Databases

H2 conceptueel model

om te babbelen

HO
GENT

H2. Conceptueel model

1. Fasen in databankontwerp
2. Entity Relationship Diagram
3. Oefeningen
4. Bronnen

HO
GENT

1. Fasen in databankontwerp

HO GENT

Fasen in databankontwerp

Het ontwerpen van een databank vertrekt vanuit de bedrijfsprocessen en bestaat uit **4 fases**:

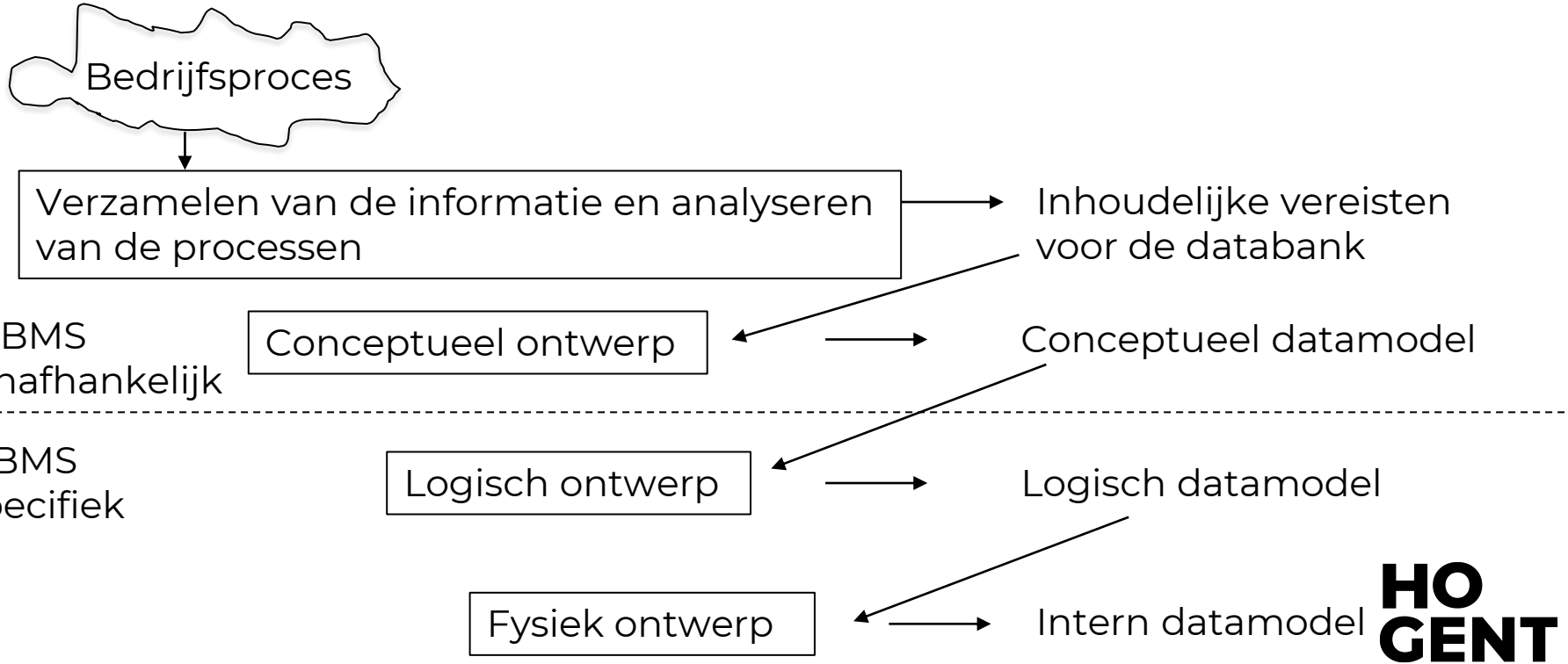
- Fase 1 = Verzamelen en analyseren van de functionele / inhoudelijke vereisten lees opdracht, versta je de info
- Fase 2 = Conceptueel ontwerp opstellen
- Fase 3 = Logisch ontwerp
- Fase 4 = Fysiek ontwerp

Mogelijke bedrijfsprocessen:

- het maken van facturen
- werkroosters en prestaties van werknemers
- voorraadbeheer
- puntenadministratie
- …

HO GENT

Fasen in databankontwerp



HO GENT

Fase 1 = Verzamelen en analyseren informatie

- Doel: de stappen en de benodigde data van het bedrijfsproces begrijpen.
  Wat nemen we op in de databank?

- Dit kan via
  - interviews met de opdrachtgever
  - analyse van bestaande formulieren en rapporten

- Vragen die moeten beantwoord worden
  - Welke data moet in de databank worden opgenomen?
  - Wat is de betekenis en context van alle data, symbolen, gebruikte coderingen?
  - Hoe zal de data worden verwerkt?
  - Wat is de beoogde functionaliteit?
  - Waarvoor zal de data gebruikt worden?

HO GENT

Fase 1 = Verzamelen en analyseren informatie

Voorbeeld: we willen info opslaan over films en acteurs.

niets verzinnen toevoegen dat niet wordt gemeld

- Relevante data is de titel van een film, het jaar waarin de film werd getoond, de rating-score, de namen van de acteurs …
- Er moeten films kunnen toegevoegd worden, de rating van een film moet kunnen aangepast worden …
- Het moet mogelijk zijn om een overzicht te krijgen van de films. Het moet bijvoorbeeld mogelijk zijn om te weten in welke verschillende films één specifieke acteur meespeelde, wat de gemiddelde rating is van romantische komedies, hoeveel thrillers er in 2015 werden geproduceerd …

HO GENT

Fase 2 = Conceptueel ontwerp

- Het conceptueel model is
  - een abstractie van de data en de onderlinge verbanden
  - formeel en ondubbelzinnig voor de databankontwerper.
  - gebruiksvriendelijk
  - doorgaans een grafische representatie
  - de basis voor communicatie en discussie tussen de gebruiker van het bedrijfsproces en de databankontwerper.
  - onafhankelijk van een databankmodel of een bestaande applicatie. Anders te vroeg gekoppeld aan dat bepaald databankmodel of die bepaalde applicatie.

HO GENT

Fase 2 = Conceptueel ontwerp

Voorbeeld: we willen info opslaan over films en acteurs.

- De data zal worden georganiseerd rond de centrale concepten 'Film' en 'Acteur'

- Gegevens die je wil opslaan
  - Film: titel, jaar waarin de film verscheen, genre, rating, aantal stemmen, adult-only, ….
  - Acteur: voornaam, familienaam, geboortejaar, jaar van overlijden

- Voorbeeld van informatie die niet in een conceptueel model opgenomen is:
  - Het geboortejaar van een acteur moet kleiner zijn dan het jaar van overlijden;
  - Van elke acteur moet het geboortejaar bekend zijn.
  - De rating van een film kan nooit kleiner zijn dan 0.

HO GENT

Fase 3 = Logisch ontwerp

- Type databank is bekend (relationele databank, NoSQL databank, hiërarchische databank, …)

- Het product zelf ligt nog niet vast
  - voor relationele databank Microsoft SQL Server of MySQL of DB2 of …
  - voor NoSQL document databank MongoDB of CouchDB of …
  - voor hiërarschische databank IMS of …

- !! Bij het opstellen van het conceptueel model en bij de overgang van het conceptueel model naar het logisch model is er mogelijk verlies van specificaties.
  → In een apart document bijhouden om te gebruiken bij de applicatie-ontwikkeling.

HO GENT

Fase 3 = Logisch ontwerp

Voorbeeld: we willen info opslaan over films en acteurs.

- We kiezen voor het relationeel databasemodel.
- De centrale concepten ‘Film’ en 'Acteur’ worden omgezet en zullen later evolueren naar tabellen.

HO GENT

Fase 4 = Fysiek ontwerp

- Is de feitelijke implementatie van het logisch model.
- Je kiest eerst een product, ook DBMS genoemd (MySQL, Microsoft SQL Server, Oracle, …).
- Je implementeert het logisch model en zet dit om in datadefinitiecode (= DDL), die kan worden verwerkt door het DBMS.
- Technische details worden toegevoegd (datatypes van de attribuuttypes, …)
- Indien mogelijk worden ook de functionele beschrijvingen ‘vertaald’ naar databaseconcepten. Zo kunnen de bedrijfsregels rond correct geboortejaar en jaar van overlijden omgezet worden naar een integriteitsrestrictie.
- DBA kan ook aanbevelingen doen in verband met de performantie.
  → zie Relational Databases and Datawarehousing (2TI)

HO GENT

Fases in Databank Ontwerp

overzicht van de hele cursus

!!!!goed kennen!!!!

Verzamelen en analyseren van de vereisten

- domeinanalyse
- functionele analyse
- behoefteanalyse

Conceptueel ontwerp

- conceptueel model (bijvoorbeeld EER – diagram)
- functionele beschrijving

databasemodel-onafhankelijk

Logisch ontwerp

- logisch databankschema (bijvoorbeeld relationeel)
- gedragsspecificaties

dbms-onafhankelijk

Fysiek ontwerp

- DDL-scripts
- implementatie van gedrag

HO GENT

Bron: Principes van Databases (G. De Tré)

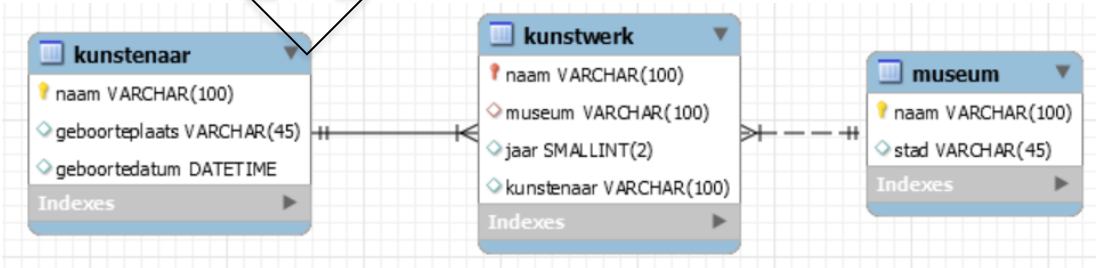
Fasen in Databank Ontwerp.

Conceptueel model

We vertalen het conceptueel model naar een relationele databank: Welke tabellen zijn er nodig? Welke kolommen? Naast relationele databanken, bestaan er ook nog andere.

Logisch model

We voorzien scripts om de databank fysisch te creëren.



Fysisch model

```
CREATE TABLE `kunstenaar` (
  `naam` varchar(100) NOT NULL,
  `geboorteplaats` varchar(45) DEFAULT NULL,
  `geboortedatum` datetime DEFAULT NULL,
  PRIMARY KEY (`naam`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8
```

2. Entity Relationship Diagram

een concept/ding
OOP: class

relaties, onderlinge verbanden
vb student in klas

tekening

HO
GENT

Inleiding

- Het Entity Relationship Diagram (ERD) werd geïntroduceerd en geformaliseerd door Peter Chen in 1976.

- Het is één van de populairste voorstellingswijzen voor het conceptueel gegevensmodel.

- Een Entity Relationship Diagram heeft de volgende bouwstenen:
  - Entiteittypes
  - Attribuuttypes
  - Relatietypes

HO GENT

Entiteittype

= beschrijving

onderscheidbaar

- Een **entiteittype**
  - bestaat in de reële wereld.
  - kan zowel abstract (een tentoonstelling, firma, cursus, job, …) als fysiek (schilderij, persoon, auto, huis, …) zijn.
  - is ondubbelzinnig gedefinieerd voor een bepaalde groep gebruikers.
  - karakteriseert een collectie van entiteiten
  - heeft een naam en inhoud en is identificeerbaar.
- Een entiteit is een instantie van een entiteittype.
- In het conceptueel model nemen we entiteittypes op (geen individuele entiteiten).

entiteit is instantie van entiteittype

HO GENT

Attribuuttype

vb AT = kleur
A = blauw

- Een attribuuttype
  - is een karakteristiek van een entiteittype
  - beschrijft het entiteittype

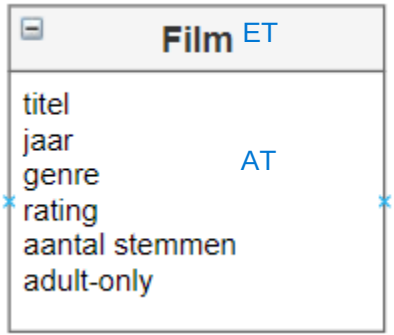
- Elke entiteit heeft een specifieke waarde voor elke attribuuttype.

HO GENT

Entiteittype en attribuuttype

Voorbeeld: visualisatie van de entiteittypes voor Film en Acteur.

- Over de ondergang van de Titanic werd in 1997 een romantische film uitgebracht die op imdb een rating van 7,9/10 haalt. Deze rating is gebaseerd op 1,2M stemmen.
- Leonardo DiCaprio (°1974) is een entiteit van het entiteittype Acteur.



de data erin zijn de atributen

HO GENT

Entiteittype en attribuuttype

- Voor een onervaren databaseontwerper kan het onduidelijk zijn of een gegeven concept al dan niet als entiteittype moet worden gemodelleerd.

- Een entiteittype is **identificeerbaar** en moet **een inhoud** hebben.

HO GENT

Attribuuttype

Het ER-model kent een aantal mogelijkheden om attribuuttypes verder te karakteriseren:

- Enkelvoudige versus samengestelde attribuuttypes
  AT kunnen nog verder worden verdeeld
  vb. student met adres
- Enkelwaardige versus meerwaardige attribuuttypes
  er bestaan meerdere dingen
  vb. student met hobbies
- Afgeleide attribuuttypes
  worden berekenen
  vb. leeftijd: kan worden berekend uit geboortejaar
- Kandidaatsleutelattribuuttypes
  sleutel is unieke indentificatie
  vb. studentenummer

onderstrepen

/
of (u)
\

Attribuuttype

Enkelvoudige versus samengestelde attribuuttypes

- Samengesteld attribuuttype: het attribuuttype kan nog opgesplitst worden. Bijvoorbeeld het attribuuttype ‘adres’ kan samengesteld zijn uit een ‘straat’, een ‘nummer’, een ‘postcode’ en een 'woonplaats'.
  Wij werken in het conceptueel model steeds op het niveau van enkelvoudige attribuuttypes.
- Afhankelijk van de context zullen attribuuttypes soms verder opgesplitst worden of niet. Bijvoorbeeld als het niet belangrijk is dat 'straat' of 'woonplaats' afzonderlijk moet gekend zijn, dan wordt 'adres' een enkelvoudig attribuuttype. In dat geval kan niet met de afzonderlijke delen (straat, stad, …) gewerkt worden.

HO GENT

Attribuuttype

Enkelwaardige versus meerwaardige attribuuttypes

- Enkelwaardig attribuuttype: het attribuuttype heeft één waarde. Bijvoorbeeld het attribuuttype ‘titel’ van ‘Film’ en de attribuuttypes ‘geboortejaar’ en ‘jaar overlijden’ van ‘Acteur’.
- Meerwaardig attribuuttype: het attribuuttype kan (meerdere) waarden bevatten. Bijvoorbeeld een 'Film' kan meerdere genres hebben. In dat geval is 'genre' een meerwaardig attribuuttype.
- In een ERD mogen beide voorkomen (zie later).
  Binnen deze cursus vermijden we meerwaardige attributen in het ERD.

HO
GENT

Attribuuttype

Afgeleide attribuuttypes

- De waarde van een afgeleid attribuuttype kan berekend worden op basis van waarden van andere attribuuttypes.
- Een typisch voorbeeld is ‘leeftijd’: de waarde kan berekend worden als het verschil van de huidige datum en de waarde van het attribuuttype ‘geboortedatum’
- Afgeleide attribuuttypes worden **niet opgeslagen** in de databank, omdat dit een potentiële bron van inconsistentie kan zijn.
  Dit wordt vervangen door de **basisinformatie** waaruit de waarde van het attribuuttype kan berekend worden.

HO GENT

Attribuuttype

Kandidaatsleutelattributen

- Één attribuut of meerdere attributen samen die de entiteiten van een entiteittype op een unieke, irreducibele manier identificeren, vormen een **kandidaatsleutel** van het entiteittype.
  Irreducibiliteit wil zeggen dat er geen uniciteit mag gelden als men een 1 of meerdere attributen weglaat.
- De attributen die deel uitmaken van een kandidaatsleutel noem je **kandidaatsleutelattributen**.
- Er kunnen meerdere **kandidaatsleutels** zijn. Later wordt uit de kandidaatsleutels één sleutel gekozen als primaire sleutel.

HO GENT

Attribuuttype

Kandidaatsleutelattributen

Voorbeeld: Stel er bestaan geen twee acteurs met eenzelfde combinatie van voornaam en familienaam => voornaam en familienaam zijn **kandidaatsleutelattributen** en vormen (samen) de kandidaatsleutel.

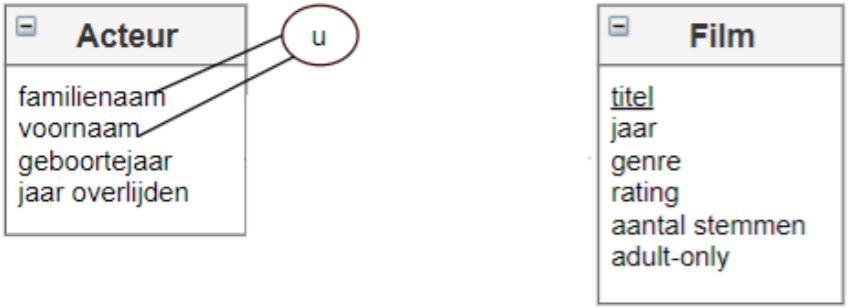
De combinatie van voornaam, familienaam en geboortedatum is geen kandidaatsleutel want de combinatie van voornaam en familienaam op zich is al uniek => voornaam, familienaam én geboortedatum zijn niet irreducibel, maw geboortedatum is overbodig om uniciteit af te dwingen.

HO GENT

Attributen

Kandidaatsleutelattributen

- Alle enkelvoudige kandidaatsleutels (bestaande uit 1 attribuuttype) worden onderlijnd.
- Indien een kandidaatsleutel uit meerdere attribuuttypes bestaat (samengestelde kandidaatsleutel), duiden we dit dan aan met de 'u'-constraint (= unique constraint).



HO GENT

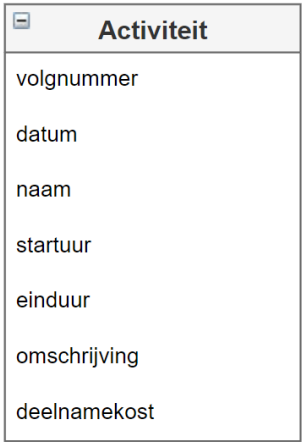
Oefening Activiteiten

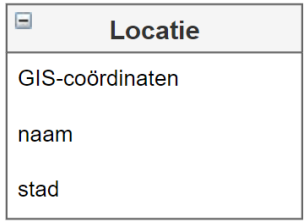
- Je wil een activiteitendatabank creëren
- Elke activiteit krijgt een uniek volgnummer, een datum, een naam, een startuur, vermoedelijk einduur, een korte omschrijving en een deelnamekost.
- Elke activiteit start en eindigt op een bepaalde locatie. Een locatie heeft unieke GIS-coördinaten, een naam en een stad.
- Welke entiteittypes en attribuuttypes herken je?

HO GENT

Oefening Activiteiten

- Je wil een activiteitendatabank creëren
- Elke activiteit krijgt een uniek volgnummer, een datum, een naam, een startuur, vermoedelijk einduur, een korte omschrijving en een deelnamekost.
- Elke activiteit start en eindigt op een bepaalde locatie. Een locatie heeft unieke GIS-coördinaten, een naam en een stad.
- Welke entiteittypes en attribuuttypes herken je?





HO GENT

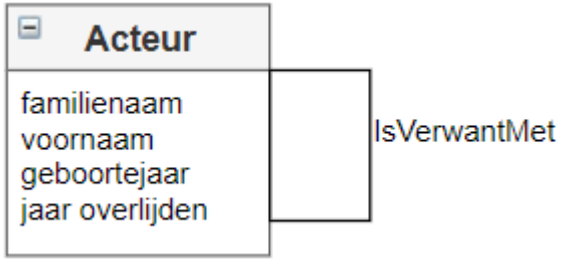
Relatietype

- Entiteittypes kunnen onderlinge verbanden hebben:
  - een acteur speelt mee in een film
  - een student volgt een aantal cursussen
- Er kunnen één, twee, drie of meer entiteittypes betrokken zijn in een relatie
  - één entiteittype: een acteur kan verwant zijn aan een andere acteur
  - twee enteittypes: in een film spelen één of meerdere acteurs
  - drie entiteittypes: een arts schrijft een medicijn voor aan een patiënt.
- Een **relatietype** is een verzameling van relaties tussen instanties van één, twee of meer al dan niet verschillende entiteittypes. Men spreekt respectievelijk van een unair (n = 1), binair (n = 2), ternair (n = 3) of n-air (n > 3) relatietype.
  Elk relatietype wordt gekenmerkt door een naam.
- In deze cursus beperken we ons tot unaire en binaire relatietypes.

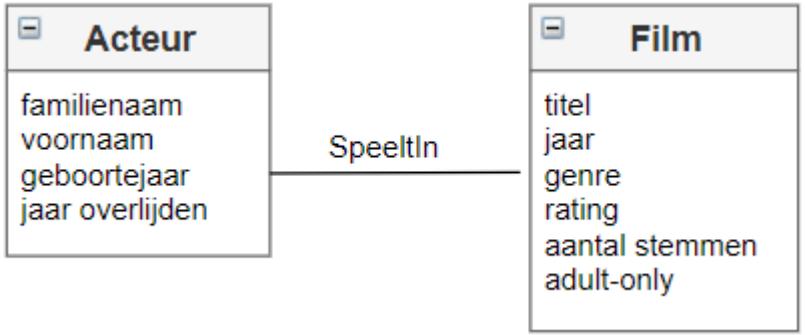
HO GENT

Relatietype

- De **graad** van een relatietype = het aantal verschillende entiteittypes die deelnemen aan het relatietype
- Voorbeeld van een unaire of recursieve relatie (Er is één entiteittype betrokken in de relatie, namelijk Acteur):

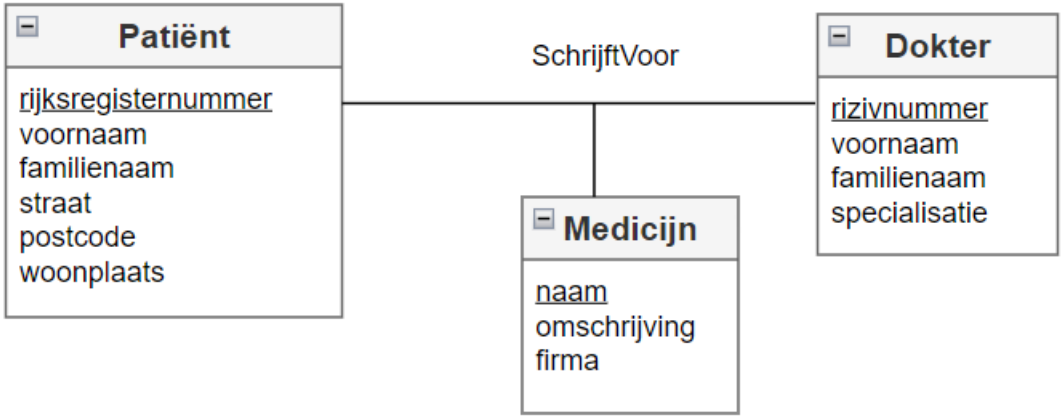


- Voorbeeld van een binaire relatie (Er zijn twee entiteittypes betrokken in de relatie, namelijk Acteur en Film):



Relatietype

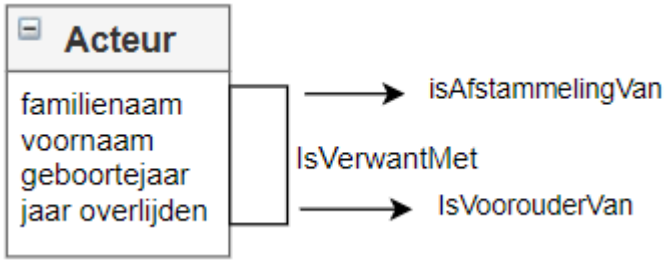
- Voorbeeld van een ternaire relatie (Er zijn 3 entiteittypes betrokken in de relatie, namelijk Patiënt, Dokter en Medicijn):



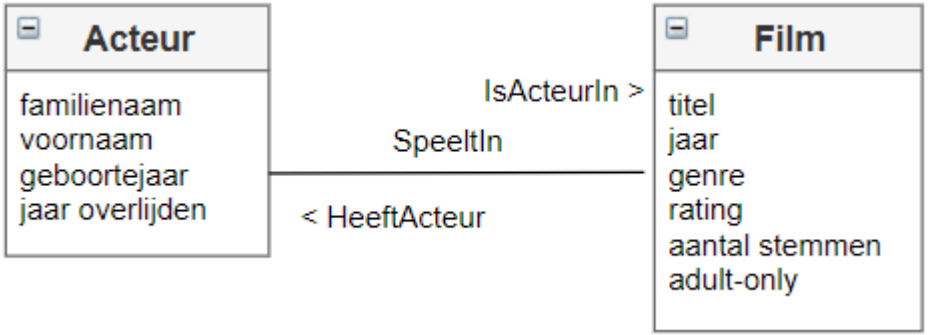
HO GENT

Relatietype

- De **rollen** van een relatietype beschrijven de specifieke rol van elk entiteittype in het relatietype.
- Voorbeeld van een unaire of recursieve relatie:



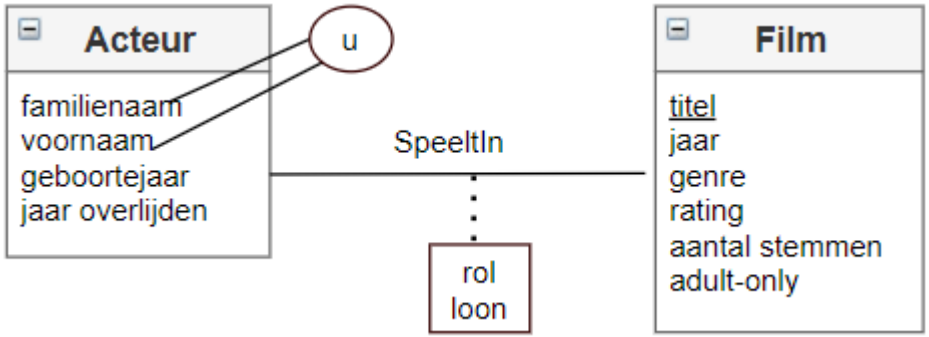
- Voorbeeld van een binaire relatie:



HO GENT

Relatie-attribuut

- Ook relatietypes kunnen eigenschappen hebben: wanneer een kenmerk een eigenschap is van het relatietype en niet van één van de betrokken entiteittypes. We spreken van een **relatie-attribuut**.
- **Voorbeeld**: de rol die een acteur vertolkte in een film. Deze rol hoort bij de 'SpeeltIn'-relatie tussen Acteur en Film en is geen eigenschap van Acteur (die verschillende rollen kan gespeeld hebben) of van Film (er zijn meerdere rollen in een film).
- Idem voor het loon dat een acteur gekregen heeft voor het meespelen in een film.



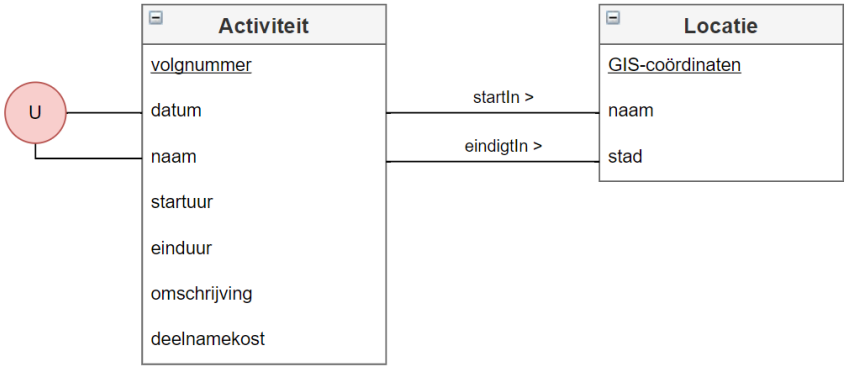
Oefening Activiteiten

- Je wil een activiteitendatabank creëren
- Elke activiteit krijgt een uniek volgnummer, een datum, een naam, een startuur, vermoedelijk einduur, een korte omschrijving en een kostprijs om deel te nemen. De combinatie van datum en naam is ook uniek.
- Elke activiteit start en eindigt op een bepaalde locatie.
  Een locatie heeft unieke GIS-coördinaten,
  een naam en een stad.
- Geef de kandidaatsleutels

HO GENT

Oefening Activiteiten

- Je wil een activiteitendatabank creëren
- Elke activiteit krijgt een uniek volgnummer, een datum, een naam, een startuur, vermoedelijk einduur, een korte omschrijving en een kostprijs om deel te nemen. De combinatie van datum en naam is ook uniek.
- Elke activiteit start en eindigt op een bepaalde locatie.
  Een locatie heeft unieke GIS-coördinaten,
  een naam en een stad.
- Geef de relatietype(s)



HO GENT

Cardinaliteiten

- Elk relatietype heeft een minimum- en een maximumcardinaliteit.
- Cardinaliteit betekent aantal en wordt uitgedrukt als een getal.
- De cardinaliteiten moeten afgetoetst worden met de opdrachtgever! Deze zijn vaak afhankelijk van de bedrijfsregels.
  Modelleer enkel wat je weet. We veronderstellen niets!

Verkeerde cardinaliteiten kunnen leiden tot minder kwalitatieve applicaties.

HO GENT

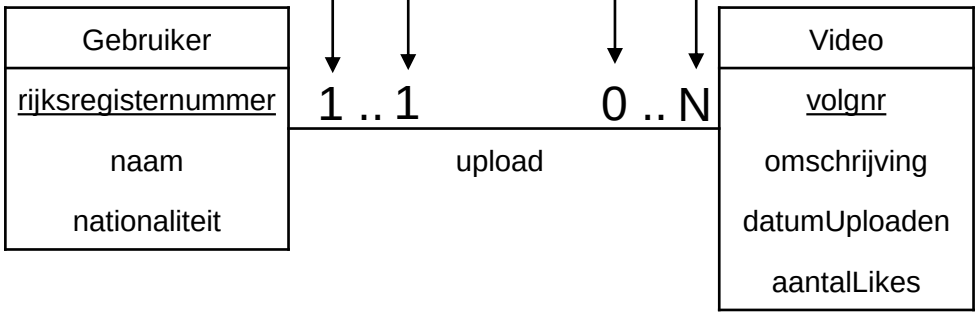
Cardinaliteiten

- Maximumcardinaliteit = het maximum aantal entiteiten van het entiteittype dat op een gegeven tijdstip **kan** deelnemen aan een relatie van het relatietype. Mogelijke waarden zijn 1 of N.
  - 1: één entiteit kan in relatie staan met maximum 1 (andere) entiteit via dit relatietype
  - N: één entiteit kan in relatie staan met N (andere) entiteiten via dit relatietype. N is een willekeurig geheel getal groter dan 1.
- Minimumcardinaliteit = het minimum aantal entiteiten van het entiteittype dat op elk tijdstip **moet** voorkomen in een relatie van het relatietype. Mogelijke waarden zijn 0 of 1.
  - 0: sommige entiteiten nemen niet deel aan de relatie. De relatie is optioneel voor dat entiteittype.
  - 1: een entiteit moet altijd in relatie staan met minimum één andere entiteit

HO GENT

Cardinaliteiten

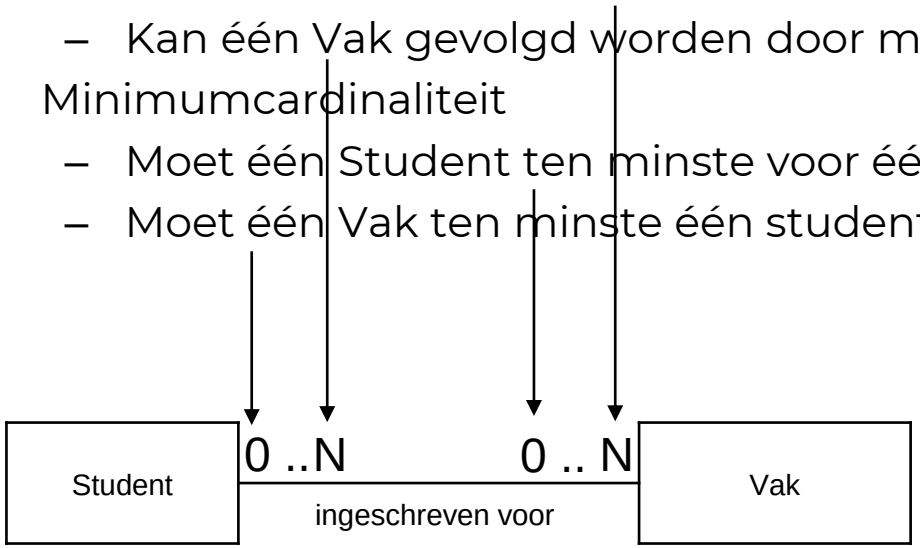
- Maximumcardinaliteit
  - Kan één Gebruiker meer dan één Video geüpload hebben? Ja => N
  - Kan één Video geüpload zijn door meer dan één Gebruiker? Neen => 1
- Minimumcardinaliteit
  - Moet één Gebruiker ten minste één Video geüpload hebben? Neen => 0
  - Moet één Video geüpload zijn door ten minste één Gebruiker? Ja => 1



HO GENT

Cardinaliteiten

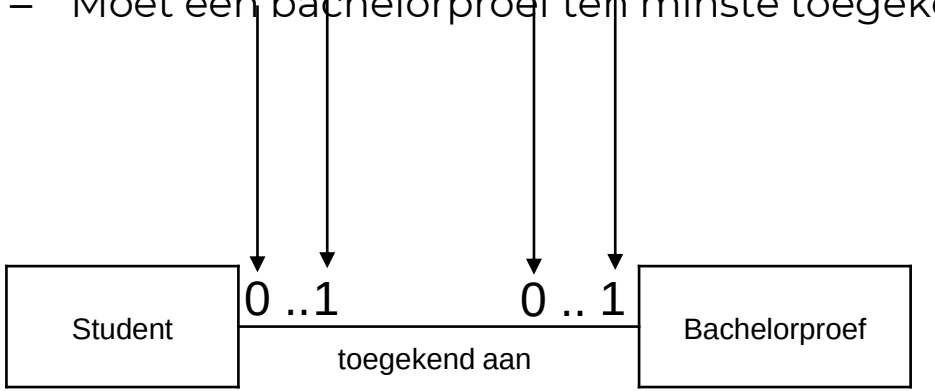
- Maximumcardinaliteit
  - Kan één Student ingeschreven zijn voor meer dan één Vak ? Ja => N
  - Kan één Vak gevolgd worden door meer dan één Student? Ja => N
- Minimumcardinaliteit
  - Moet één Student ten minste voor één Vak ingeschreven zijn? Neen => 0
  - Moet één Vak ten minste één student hebben? Neen => 0



HO GENT

Cardinaliteiten

- Maximumcardinaliteit
  - Kan één Student toegekend zijn aan meer dan één bachelorproef? Neen => 1
  - Kan één bachelorproef toegekend zijn aan meer dan één Student? Neen => 1
- Minimumcardinaliteit
  - Moet één Student toegekend zijn aan ten minste één bachelorproef? Neen => 0
  - Moet één bachelorproef ten minste toegekend zijn aan één Student? Neen => 0

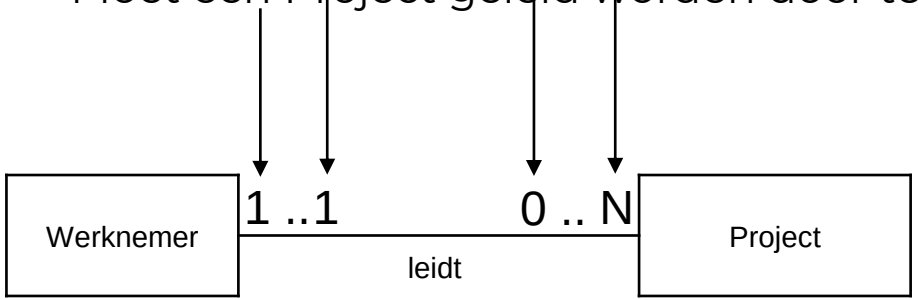


HO GENT

Cardinaliteiten

Voorbeeld van bedrijfsspecifieke cardinaliteiten.

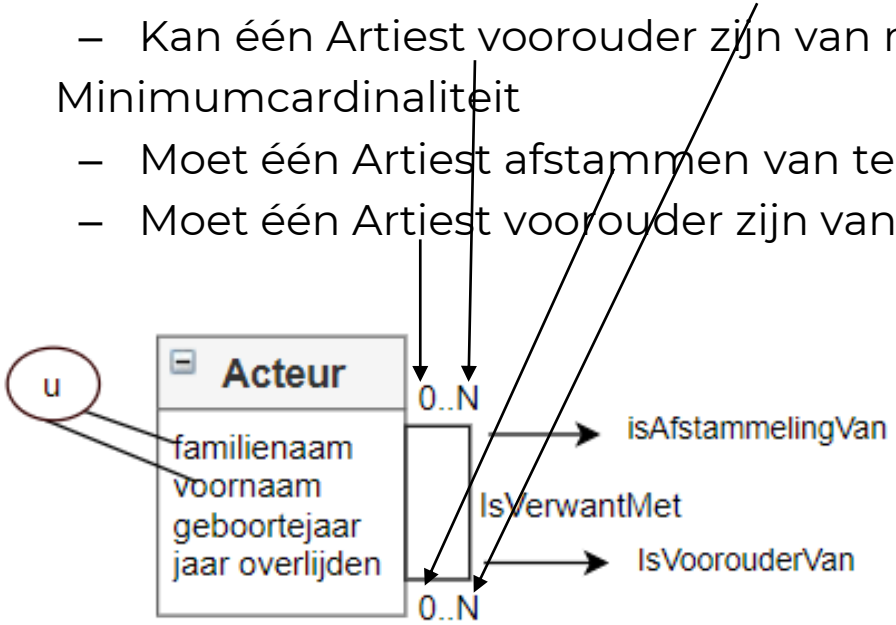
- Maximumcardinaliteit
  - Kan één Werknemer meer dan één Project leiden? Ja => N
  - Kan één Project geleid worden door meer dan één Werknemer? Neen => 1
- Minimumcardinaliteit
  - Moet één Werknemer ten minste één Project leiden? Neen => 0
  - Moet één Project geleid worden door ten minste één Werknemer? Ja => 1



HO GENT

Cardinaliteiten

- Maximumcardinaliteit
  - Kan één Artiest afstammen van meer dan één Artiest? Ja => N
  - Kan één Artiest voorouder zijn van meer dan één Artiest? Ja => N
- Minimumcardinaliteit
  - Moet één Artiest afstammen van ten minste één Artiest? Neen => 0
  - Moet één Artiest voorouder zijn van ten minste één Artiest? Neen => 0



HO GENT

Cardinaliteiten

- Relatietypes worden vaak gekarakteriseerd door de maximumcardinaliteit van elk van de rollen.

- In het geval van een unaire of binaire relatie geeft dit aanleiding tot:
  - 1-op-1 relatie → 1:1
  - 1-op-veel-relatie → 1:N of N:1
  - veel-op-veel-relatie → M:N of N:N

- Een verplichte minimumcardinaliteit wijst op bestaansafhankelijkheid (zie later).

HO GENT

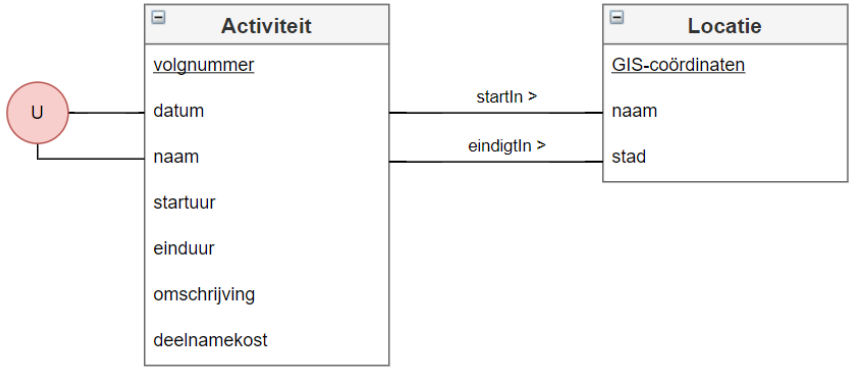
Cardinaliteiten

- Voorbeelden
  - Een student kan voor minimaal 0 en voor maximaal M vakken ingeschreven zijn.
    Voor een vak kunnen minimaal 0 en maximaal N studenten ingeschreven zijn.
    → M:N (een veel-op-veel-relatie)
  - Een bachelorproef kan aan minimaal 0 en aan maximaal 1 student toegekend zijn.
    Een student kan minimaal 0 en maximaal 1 bachelorproef uitwerken.
    → 1:1 (een één-op-één-relatie)
  - Een project wordt geleid door juist 1 werknemer.
    Een werknemer kan min 0 en max N projecten leiden.
    → 1:N (een één-op-veel-relatie)

HO GENT

Oefening Activiteiten

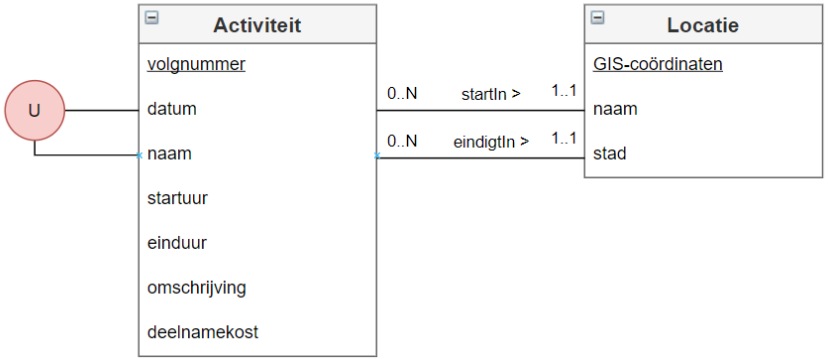
- Je wil een activiteitendatabank creëren
- Elke activiteit krijgt een uniek volgnummer, een datum, een naam, een startuur, vermoedelijk einduur, een korte omschrijving en een kostprijs om deel te nemen. De combinatie van datum en naam is ook uniek.
- Elke activiteit start en eindigt op een bepaalde locatie.
  Een locatie heeft unieke GIS-coördinaten, een naam en een stad.
- Geef de cardinaliteiten



HO GENT

Oefening Activiteiten

- Je wil een activiteitendatabank creëren
- Elke activiteit krijgt een uniek volgnummer, een datum, een naam, een startuur, vermoedelijk einduur, een korte omschrijving en een kostprijs om deel te nemen. De combinatie van datum en naam is ook uniek.
- Elke activiteit start en eindigt op een bepaalde locatie.
  Een locatie heeft unieke GIS-coördinaten, een naam en een stad.
- Geef de cardinaliteiten

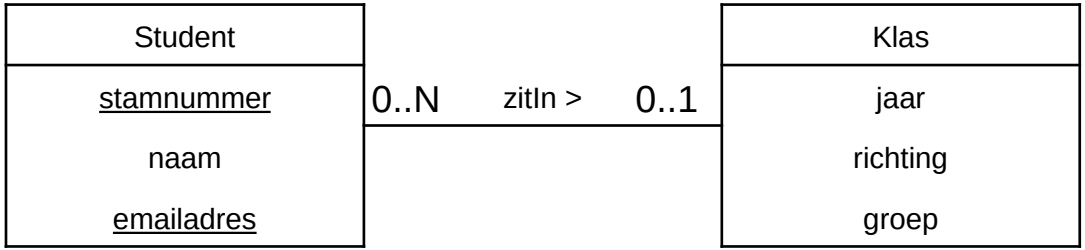


HO GENT

3. Oefeningen

HO
GENT

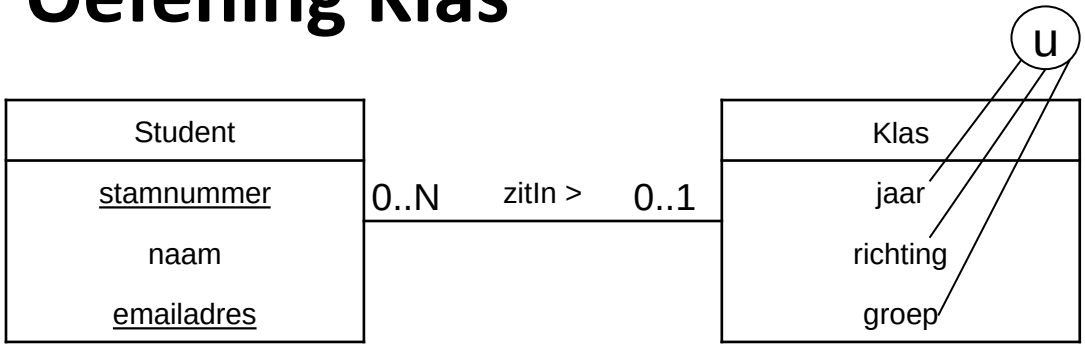
Oefening Klas



- Een student kan worden geïdentificeerd aan de hand van zijn stamnummer of aan de hand van zijn e-mailadres. Een student heeft een naam.
- Een klas heeft niet verplicht meerdere studenten.
- Een student zit maximaal in 1 klas.
- De klasgroepen zijn genummerd per jaar en richting

HO GENT

Oefening Klas

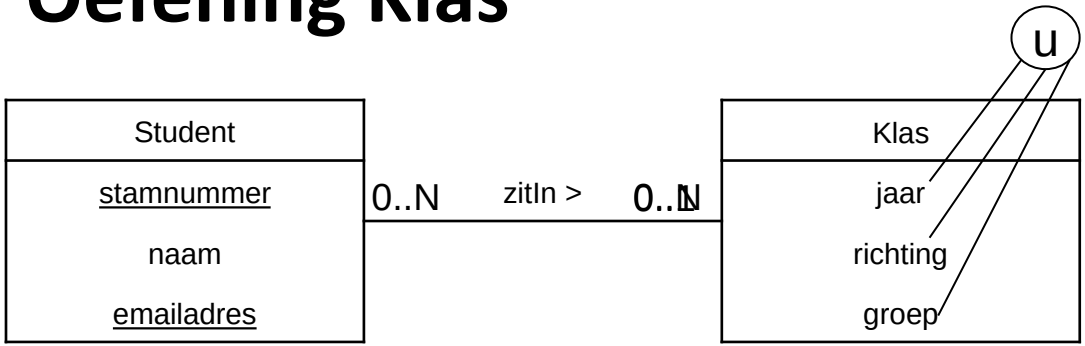


Pas het ERD aan

- Een klas wordt uniek geïdentificeerd aan de hand van de combinatie van jaar, richting en groep.

HO GENT

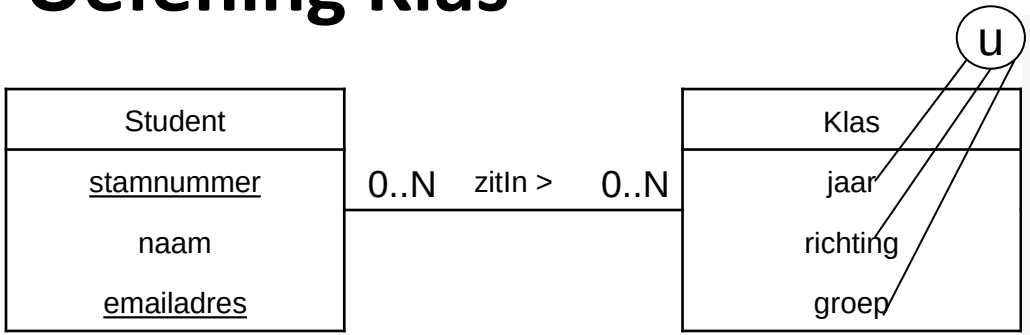
Oefening Klas



Pas het ERD aan

- Een student kan in meerdere klassen zitten.

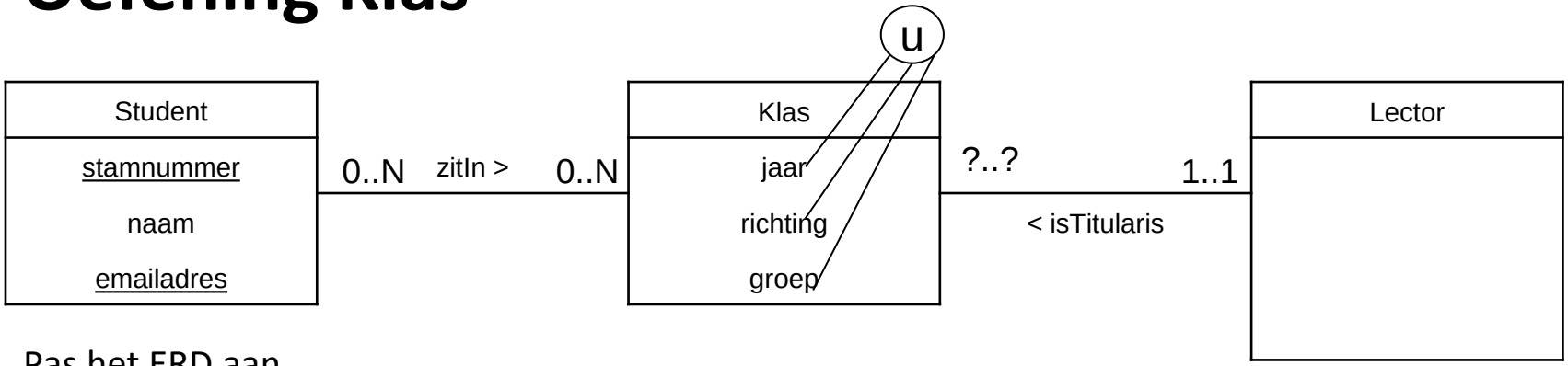
Oefening Klas



Pas het ERD aan

- Elke klas heeft juist één lector als titularis.

Oefening Klas

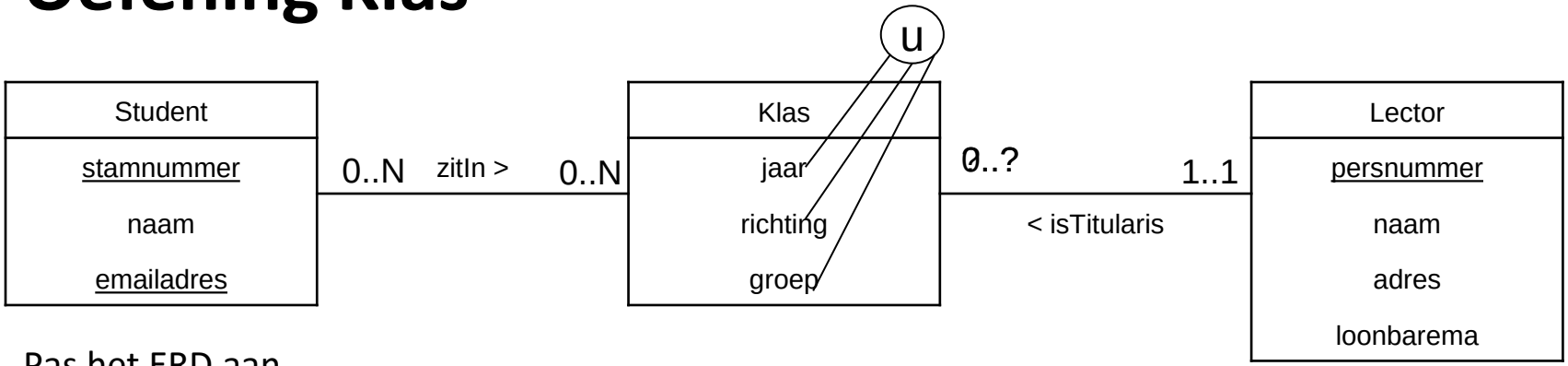


Pas het ERD aan

- Van een lector worden een uniek personeelsnummer, naam, adres en loonbarema bijgehouden

HO GENT

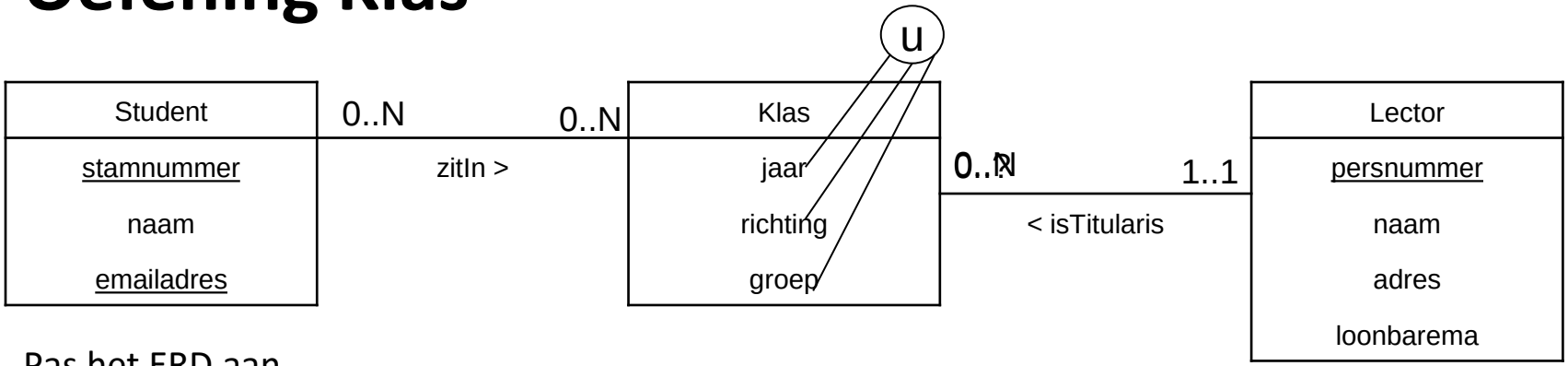
Oefening Klas



Pas het ERD aan

- Een lector hoeft geen titularis te zijn

Oefening Klas



Pas het ERD aan

- Een lector kan titularis zijn van meerdere klassen

Oefening Bibliotheek

- Een bibliotheek wil een databank ontwerpen voor het bijhouden van informatie over de boeken die er aanwezig zijn.
- Van elk boek is maar 1 exemplaar aanwezig in de bibliotheek.
- Er moet kunnen opgevraagd worden of een boek aanwezig is of niet en zo ja in welke rij het kan gevonden worden.
- Van elk boek moet volgende info kunnen opgevraagd worden: ISBN (unieke identificatie van een boek), titel, auteur(s), aantal blz, prijs, korte inhoud.
- Elk boek heeft minstens 1 auteur. Van die auteur kennen we een unieke naam.

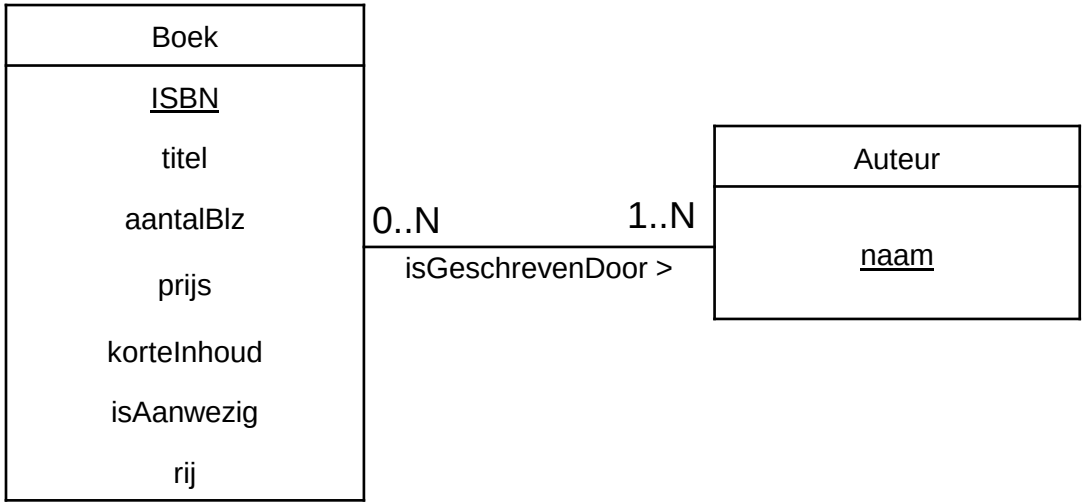
HO GENT

Oefening Bibliotheek

Boek
ISBN
titel
auteurs
aantalBlz
prijs
korteInhoud
isAanwezig
rij

HO GENT

Oefening Bibliotheek



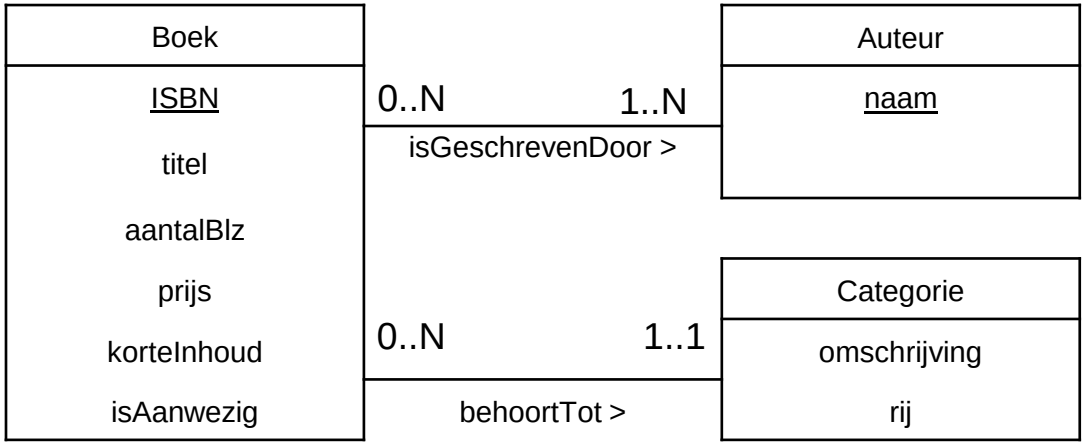
HO GENT

Oefening Bibliotheek

Breid het ERD uit:

- Elk boek behoort tot een bepaalde categorie (historische roman, thriller, ….).
- Elke categorie heeft een eigen plaats (rij) in de bibliotheek.

Oefening Bibliotheek



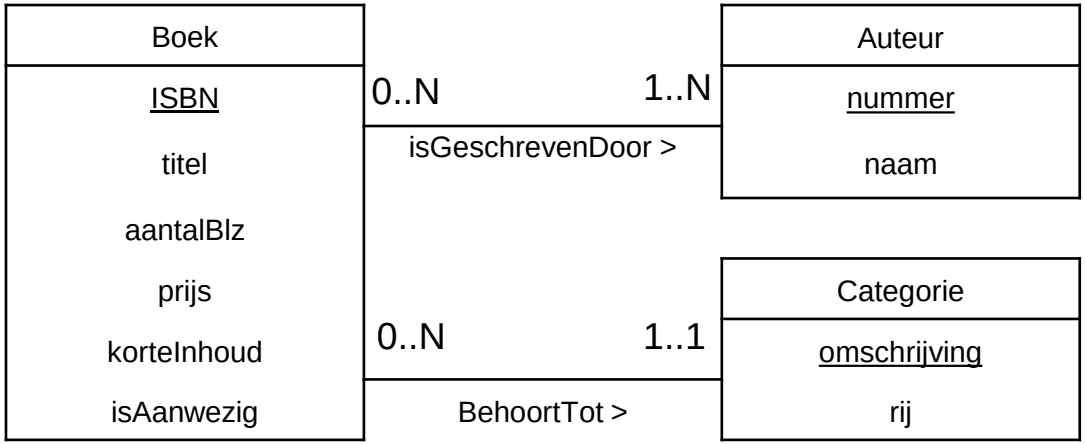
HO GENT

Oefening Bibliotheek

- Elk entiteittype moet zich kunnen identificeren.
- Hier hebben we geen identifier voor Categorie.
- Als ontwerper mag je nooit zelf iets beslissen maar moet je overleggen met de opdrachtgever!
- Na overleg blijkt dat
  - een categorie geïdentificeerd wordt aan de hand van zijn omschrijving
  - naam van een auteur is geen goede identificatie → een oplopend nummer
- Pas het ERD aan!

HO GENT

Oefening Bibliotheek



HO GENT

4. Bronnen

HO
GENT

Bronnen

- Principles of database management (W. Lemahieu, S. Vanden Broucke, B. Baesens)
  - 3.1 + 3.2.1 + 3.2.2 + 3.2.3 + 3.2.4 + 3.2.6
- Principes van databases (G. De Tré)

HO GENT