

Datenmodellierung

VU 184.685, WS 2015

Datenbankentwurf

Sebastian Skritek

Institut für Informationssysteme
Technische Universität Wien



Acknowledgments

Die Folien sind eine kleine Erweiterung der Folien von [Katrin Seyr](#).

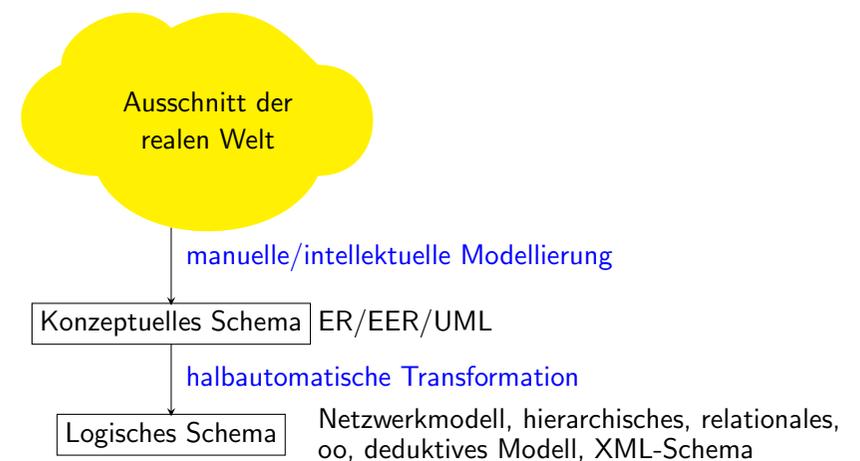


Überblick

- Wiederholung: Datenmodellierung
- Allgemeine Entwurfsmethodik
- Datenbankentwurfsschritte
 - Anforderungsanalyse
 - Konzeptueller Entwurf
 - Implementationsentwurf (Kap. 3)
 - Physischer Entwurf (VU Datenbanksysteme)

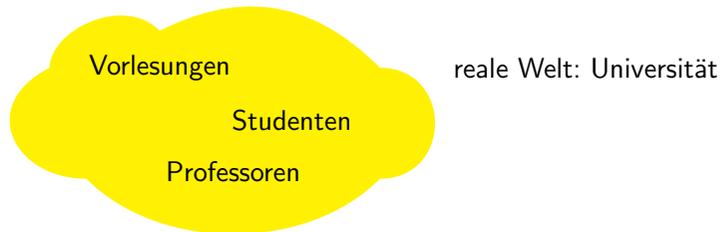


Wiederholung Datenmodellierung

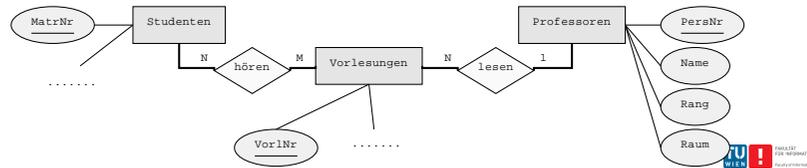


Wiederholung: Bsp Datenmodellierung

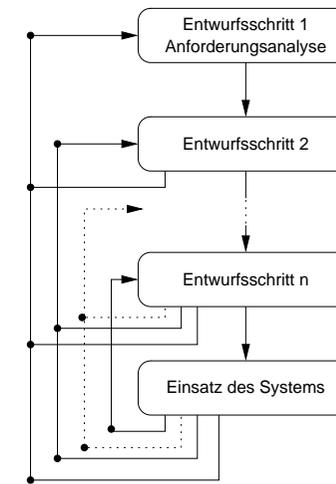
1 Abgrenzung der zu modellierenden Welt



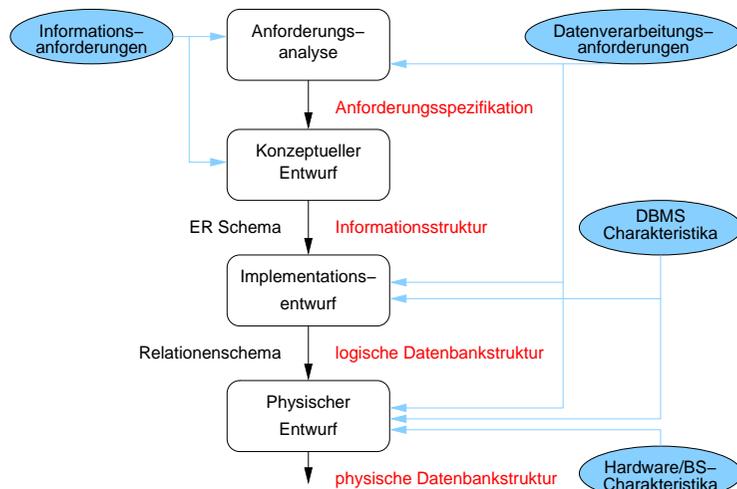
2 Überführung der zu modellierenden Welt in ein Konzeptuelles Schema (in der Vorlesung: EER)



Allgemeine Entwurfsmethodik



Datenbankentwurfsschritte



Anforderungsanalyse

- 1 Identifikation von Organisationseinheiten
- 2 Identifikation der zu unterstützenden Aufgaben
- 3 Anforderungs-Sammelplan
- 4 Anforderungs-Sammlung
- 5 Filterung
(Prüfung der Information auf Verständlichkeit und Eindeutigkeit)
- 6 Satzklassifikation
(Objekte, Beziehungen zwischen Objekten, Operationen, Ereignisse)
- 7 Formalisierung - Erstellung des Pflichtenhefts, aufgebaut wie folgt:
 - Informationsstrukturanforderungen: strukturierte Information über
 - Objekte
 - Attribute
 - Beziehungen
 - Datenverarbeitungsanforderungen: strukturierte Information über Prozessbeschreibungen

Objekt- und Attributbeschreibung

Objekte

- Uni-Angestellte:
 - Anzahl: 1000
 - Attribute:
 - Personalnummer
 - Name
 - Gehalt
 - Rang
- Studenten:
 - Anzahl: 20.000
 - Attribute:
 - Matrikelnummer
 - Name
 - Adresse

Attribute

- Personalnummer
 - Typ: char
 - Länge: 9
 - Wertebereich: 0...999.999
 - Anzahl Wiederholungen: 0
 - Definiertheit: 100%
 - Identifizierend: ja
- Gehalt
 - Typ: dezimal
 - Länge: (8,2)
 - Anzahl Wiederholung: 0
 - Definiertheit: 90%
 - Identifizierend: nein

Beziehungsbeschreibung

Beziehung: *prüfen*

- Beteiligte Objekte:
 - Professor als Prüfer
 - Student als Prüfling
 - Vorlesung als Prüfungstoff
- Attribute der Beziehung:
 - Datum
 - Uhrzeit
 - Note
- Anzahl: 100 000 pro Jahr

Prozessbeschreibung

Prozess: *Zeugnisausstellung*

- Häufigkeit: halbjährlich
- benötigte Daten:
 - Prüfungen
 - Studienordnungen
 - Studenteninformatio
 - ...
- Priorität: hoch
- zu verarbeitende Datenmenge:
 - 500 Studenten
 - 3000 Prüfungen
 - 10 Studienordnungen

Wiederholung: Überblick

- Wiederholung: Datenmodellierung
- Allgemeine Entwurfsmethodik
- Datenbankentwurfsschritte
 - Anforderungsanalyse
 - Konzeptueller Entwurf
 - Das ER Modell
 - Sichtenintegration und Konsolidierung

Das ER Modell

- 1 Entities und Beziehungen
- 2 Rollen und Attribute
- 3 Schlüssel
- 4 Funktionalitäten
- 5 (min,max) Notation
- 6 Schwache Entities
- 7 Generalisierung oder Spezialisierung (EER)
- 8 Aggregation (EER)

Entities und Beziehungen

Entities: wohlunterscheidbare Konzepte der zu modellierenden Welt

Beziehungen: verknüpfen mehrere Entities

Beispiel

- Entities:**
- Studierende Müller, Maier, Alice, Bob
 - LVAs Datenmodellierung, Formale Modellierung
 - Prüfungen DM 2.2.2016, FMod 12.12.2015
- Beziehungen:**
- Müller hört Datenmodellierung, Maier hört Formale Modellierung
 - Alice schreibt Prüfung FMod 12.12.2015

Entitytypen und Beziehungstypen

Abstrahiere Entities und Beziehungen zu **gleichartigen Typen**

→ Beschreibe Typen und ihre Beziehungen im ER Diagramm

Entitytypen: Rechtecke

Beziehungstypen: Rauten



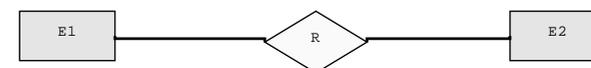
Beispiel



Beziehungen und Beziehungstypen – Formal

n -stellige Beziehung: Ein n -Tupel von Entities

n -stelliger Beziehungstyp: Beschreibt eine Menge von n -stelligen Beziehungen $R \subseteq E_1 \times E_2 \times \dots \times E_n$



$$R \subseteq E_1 \times E_2$$

Beispiel

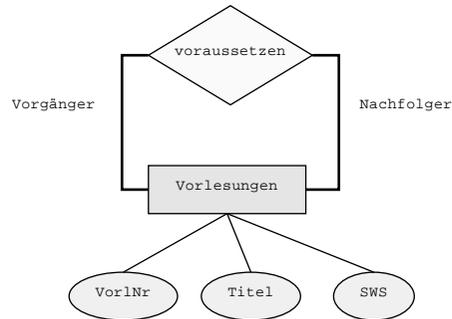


$$\text{hören} \subseteq \text{Student} \times \text{Vorlesungen}$$

Rollen und Attribute

Attribute charakterisieren Entity- bzw. Beziehungstypen.

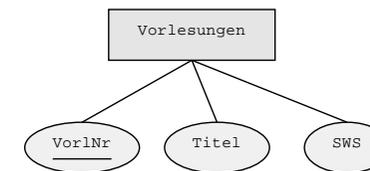
Rollen können verwendet werden, um zu beschreiben, wie die an einer Beziehung beteiligten Entities sich verhalten.



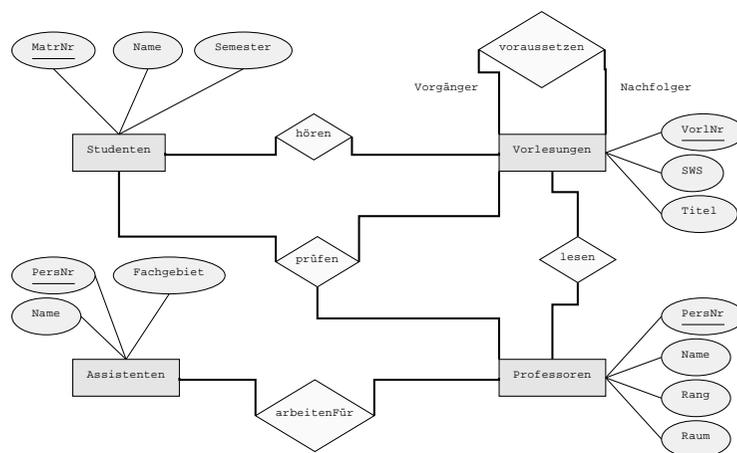
Schlüssel

Schlüssel sind eine **minimale** Menge von Attributen, deren Wert eine Entity eindeutig innerhalb aller Entities eines Typs identifizieren.

- Im ER-Diagramm: Pro Entitytyp wird ein **Schlüssel** markiert.
- Im Schlüssel enthaltenen Attribute werden **unterstrichen**.
- Bei mehreren Schlüsselkandidaten: **wähle einen Primärschlüssel** aus.



Bsp: Universitätsschema



Funktionalitäten von Beziehungstypen

Funktionalitäten Beschreiben, mit wie vielen Entities eine bestimmte Entity in Beziehung stehen kann.

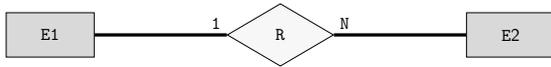


Mögliche Werte:

- 1:1** Jede Entity vom Typ E1 mit **maximal einer** Entity vom Typ E2 **und umgekehrt**
- N:1** Jede Entity vom Typ E1 mit **maximal einer** Entity vom Typ E2, keine Einschränkung für E2
- 1:N** Jede Entity vom Typ E2 mit **maximal einer** Entity vom Typ E1, keine Einschränkung für E1
- N:M** Keine Einschränkungen

Funktionalitäten von Beziehungstypen

- Notation: Bei jedem Entitytyp wird **ein Wert** (1 oder M,N,...) der Funktionalität des Beziehungstyp angeschrieben



- Intuitive Bedeutung:** Fixiert man eine Entity eines Typs, so gibt der Wert auf der Seite des anderen Entitytyp an, wie viele Entities dieses Typs **maximal** mit der fixierten Entity in Beziehung stehen können (M,N, ... : beliebig viele).

Funktionalitäten von Beziehungstypen

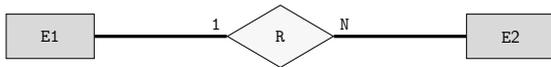


Beispiel

- Wird eine Entity vom Typ E2 fixiert, so kann diese mit maximal **einer** Entity vom Typ E1 in Beziehung stehen.
- Wird eine Entity vom Typ E1 fixiert, so kann diese mit maximal **N** (unbestimmt viele) Entities vom Typ E2 in Beziehung stehen.

Funktionalitäten – Formal

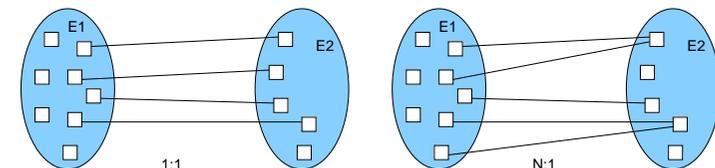
Bei Funktionalitäten 1:1, N:1, und 1:N:
Beziehungstyp beschreiben **partielle Funktionen** zwischen Entities



partiell: Nicht jede Entity muss mit einer anderen Entity in Beziehung stehen.

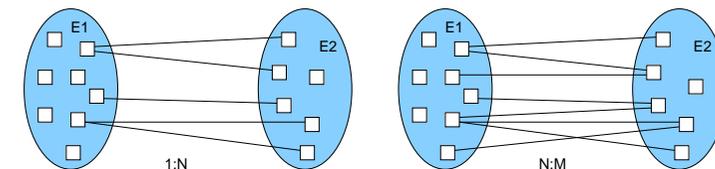
Funktion: Auf der Seite die "1 gegenüber liegt" ist jede an der Beziehung teilnehmende Entity genau einer Entity des anderen Typs zugeordnet.

Durch Funktionalitäten definierte partielle Funktionen



$R : E1 \rightarrow E2$ bzw. $R^{-1} : E2 \rightarrow E1$

$R : E1 \rightarrow E2$



$R : E2 \rightarrow E1$

keine partielle Funktion

Funktionalitäten bei n -stelligen Beziehungstypen

Verallgemeinerung von binären Beziehungstypen

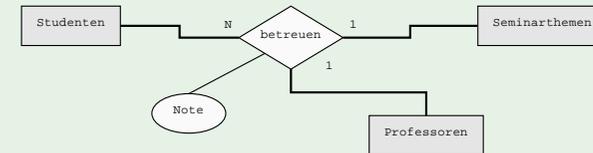
- **Intuitive Bedeutung:** Fixiert man eine Entity von allen beteiligten Typen **bis auf einen**, dann gibt der Wert auf der Seite des fehlenden Typs an wie viele Entities dieses Typs **maximal** mit dem $n - 1$ -Tupel an Entities in Beziehung stehen können.
- **Formal:** Ist R eine Beziehung zwischen mehreren Entities E_1, \dots, E_n , wobei die Funktionalität der Entity $E_k, 1 \leq k \leq n$ mit "1" spezifiziert wird, so wird durch R folgende **partielle Funktion** vorgegeben:

$$R : E_1 \times E_2 \times \dots \times E_{k-1} \times E_{k+1} \times \dots \times E_n \rightarrow E_k$$

Funktionalitäten bei n -stelligen Beziehungstypen

Beispiel

Gegeben ist die ternäre Beziehung *betreuen* zwischen den Entities *Studenten*, *Professoren*, *Seminarthemen*.



betreuen: Professoren \times Studenten \rightarrow Seminarthemen

betreuen: Seminarthemen \times Studenten \rightarrow Professoren

Funktionalitäten bei n -stelligen Beziehungstypen

Beispiel (ctd.)

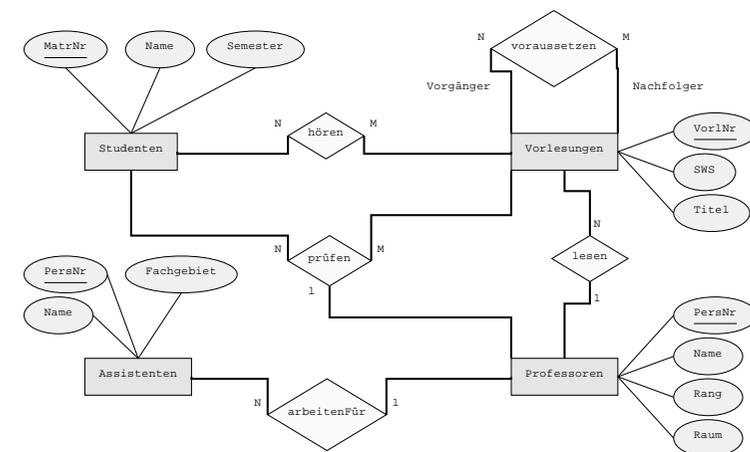
betreuen: Professoren \times Studenten \rightarrow Seminarthemen
bedeutet: Studierende dürfen bei demselben Professor nur ein Seminarthema bearbeiten

betreuen: Seminarthemen \times Studenten \rightarrow Professoren
bedeutet: Studierende dürfen dasselbe Seminarthema nur bei einem Professor bearbeiten (= nicht wiederverwenden)

weiterhin möglich sind:

- Professoren können ein Seminarthema an mehrere Studierende vergeben
- das selbe Seminarthema kann von mehreren Professoren vergeben werden - aber nur an verschiedene Studierende (s.o.)

Bsp: Universitätsschema mit Funktionalitäten



(min,max) Notation

Alternative Notation von Kardinalitäten von Beziehungstypen

Gegeben:

n -stelliger Beziehungstyp R zwischen den Entitytypen E_1, \dots, E_n .

R definiert Relation $R \subseteq E_1 \times \dots \times E_i \times \dots \times E_n$

(min,max) Notation: Gibt für alle Entities eines Typs E_i an, wie oft sie **mindestens/maximal** in R auftreten dürfen.

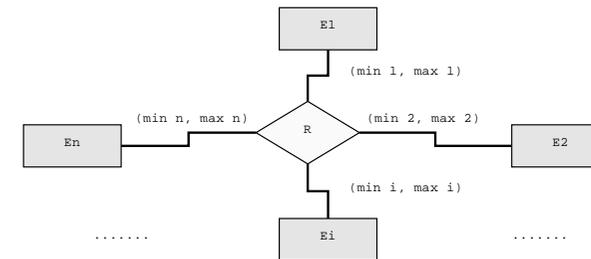
"1:N"-Notation: Gibt für ein bestimmtes $n - 1$ Tupel von Entities an, mit wie vielen Entities des verbliebenen Typs es **maximal** in Verbindung steht.

Ausdruckskraft unvergleichbar!

(min,max) Notation

Gegeben:

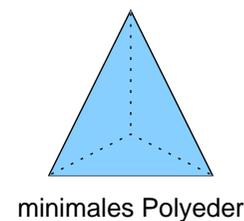
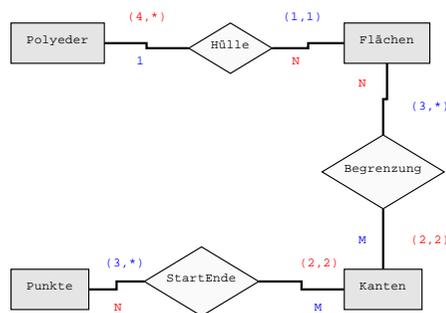
n -stelliger Beziehungstyp R zwischen den Entitytypen E_1, \dots, E_n .



Es gilt: für jede Entity e_i von Typ E_i gibt es

- mindestens min_i Tupel der Art $(\dots, e_i, \dots) \in R$
- maximal max_i Tupel der Art $(\dots, e_i, \dots) \in R$

Bsp: (min,max) Notation

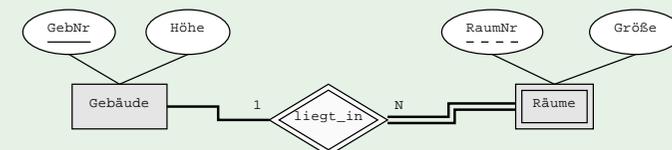


Schwache Entities

Schwache Entities sind Entities, deren Existenz von einer anderen, übergeordneten Entity abhängen und die durch eine Kombination mit dem Schlüssel der übergeordneten Entity identifizierbar sind.

Beispiel

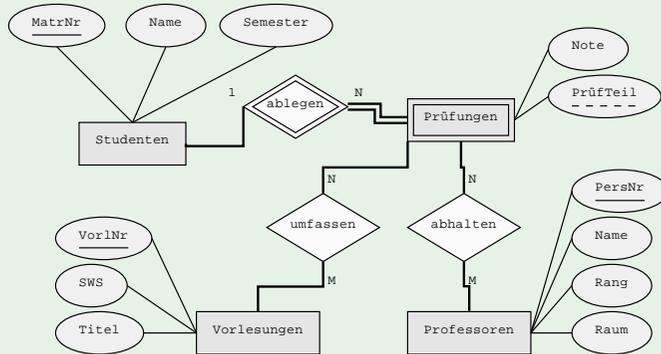
Ist die Nummer eines Raumes nur innerhalb eines Gebäudes eindeutig, so ist der Schlüssel von Räume eine Kombination aus Raum- und Gebäudenummer.



Schwache Entities

Beispiel

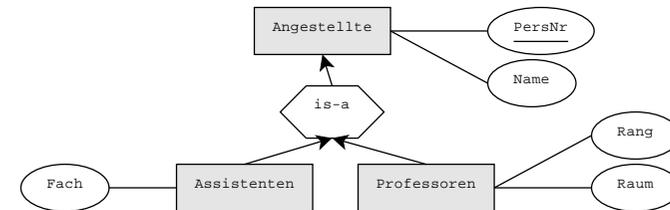
Ein Diplomprüfungsfach besteht aus mehreren Vorlesungen, die alle in Teilprüfungen von Professoren beurteilt werden. Eine Prüfung selbst ist abhängig vom Studierenden, der die Prüfung ablegt.



Generalisierung oder Spezialisierung (EER)

Generalisierung wird verwendet, um eine Strukturierung der Entitytypen zu erzielen:

- gemeinsame Eigenschaften von ähnlichen Entitytypen werden einem Obertyp (O) zugeordnet
- unterschiedliche Eigenschaften verbleiben bei Untertypen (U)

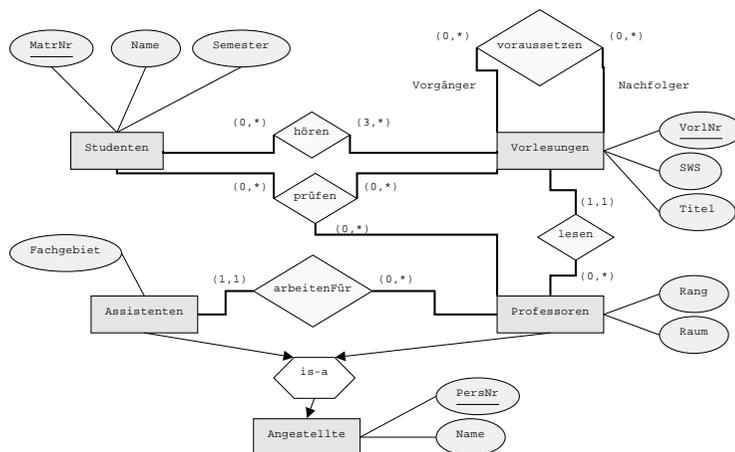


Es gibt:

- disjunkte Generalisierung: $U_1 \cap U_2 = \emptyset$
- vollständige Generalisierung: $U_1 \cup U_2 = O$

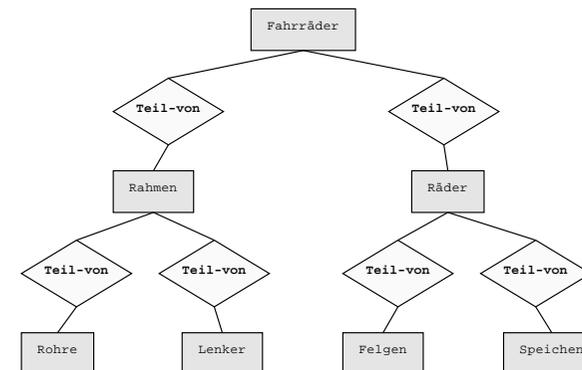


Bsp: Universitätsschema mit (min, max) Notation und Generalisierung



Aggregation (EER)

Die Aggregation ordnet unterschiedliche Entitytypen, die in ihrer Gesamtheit einen strukturierten Objekttyp bilden, einander zu.



Sichtenintegration und Konsolidierung

Bei großen Anwendungen: Aufteilung der Anforderungsanalyse in verschiedene Sichten

Beispiel (Universität)

Mögliche unterschiedliche Sichten:

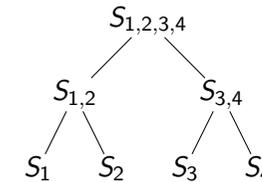
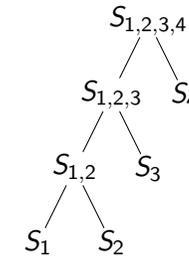
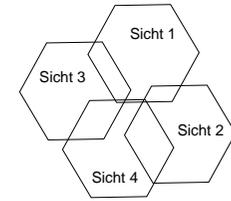
- Lehrendensicht
- Studierendensicht
- Verwaltungssicht
-

Konsolidierung: ein globales Schema wird erstellt, das redundanzfrei, widerspruchsfrei und um Synonyme und Homonyme bereinigt ist.

Konsolidierungsbaum

Mögliche Konsolidierungsbäume:

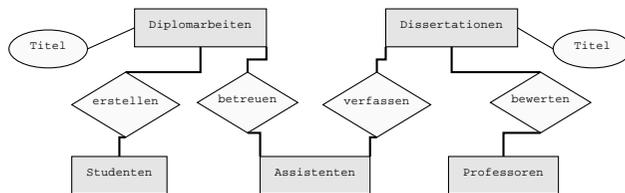
- maximal hoher Konsolidierungsbaum
- minimal hoher Konsolidierungsbaum



Bsp: Konsolidierung

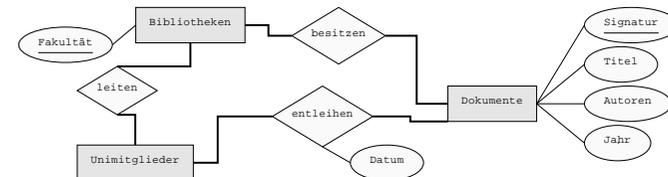
Gegeben: 3 Sichten ...

Sicht 1: Dokumenterstellung als Prüfungsleistung

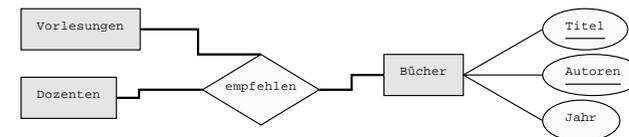


Bsp: Konsolidierung

Sicht 2: Bibliotheksverwaltung



Sicht 3: Buchempfehlung für Vorlesungen



Bsp: Konsolidierung

... und folgende Informationen

- Dissertationen, Diplomarbeiten und Bücher sind Spezialisierungen von Dokumenten der Bibliotheken.
- Alle Dokumente werden durch die Signatur identifiziert.
- Unimitglieder ist Generalisierung von Studenten, Professoren und Assistenten.
- Synonyme Verwendung von Dozenten und Professoren.
- Alle Diplomarbeiten und Dissertationen werden in Bibliotheken verwaltet.
- Fakultätsbibliothek: geleitet von Angestellten - Revision von leiten bei Spezialisierung von Unimitglieder.
- Die Beziehungen erstellen und verfassen aus Sicht 1 entsprechen den Autoren von Büchern in Sicht 3.

Bsp: Konsolidierung

