

Power Designer (Sybase)

- Modellierung innerhalb eines Workspace (Modellierung unabhängig von einer Datenbank)
- Es können konzeptionelle, objektorientierte und physische Modelle erzeugt werden
- Forward- und Revers Engineering ist möglich
- Die Umsetzung vom konzeptionellen ins physische (relationale) Modell kann über ODBS oder SQL-DDL-Anweisungen geschehen
- Die Einarbeitung in die Bedienung des graphischen Tools zur Erzeugung eines ER-Modells ist einfach und kann intuitiv geschehen.

Power Designer

- ProcessAnalyst: Erfassung von Betriebsabläufen und Analyse von Organisationsstrukturen (Einsatz von Datenflußdiagrammen (DFD))
- DataArchitect: Werkzeug zum Entwurf relationaler Datenbanken (Einsatz von Entity Relationship Modellen (ERM))
- AppModeler: Erstellen von Objekten für VB, PowerBuilder, Power++, Delphi, HTML und JavaScript auf Basis von Datenmodellen
- WarehouseArchitect: Unterstützung bei der DataWarehouse-Planung
- MetaWorks: Komponentenübergreifende Verwaltung der Modellierdaten
- Viewer: Nur-Lese-Komponente zur Anzeige von mit dem PowerDesigner erstellten Projektergebnissen

ER-Notation – Entity und Relationship

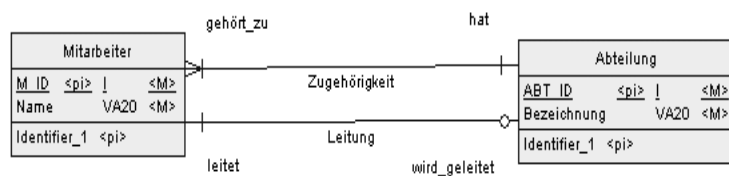
Entity:

Mitarbeiter		
<u>m_id</u>	<pi>	<M>
name	VA20	<M>
Identifizier_1	<pi>	

<pi> Primärschlüssel
 | Datentyp (integer)
 M zwingend (mandantory)
 Identifizier_1 → autom. erzeugt

Relationship:

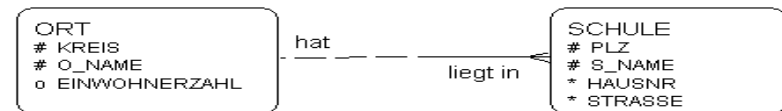
1:n und 1:1



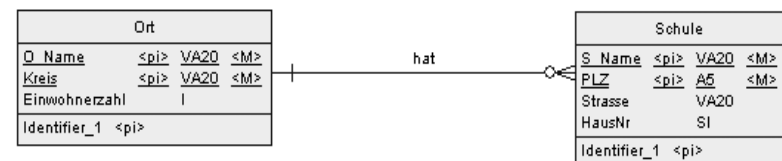
ER-Notation - Vergleich zum Oracle Designer

Vergleich - Kardinalitäten

Oracle Designer

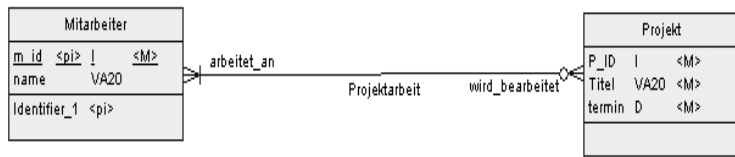


Power Designer

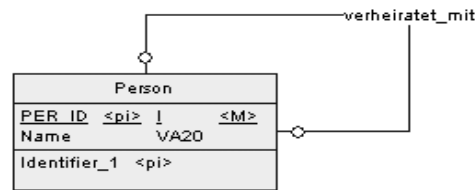


ER-Notation - Relationship

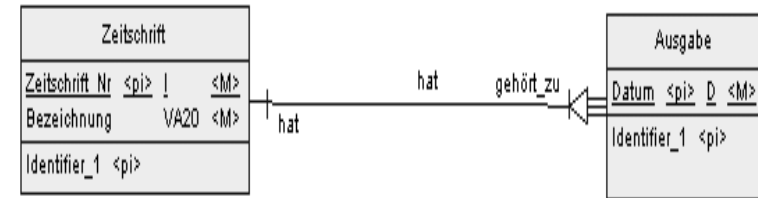
Relationship: n:m



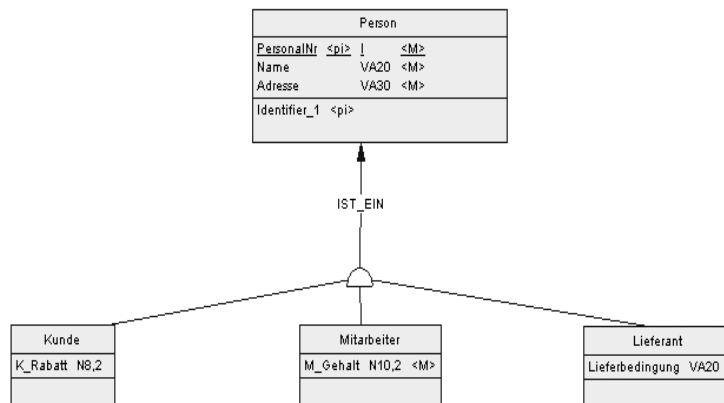
rekursiv:



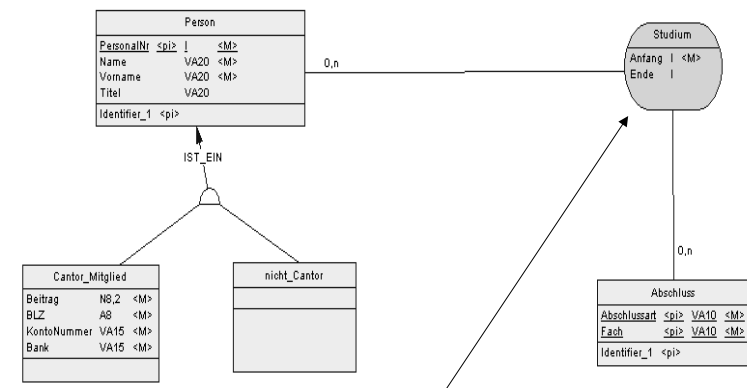
ER – Notation - Weak-Entity



ER – Notation - Spezialisierung



ER – Notation - Association-Entity



Spezielles Element zur Darstellung von Relationships mit Attribut – im Oracle Designer durch zweimal Weak-Entity dargestellt

Umsetzung ins Relationale Modell (1)

1. Schritt – ER-Diagramm - abspeichern



2. Schritt – Korrektheit überprüfen

Tools – Check Models

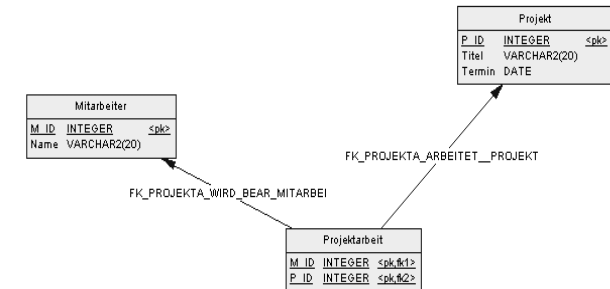
Umsetzung ins Relationale Modell (2)

3. Schritt – Übersetzen ins Physische Modell

Tools – Generate Physical Data Model

Es entsteht ein File *name.pdm*, welches das Physische Modell (entspricht dem Server-Modell vom Oracle Designer) enthält.

Beispiel:



Umsetzung ins Relationale Modell (3)

4. Schritt – Generieren des Skriptes

Database Database Generation

Verschiedene Einstellungen sind möglich, auch die Einstellung des

Zielverzeichnis. Es entsteht ein Skript. Beispiel:

```
alter table PROJEKTARBEIT
drop constraint FK_PROJEKTA_ARBEITET__PROJEKT;
alter table PROJEKTARBEIT
drop constraint FK_PROJEKTA_WIRD_BEAR_MITARBEI;
drop table MITARBEITER cascade constraints;
drop table PROJEKT cascade constraints;
drop table PROJEKTARBEIT cascade constraints;
create table MITARBEITER (
M_ID          INTEGER          not null,
NAME          VARCHAR2(20)     not null,
constraint PK_MITARBEITER primary key (M_ID)
);
create table PROJEKT (
P_ID          INTEGER          not null,
TITEL         VARCHAR2(20)     not null,
TERMIN        DATE             not null,
constraint PK_PROJEKT primary key (P_ID)
);
```

```
create table PROJEKTARBEIT (
M_ID          INTEGER          not null,
P_ID          INTEGER          not null,
constraint PK_PROJEKTARBEIT
primary key (M_ID, P_ID));

alter table PROJEKTARBEIT
add constraint
FK_PROJEKTA_ARBEITET__PROJEKT
foreign key (P_ID) references PROJEKT (P_ID);

alter table PROJEKTARBEIT
add constraint
FK_PROJEKTA_WIRD_BEAR_MITARBEI
foreign key (M_ID)
references MITARBEITER (M_ID);
```

Umsetzung ins Relationale Modell (4)

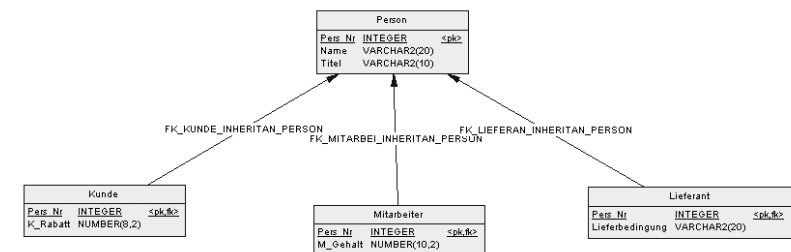
Übersetzung von Spezialisierungen

1. und 2. Schritt wie bei anderen Modellen

3. Schritt auf das Spezialisierungssymbol (Halbkreis) doppelklicken unter Generation verschiedene Modi einstellen

- Generate Parent
- Generate Children
- Generate beides
- Vererbung der Attribute

4. Schritt – Übersetzen ins Physische Modell



Umsetzung ins Relationale Modell (5)

Übersetzung von Spezialisierungen

6. Schritt – Generieren des Skriptes

Database Database Generation Beispiel:

```

create table KUNDE (
  PERS_NR      INTEGER          not null,
  K_RABATT     NUMBER(8,2),
  constraint PK_KUNDE primary key (PERS_NR)
);
create table LIEFERANT (
  PERS_NR      INTEGER          not null,
  LIEFERBEDINGUNG VARCHAR2(20),
  constraint PK_LIEFERANT primary key (PERS_NR)
);
create table MITARBEITER (
  PERS_NR      INTEGER          not null,
  M_GEHALT     NUMBER(10,2)     not null,
  constraint PK_MITARBEITER primary key (PERS_NR)
);
create table PERSON (
  PERS_NR      INTEGER          not null,
  NAME         VARCHAR2(20)     not null,
  TITEL        VARCHAR2(10),
  constraint PK_PERSON primary key (PERS_NR)
);

alter table KUNDE
  add constraint FK_KUNDE_INHERITAN_PERSON
  foreign key (PERS_NR)
  references PERSON (PERS_NR);

alter table LIEFERANT
  add constraint FK_LIEFERAN_INHERITAN_PERSON
  foreign key (PERS_NR)
  references PERSON (PERS_NR);

alter table MITARBEITER
  add constraint FK_MITARBEI_INHERITAN_PERSON
  foreign key (PERS_NR)
  references PERSON (PERS_NR);

```

Umsetzung ins Relationale Modell (5)

Association - Entity

Keine Besonderheiten bei der Umsetzung

Physisches Modell (Beispiel):

