

Einführung in Datenbanken

Ralf Möller, TU Hamburg Harburg

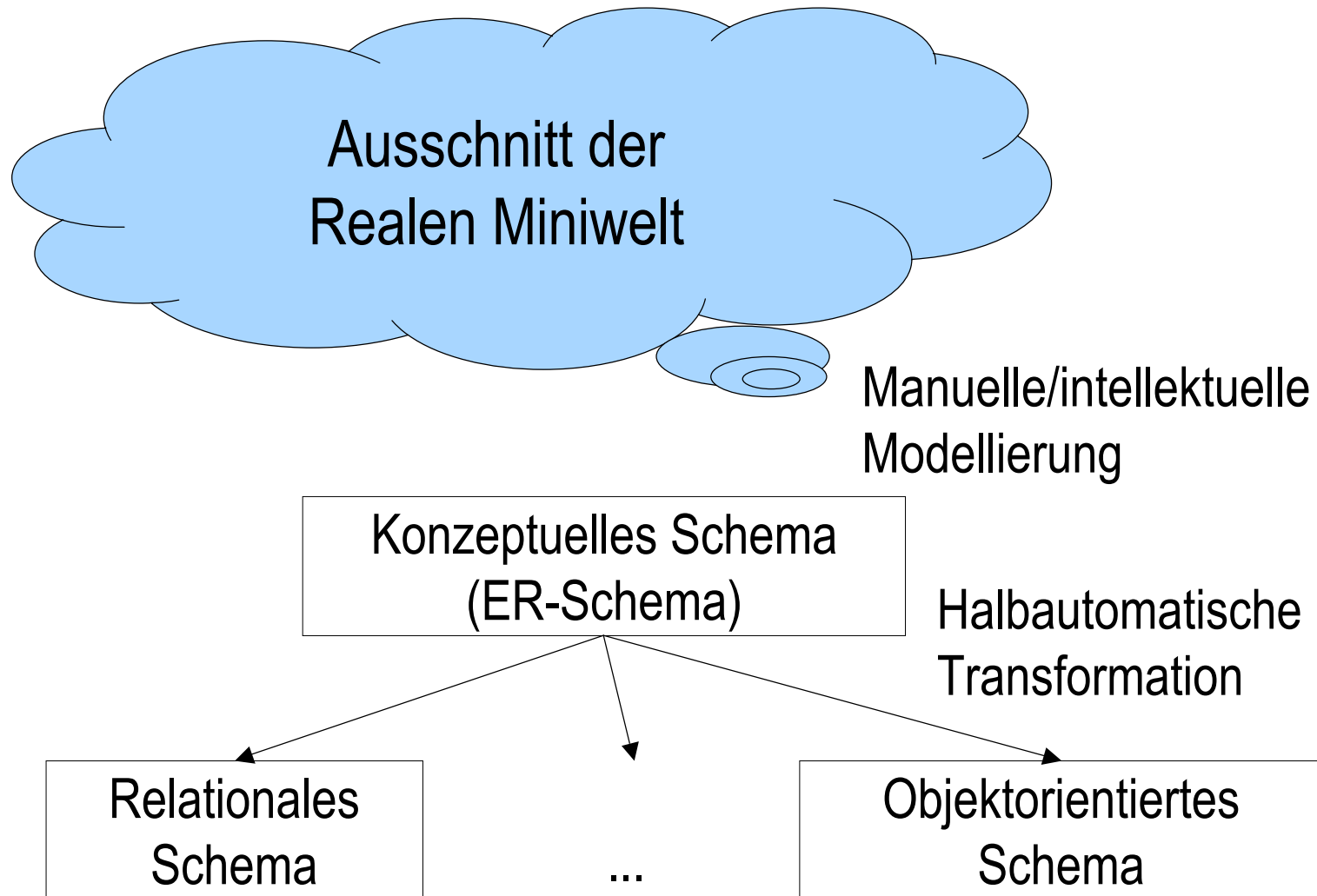
- Voraussetzungen:
 - Einführung in Entity-Relationship-Modellierung
- Lernziele heute:
 - Vertiefung der ER-Modellierung
- Vorbereitung für
 - Relationales Datenmodell
 - Objektorientiertes Datenmodell

Literatur und Danksagung

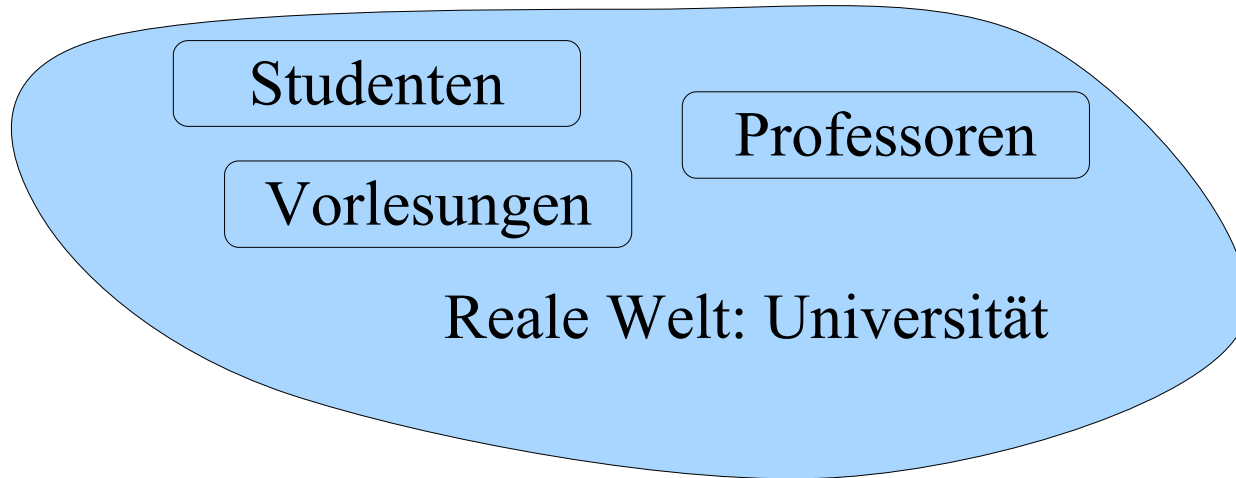
- A. Kemper, A. Eickler, Datenbanksysteme: Eine Einführung
- Diese Vorlesung basiert auf Präsentationsmaterial zu diesem Buch



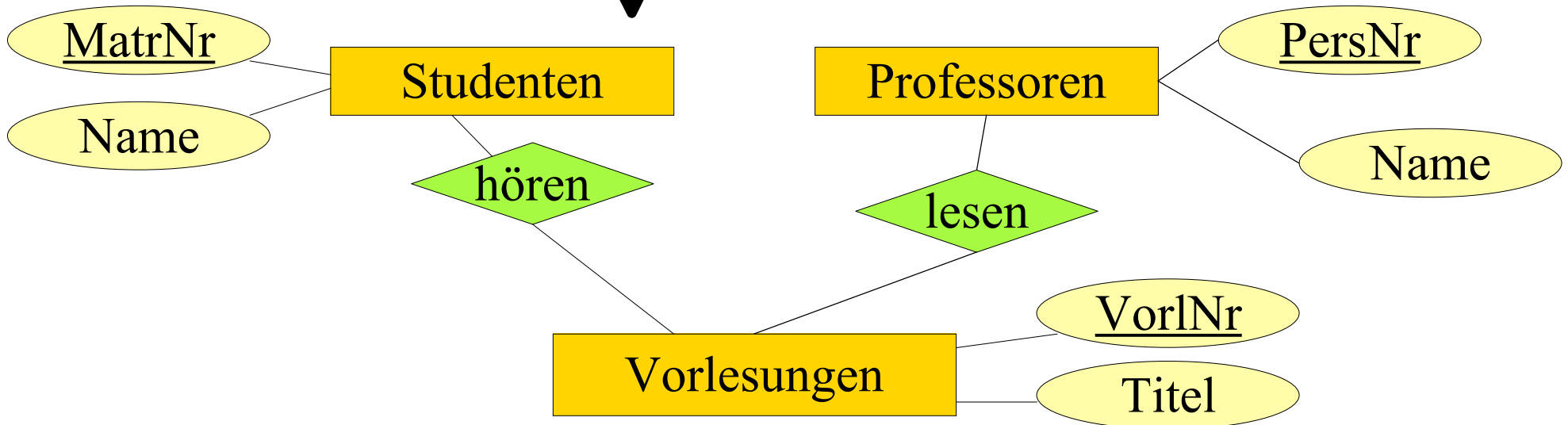
Datenmodellierung



Modellierung einer kleinen Beispielanwendung

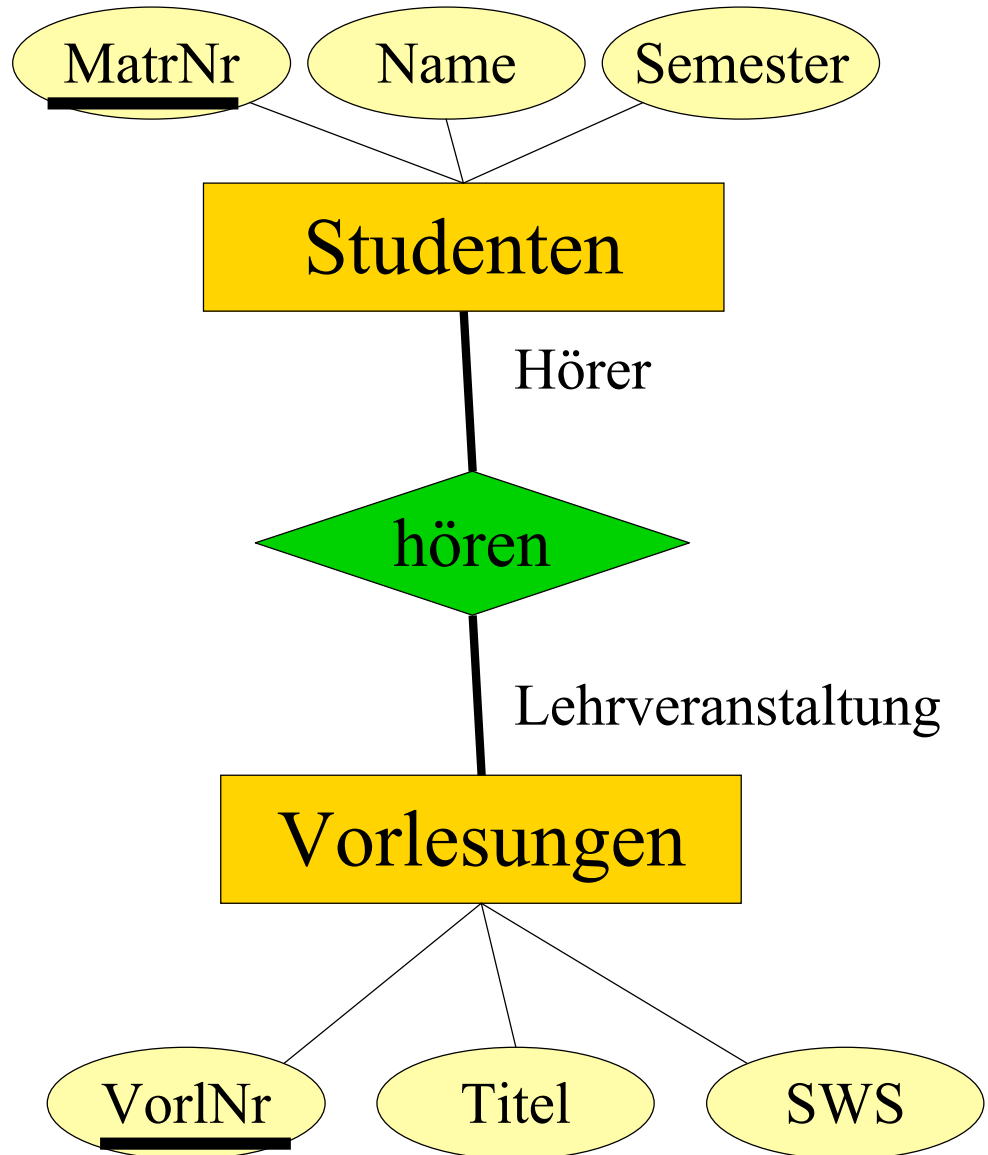


↓ Konzeptuelle Modellierung

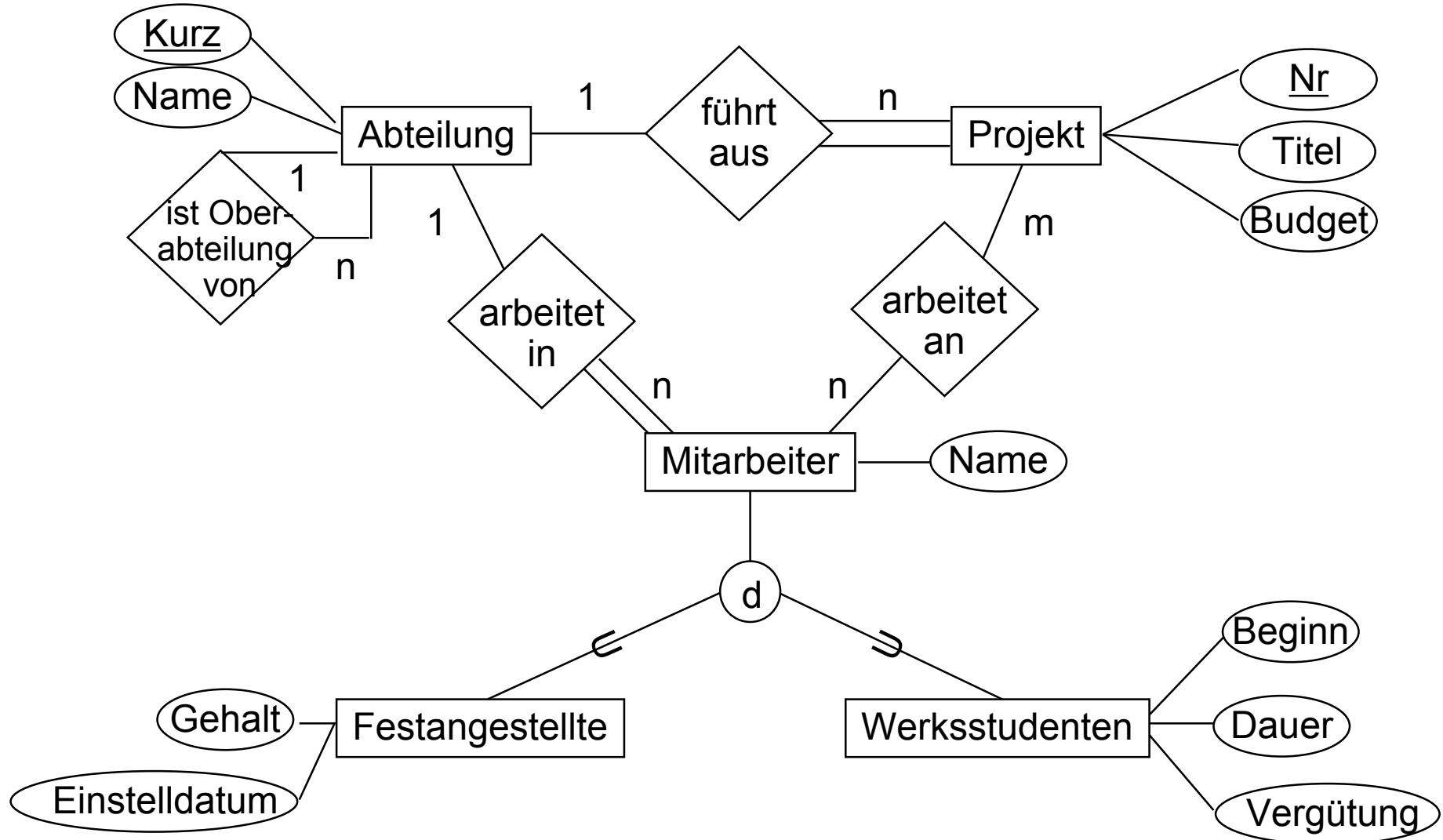


Entity/Relationship-Modellierung

- Entity (Gegenstandstyp)
- Relationship (Beziehungstyp)
- Attribut (Eigenschaft)
- Schlüssel (Identifikation)
- Rolle



Beispiel: ER-Diagramm

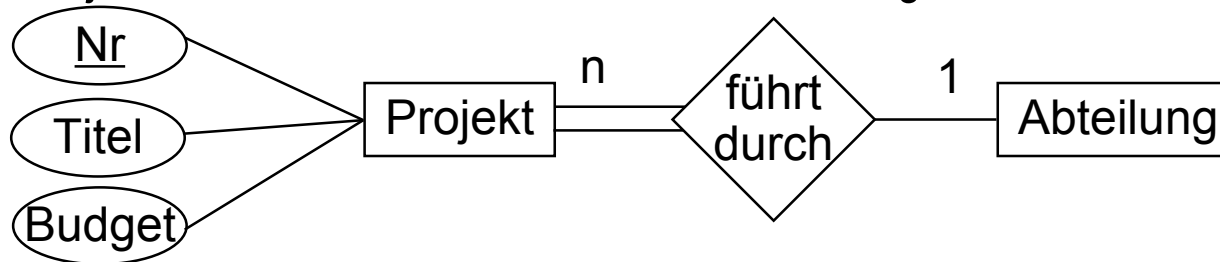


Identifikation und Schlüssel (1)

- Zur **Identifikation** existieren zwei grundlegende Ansätze in Datenbankmodellen:
 - **Referentielle Identifikation** bezeichnet direkte Verweise auf Objekte (Zeiger in Programmiersprachen).
 - **Assoziative Identifikation** verwendet die Werte von Attributen oder Attributkombinationen, um sich eindeutig auf Objekte zu beziehen (Schlüssel: Ausweisnummer, Fahrgestellnummer, ...).
 - In der Praxis benötigt man häufig beide Formen der Identifikation.
- **Schlüssel:**
 - Schlüssel sind Attribute oder Attributkombinationen mit innerhalb einer Klasse eindeutigen Werten und eignen sich deshalb zur Identifikation.
 - Es kann mehrere Schlüsselkandidaten geben (Primärschlüssel, Sekundärschlüssel).
 - Schlüssel stellen als Attributwerte Beziehungen zu anderen Objekten her (Fremdschlüssel).
 - Durch Fremdschlüssel referenzierte Objekte müssen existieren (referentielle Integrität).

Identifikation und Schlüssel (2)

- **Beispiel:** Projekte können durch eine Nummer eindeutig identifiziert werden.



Dabei existieren zwei Möglichkeiten zur Identifikation von Projekten innerhalb der Assoziation "führt durch":

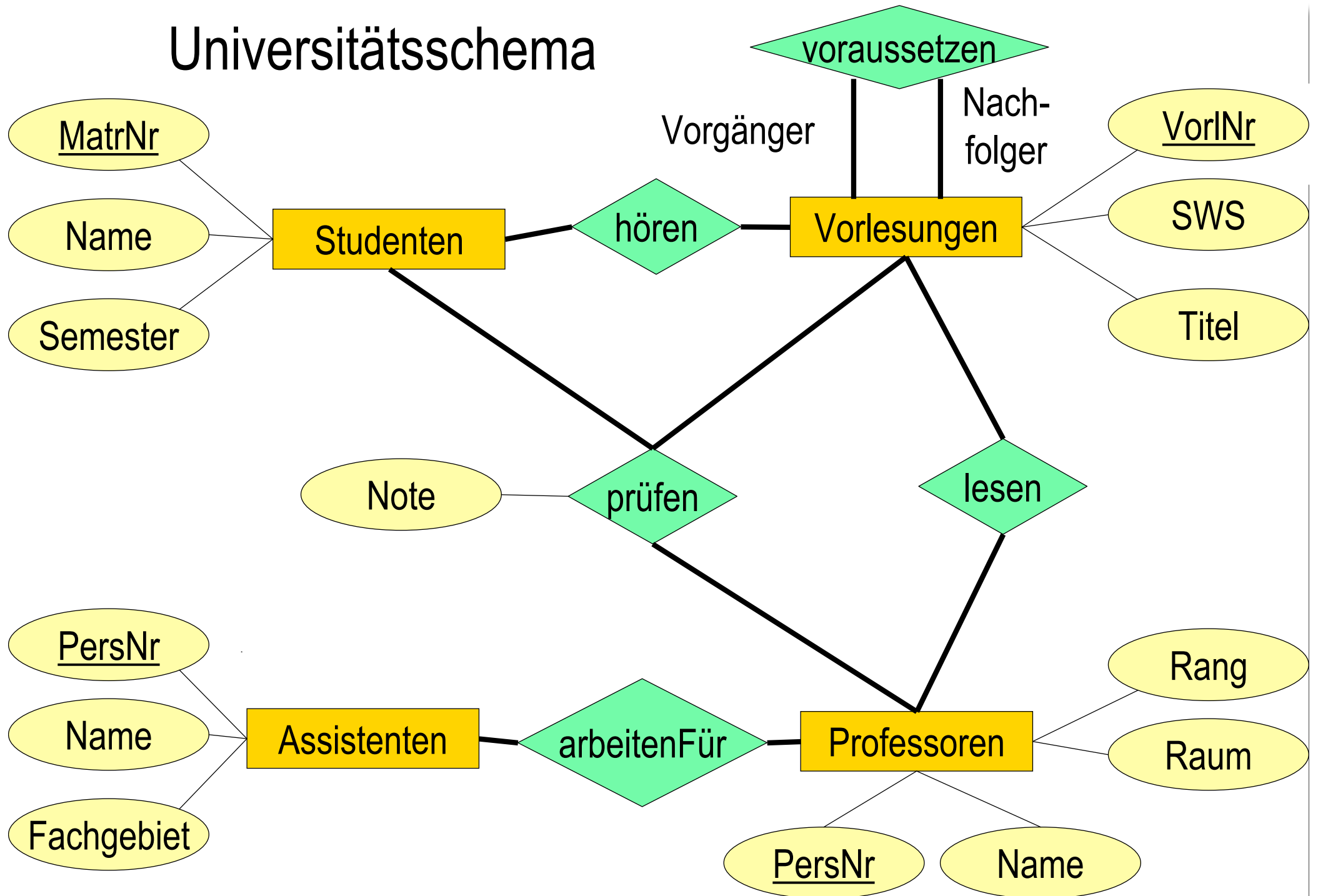
Referentielle Identifikation

	Projekt	Abteilung
←	●	...
←	●	...
←	●	...

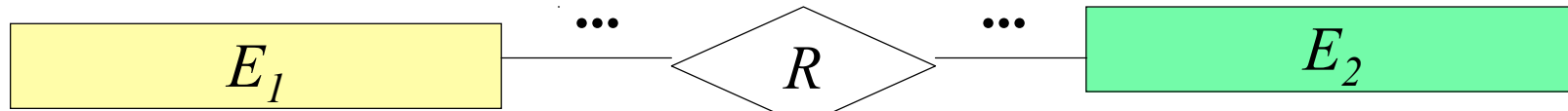
Assoziative Identifikation

Projekt	Abteilung
4711	...
4712	...
4713	...
...	...

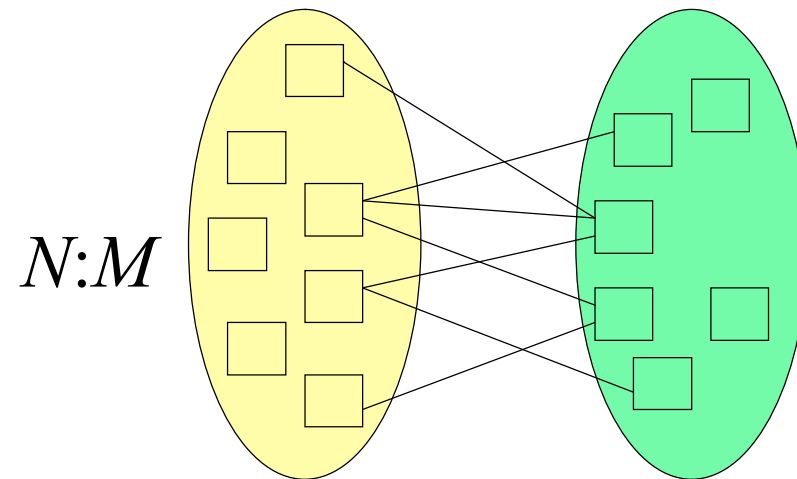
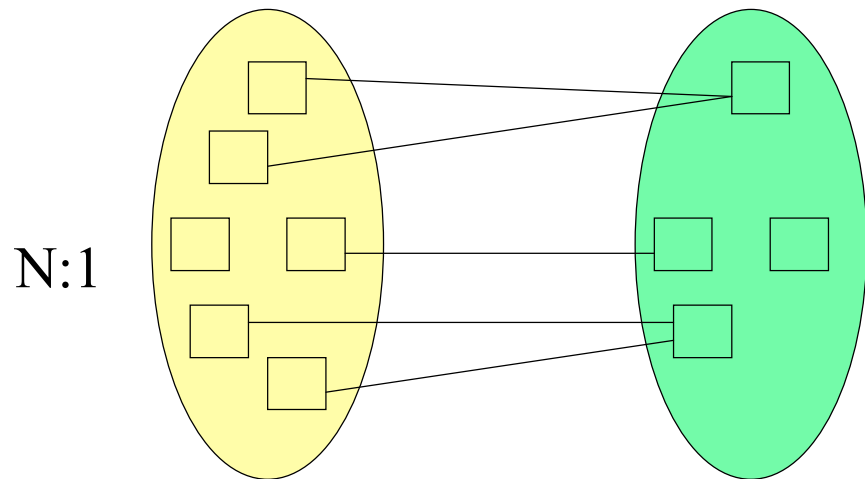
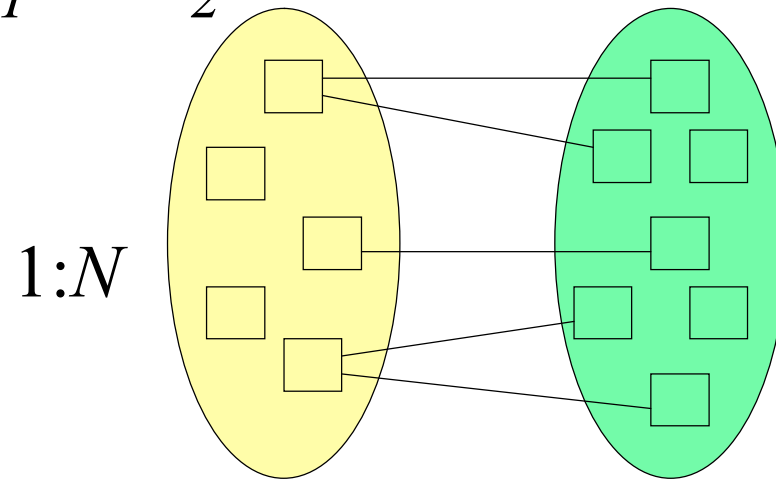
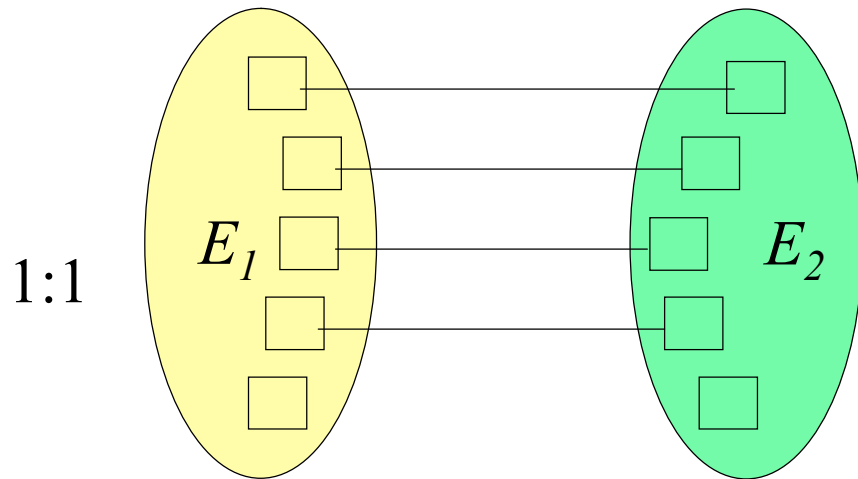
Universitätschema



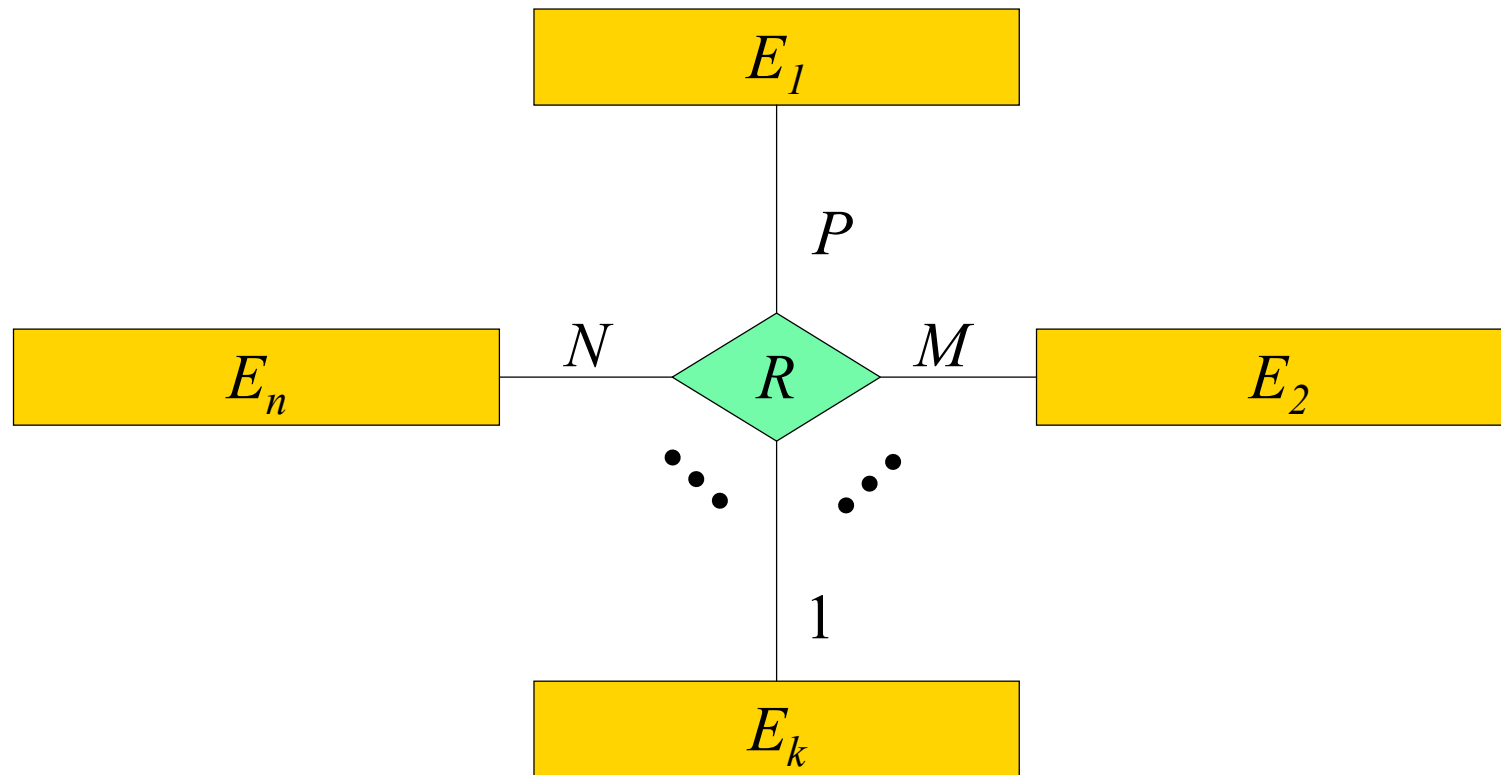
Funktionalitäten



$$R \subseteq E_1 \times E_2$$

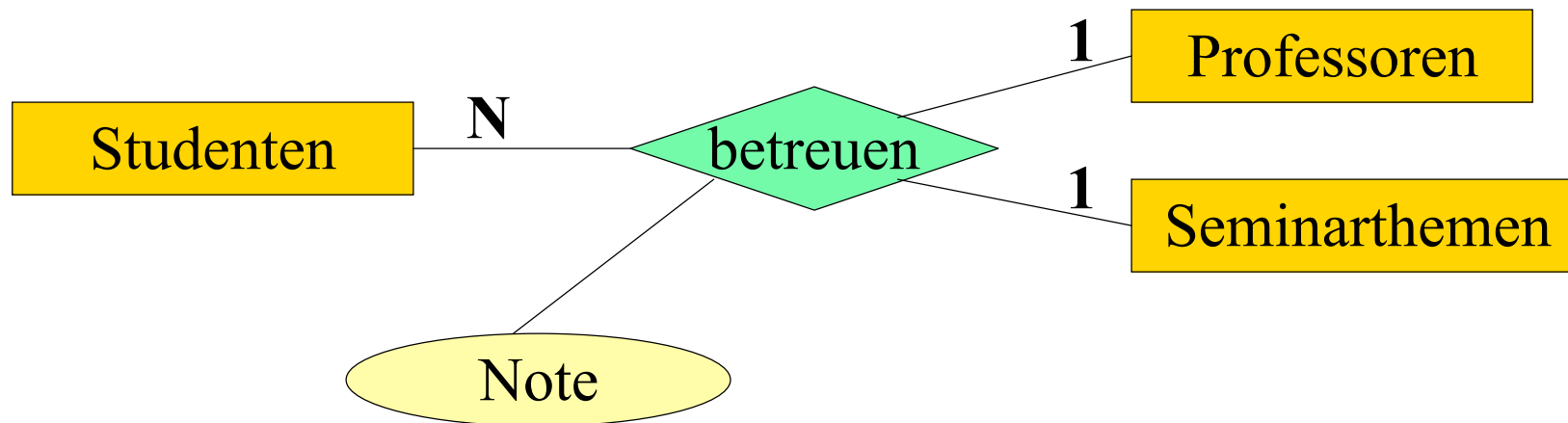


Funktionalitäten bei n -stelligen Beziehungen



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

Beispiel-Beziehung: *betreuen*



betreuen : Professoren \times Studenten \rightarrow Seminarthemen

betreuen : Seminarthemen \times Studenten \rightarrow Professoren

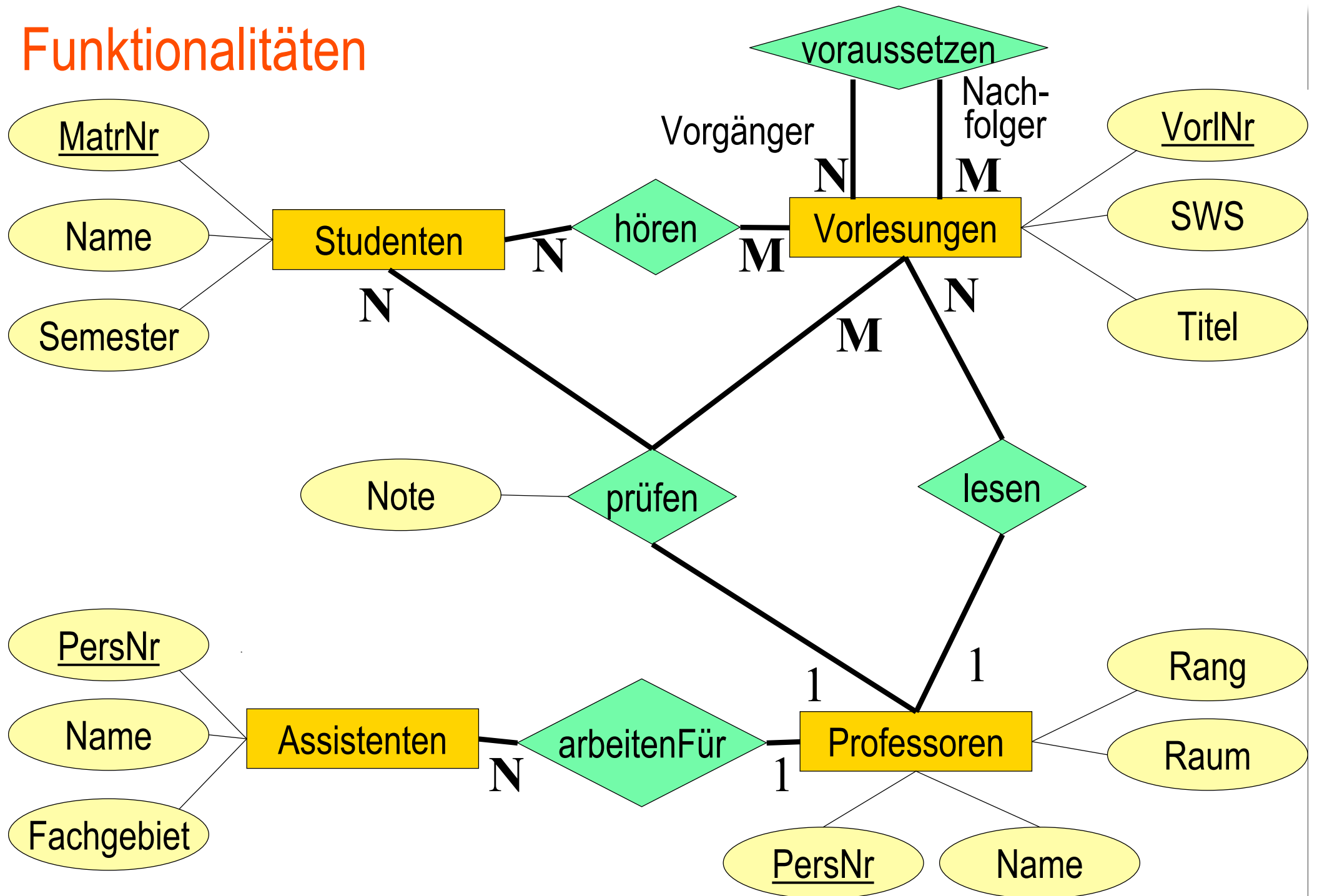
Dadurch erzwungene Konsistenzbedingungen

- ☐ Studenten dürfen bei demselben Professor bzw. derselben Professorin nur ein Seminarthema "ableisten" (damit ein breites Spektrum abgedeckt wird).
- ☐ Studenten dürfen dasselbe Seminarthema nur einmal bearbeiten – sie dürfen also nicht bei anderen Professoren ein schon einmal erteiltes Seminarthema nochmals bearbeiten.

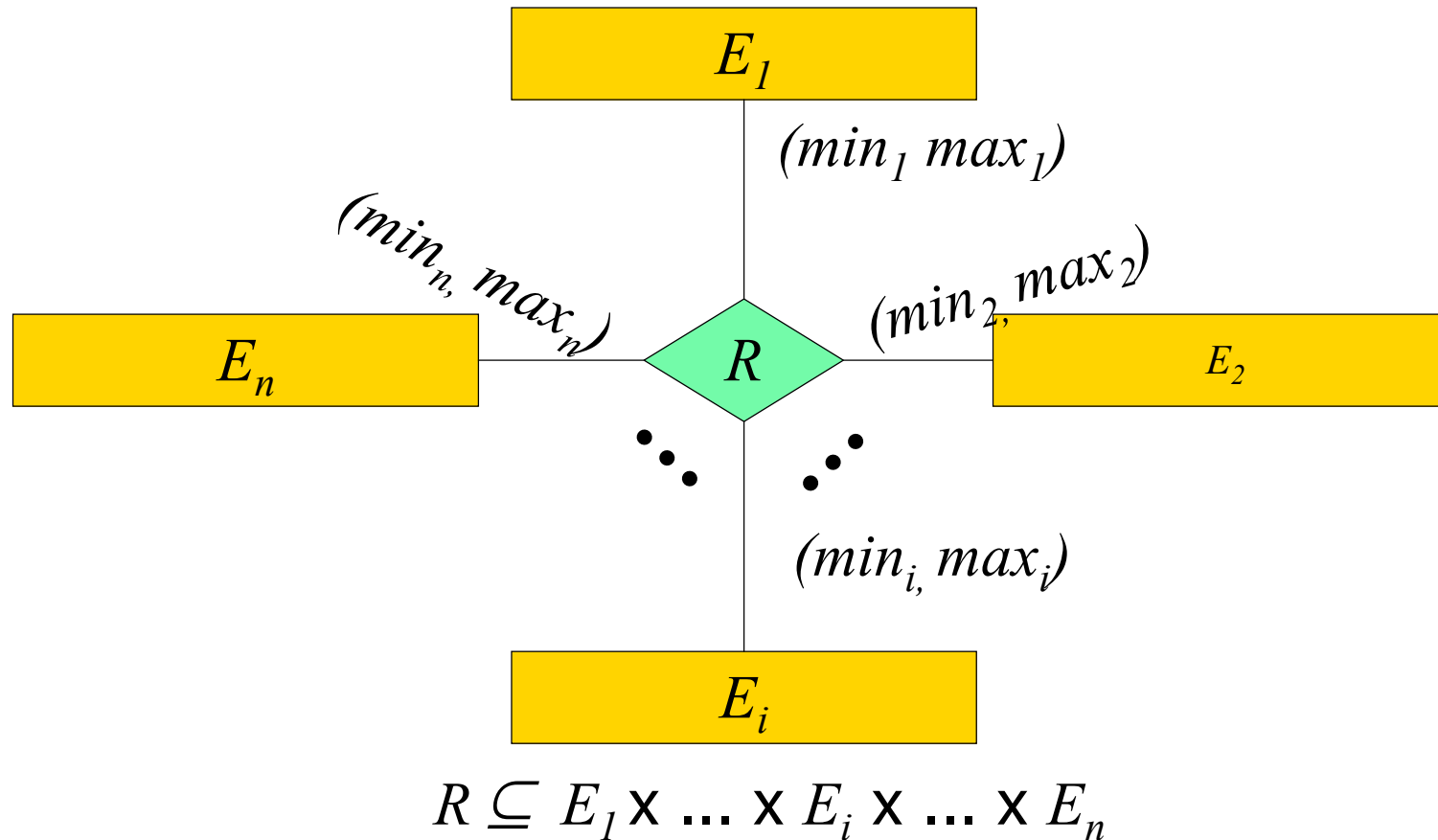
Folgende Datenbankzustände nach wie vor möglich:

- ▮ Professoren können dasselbe Seminarthema „wiederverwenden“ – also dasselbe Thema auch mehreren Studenten erteilen.
- ▮ Ein Thema kann von mehreren Professoren vergeben werden – aber an unterschiedliche Studenten.

Funktionalitäten



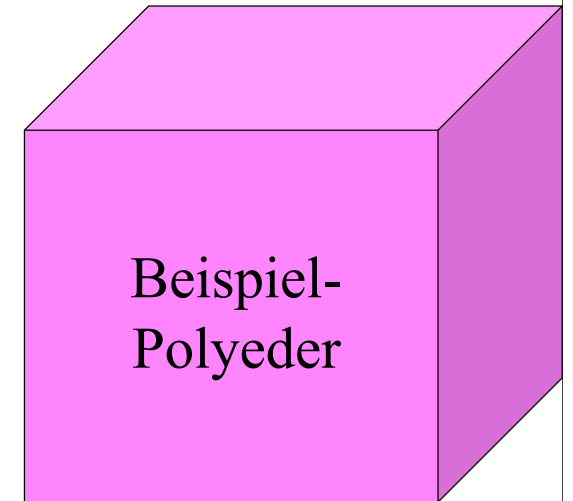
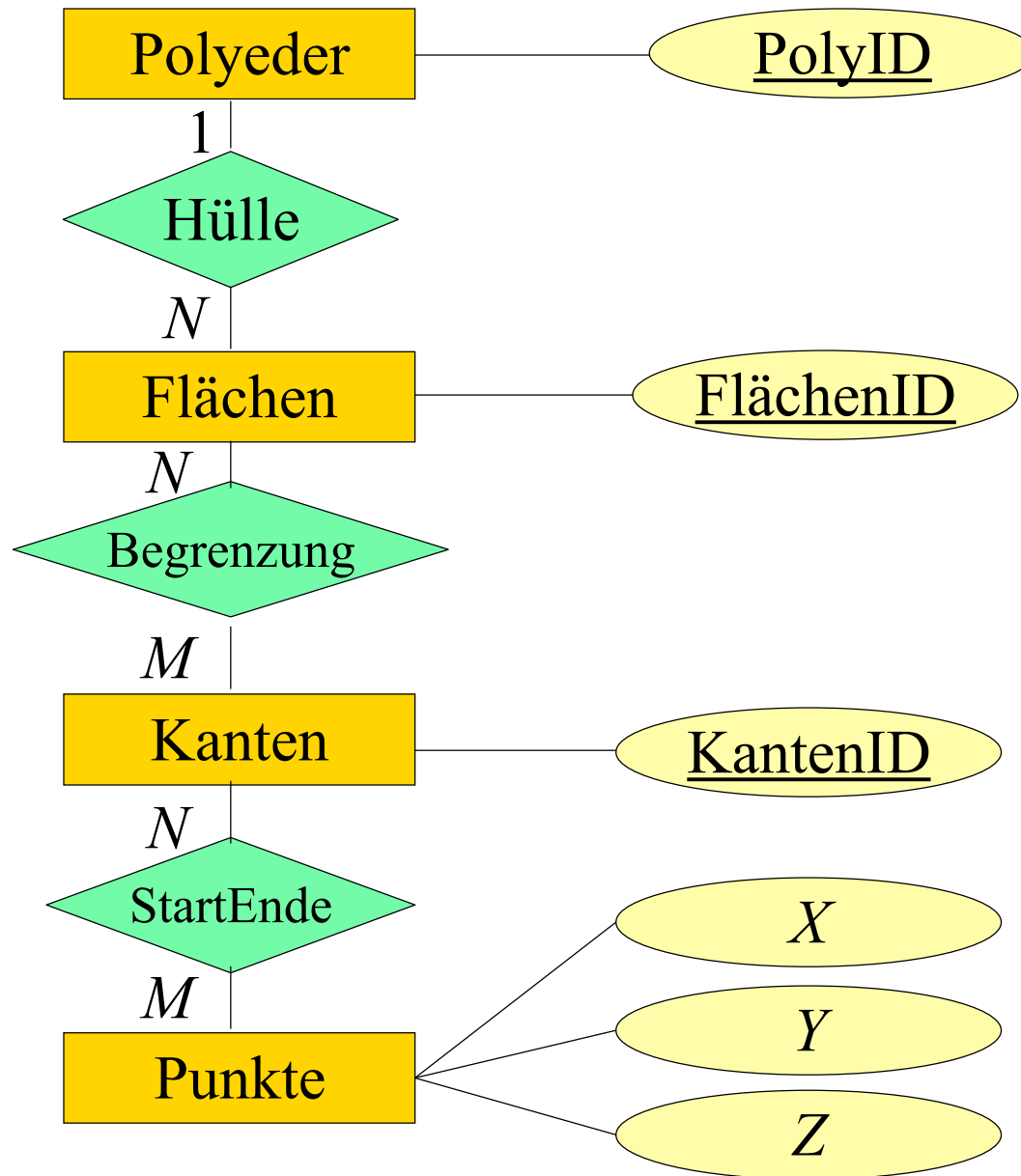
(Min, Max)-Notation



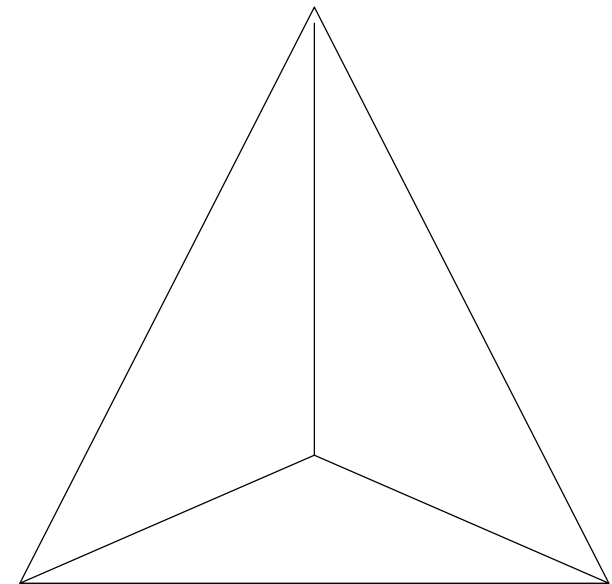
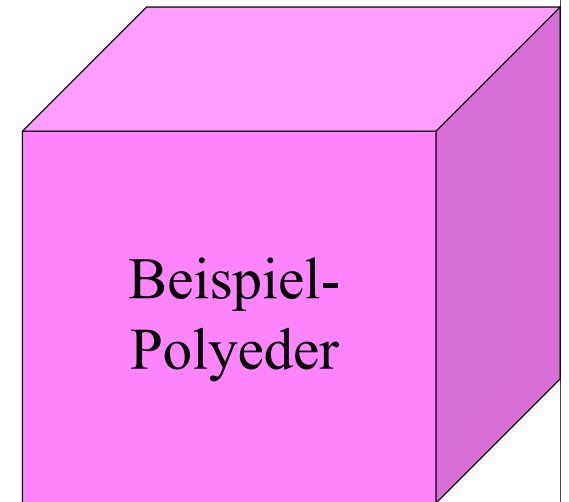
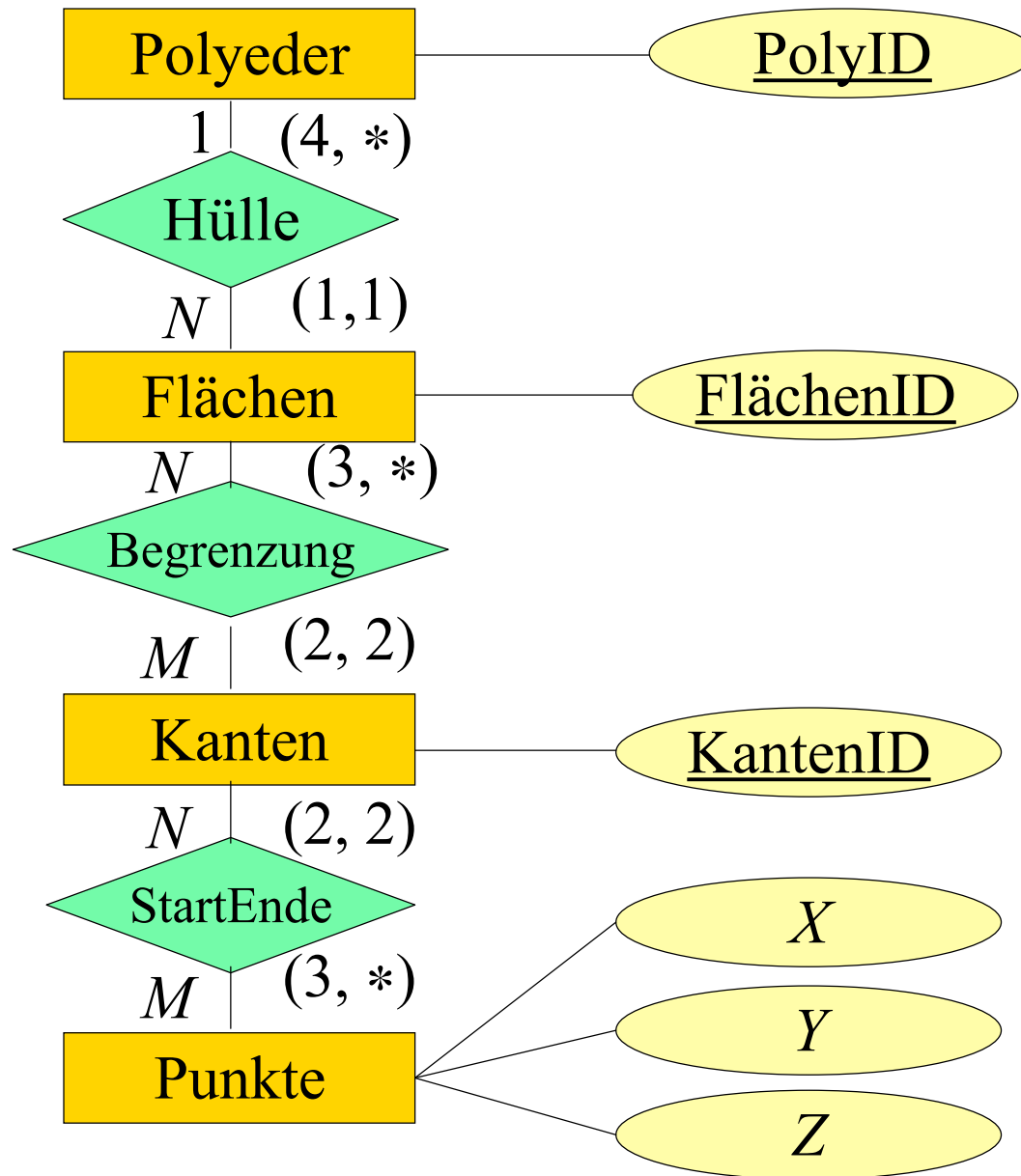
Für jedes $e_i \in E_i$ gibt es

- Mindestens min_i Tupel der Art (\dots, e_i, \dots) und
- Höchstens max_i viele Tupel der Art $(\dots, e_i, \dots) \in R$

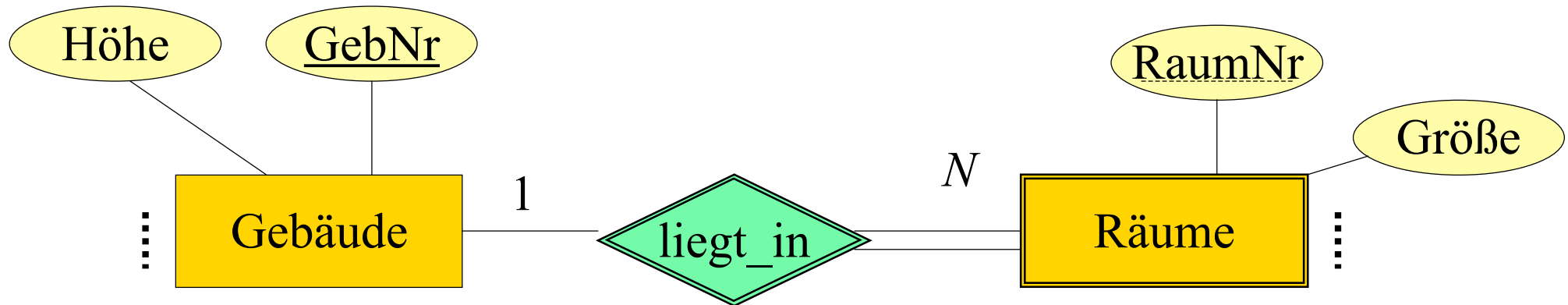
Komplex-strukturierte Entities



Komplex-strukturierte Entities

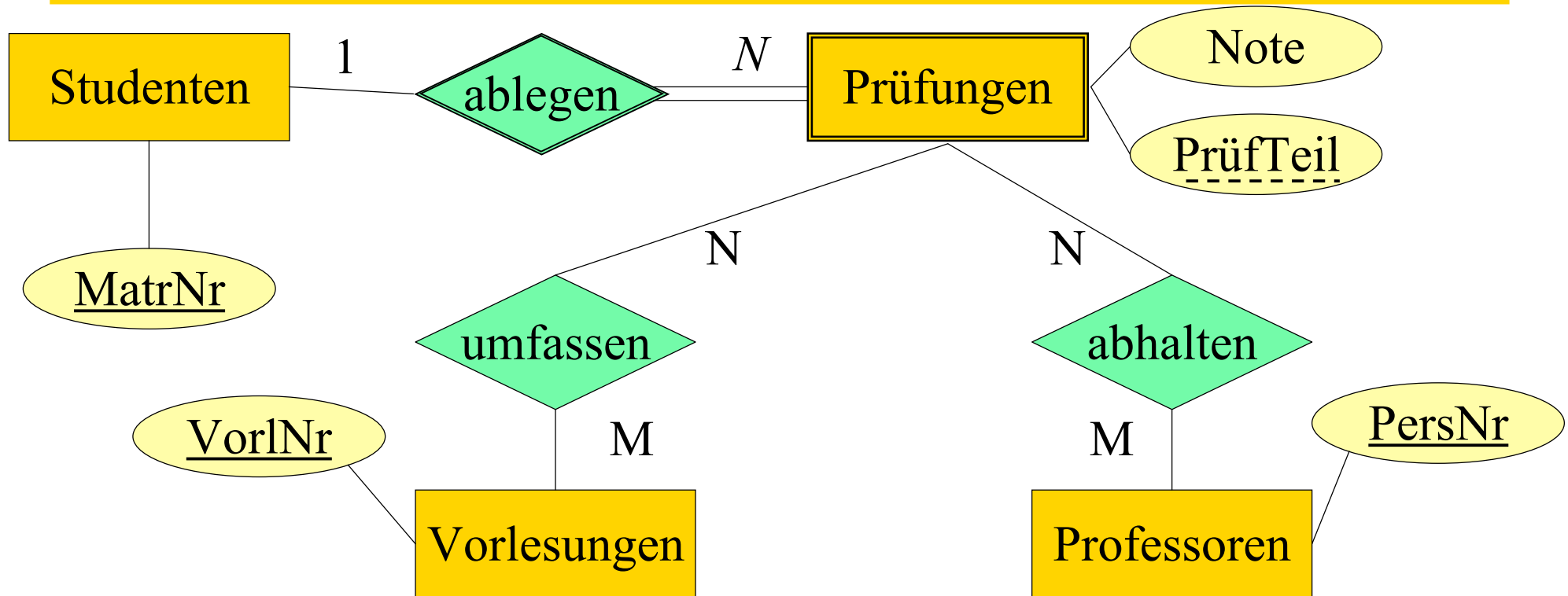


Schwache, existenzabhängige Entities



- Beziehung zwischen "starken" und schwachem Typ ist immer 1:N (oder 1:1 in seltenen Fällen)
- Warum kann das keine $N:M$ -Beziehung sein?
- RaumNr ist nur innerhalb eines Gebäudes eindeutig
- Schlüssel ist: GebNr **und** RaumNr

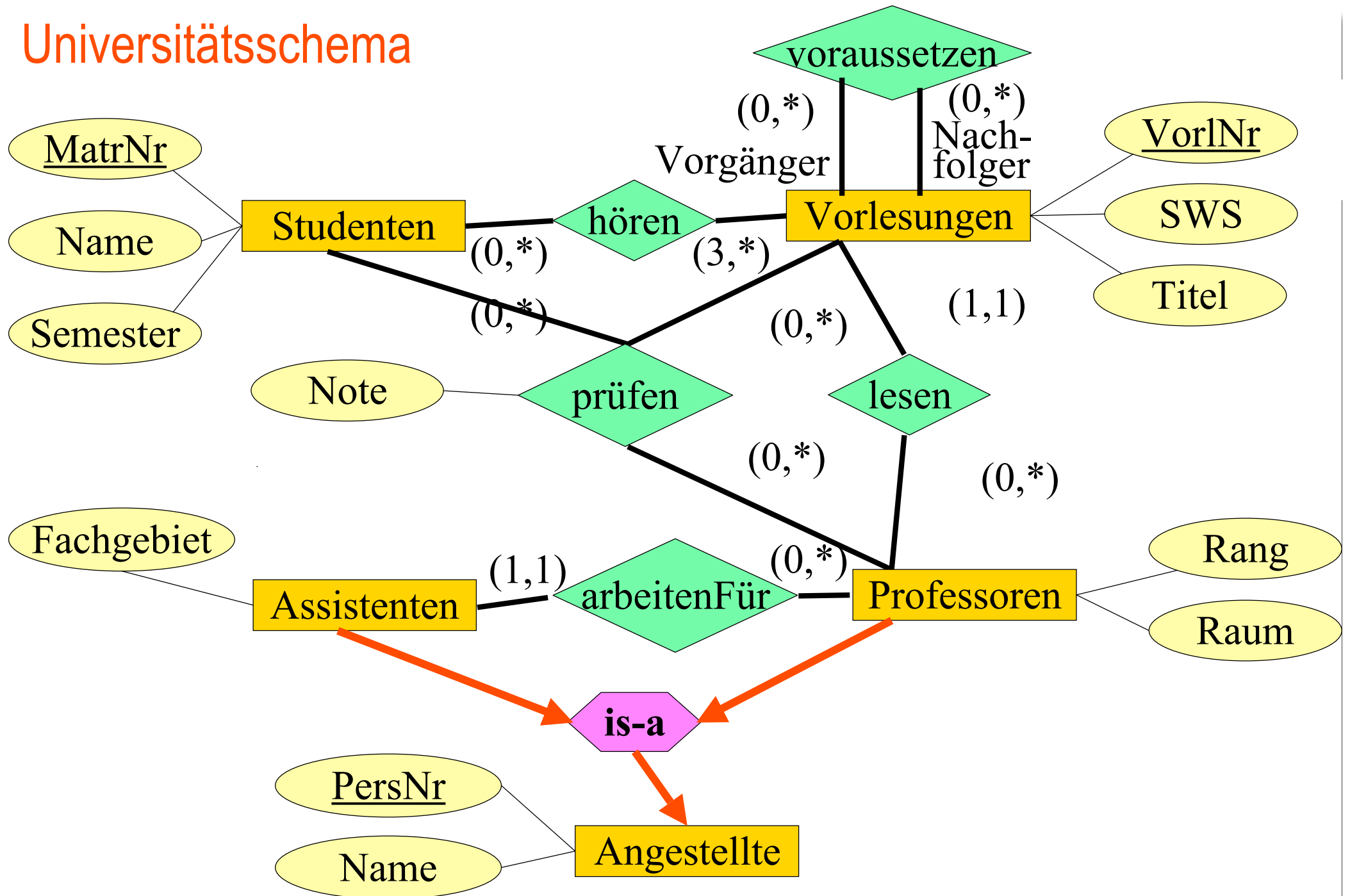
Prüfungen als schwacher Entitytyp



Mehrere Prüfer in einer Prüfung

Mehrere Vorlesungen werden in einer Prüfung abgefragt

Universitätsschema



Zusammenfassung, Kernpunkte



- Grundlagen von Datenbanksystemen
 - Architektur eines Datenbanksystems
 - Entwurfsebene: Entity-Relationship-Modell

Was kommt beim nächsten Mal?



- Implementierungsebene:
Das relationale Datenmodell