	Fid	Pid
F9	31.12.18	23:45	P1	Mutti	F9	P2
F10	31.12.18	23:50	P2	Vadder	F11	P2
F11	31.12.18	23:55	P3	Omma	F12	P4
F12	31.12.18	23:59	P4	Filius		

Relationen und Kardinalität

Mini-Welt Fotoshooting IV

Hintergrund: Auf einer Familienfeier [...]

An Neujahr ist alles erlaubt, d.h. ein Foto zeigt beliebig viele Personen und eine Person kann auf beliebig vielen Fotos zu sehen sein.

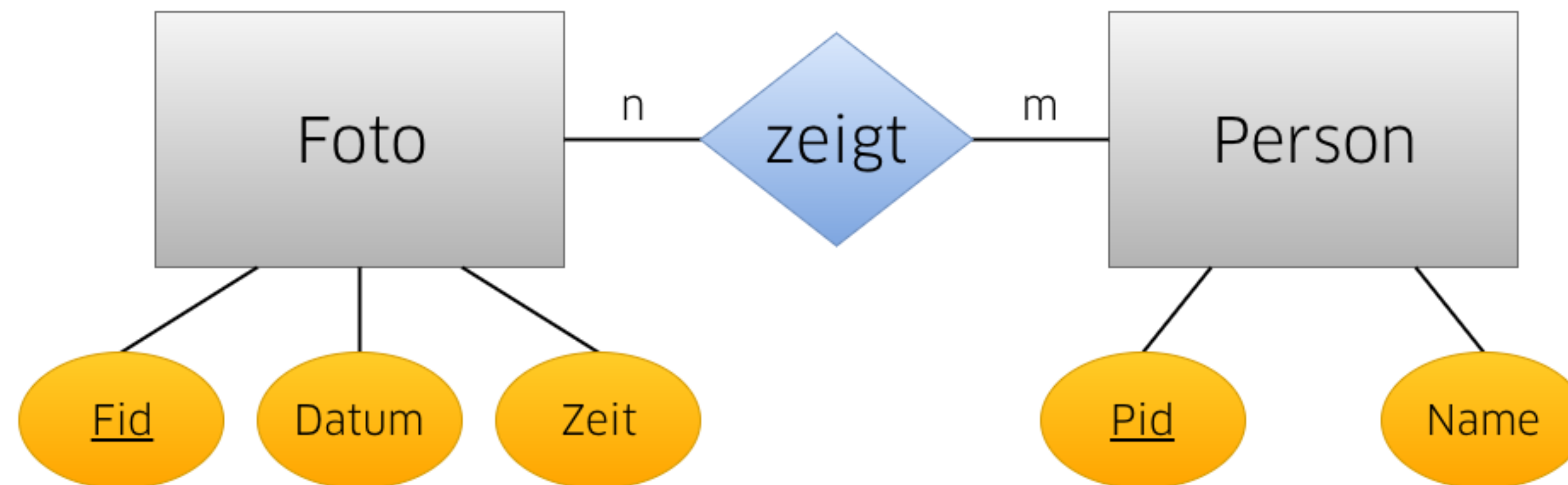
Q&A

- Kardinalitäten?
- Mengensituation?
- Tabellen?

Relationen und Kardinalität

n:m Relation

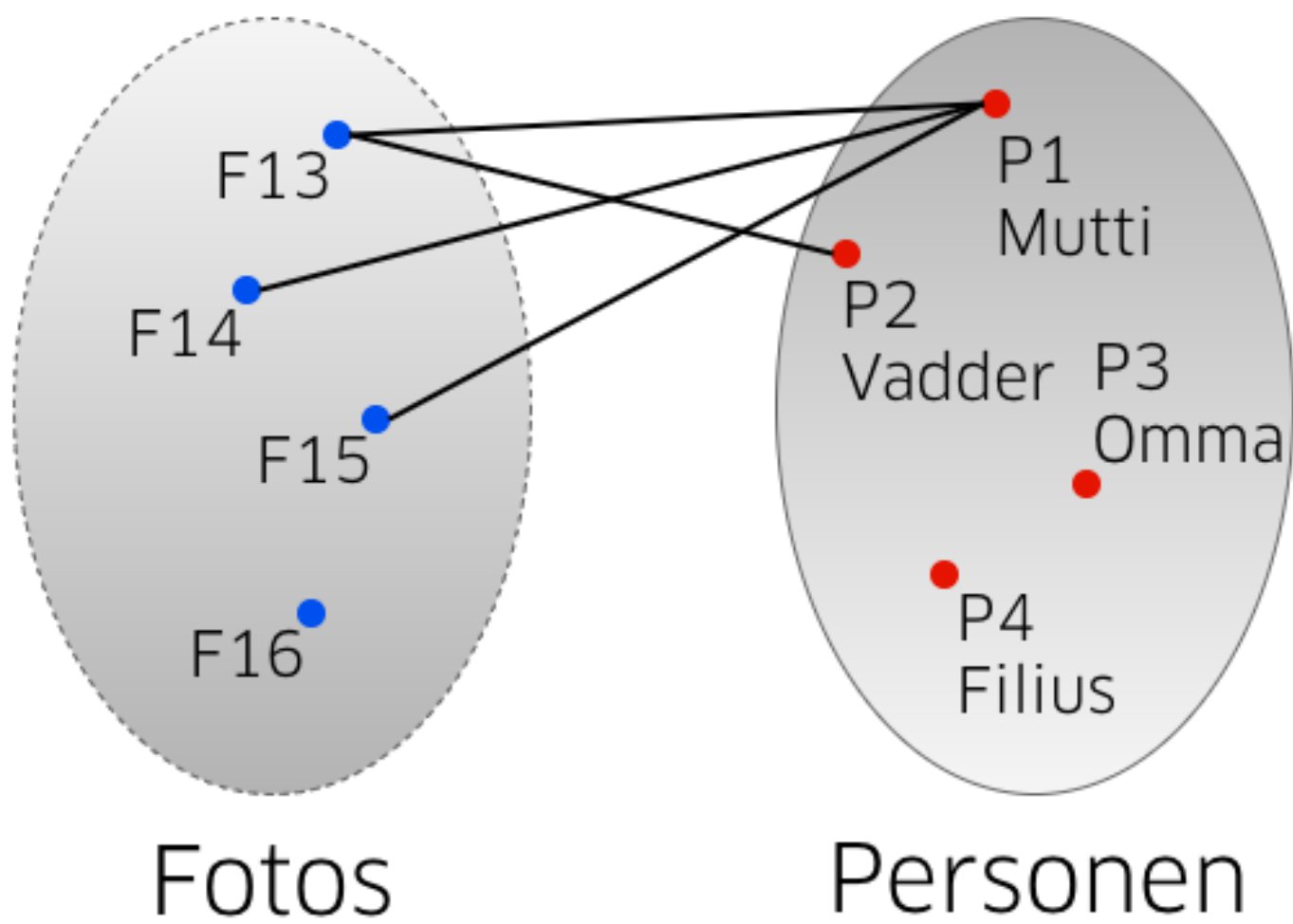
- Ein (fest gewähltes) Foto zeigt beliebig viele ('m' oder '*') Personen.
- ← Eine (fest gewählte) Person ist auf mehreren ('n' oder '*') Fotos zu sehen.



Relationen und Kardinalität

n:m Relation

- Achtung:** Fotos und Personen kommen in der Relationen-Tabelle beliebig oft vor (z.B. F13, P1).

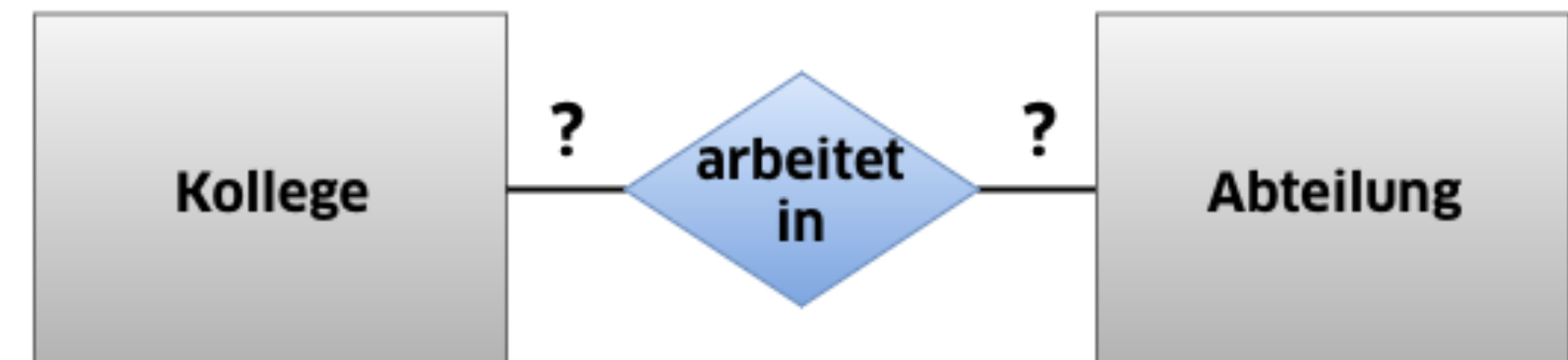
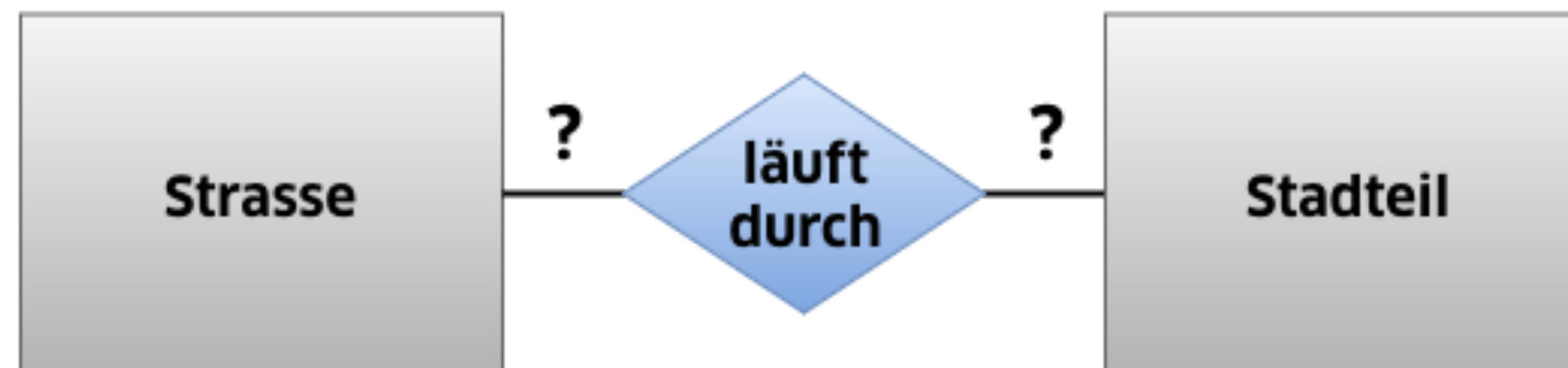
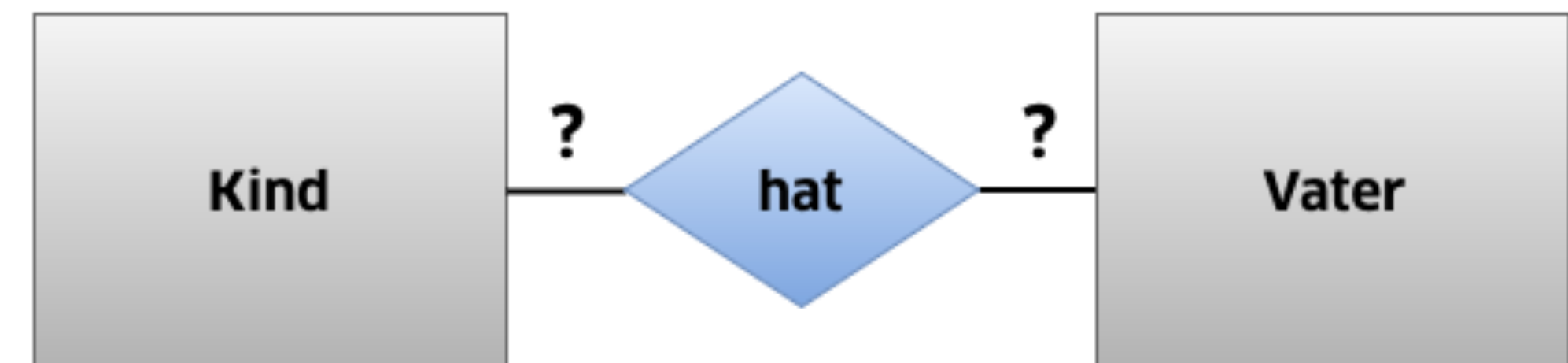
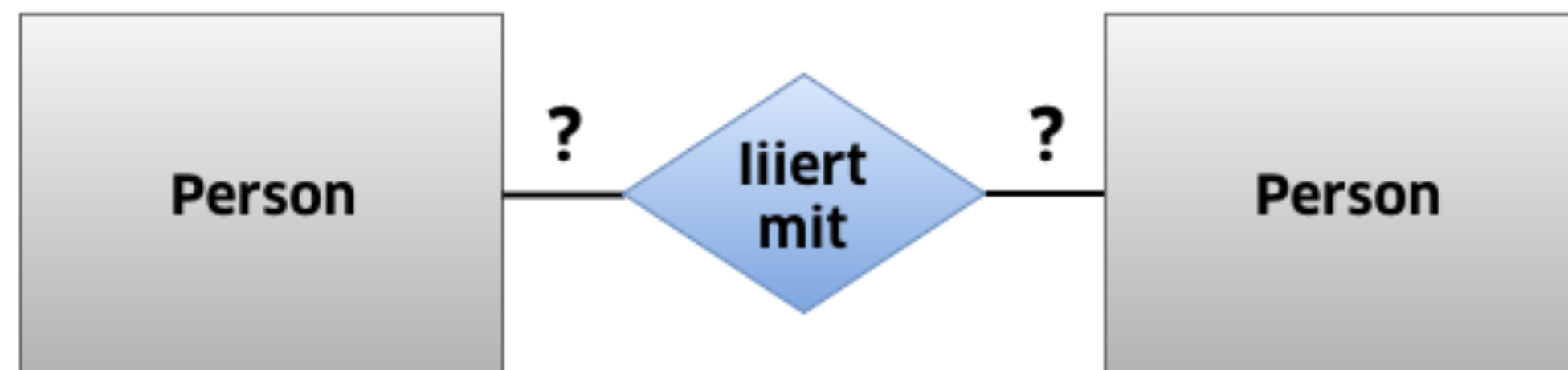


Ein Foto zeigt beliebig viele Personen und eine Person kann auf beliebig vielen Fotos zu sehen sein.

Fid	Datum	Zeit	Pid	Name	Fid	Pid
F13	01.01.18	13:01	P1	Mutti	F13	P1
F14	01.01.18	13:02	P2	Vadder	F13	P2
F15	01.01.18	13:03	P3	Omma	F14	P1
F16	01.01.18	13:04	P4	Filius	F15	P1

Relationen und Kardinalität

Check Chen-Notation



Relationen und Kardinalität

Pro/Contra Chen-Notation

Pro

- Einfache Darstellung.
- 1:1, 1:n, n:1, n:m reicht oft für die Modellierung aus.

Contra

- Ungenau, insbesondere ist oftmals interessant, ob 0 erlaubt ist.

Q&A

- Wie ginge es besser?

Zusammenfassung

Was ist wichtig?

- Ablauf einer Modellierung, insbes. Trennung der Ebenen (ANSI/SPARC-Architektur).
- Rolle der ER-Diagramme und Abgrenzung zur Abbildung in die Datenbankstrukturen.
- ER-Diagramme, Symbolik, Chen-Notation.

Was folgt?

- Fortsetzung ER-Diagramme mit Angabe der Kardinalität in der Min-Max-Notation und der UML-Notation.
- Implementationsentwurf und 'Relationales Modell'.