



# Informatie Systeem Ontwikkeling ISO 2R290

docent:  
**Prof. dr. Paul De Bra**

Gebaseerd op:  
**Database System Concepts, 5th Ed.**  
©Silberschatz, Korth and Sudarshan



## doel van dit vak

- kennis van en inzicht in basisbegrippen over informatiesystemen
- kennis van en ervaring met technieken voor de ontwikkeling en het gebruik van informatiesystemen, met name DATABASES
- inzicht en ervaring in het ontwikkelingsproces voor informatiesystemen, met name het DATAMODELLEREN





## onderwerpen college/instructies

### college

- informatiesystemen
- relationele databases
- relationele algebra:  
rekenen met tabellen
- “querying”: SQL / QBE
- het E-R model
- datamodelleren

### instructie

- oefeningen rekenen  
met tabellen
- oefeningen querying  
met SQL en QBE
- oefeningen data-  
modelleren en E-R
- proeftentamen



## rooster

- colleges: woensdag 1e en 2e uur op  
6/9, 13/9, 20/9, 27/9, 11/10, 18/10, 25/10, 1/11  
(dus niet op 4/10, en wellicht ook niet meer op 8/11)
- instructies: maandag 5e en 6e uur op  
11/9, 18/9, 25/9, 2/10, 16/10, 23/10, 30/10,  
6/11 (dus niet op 4/9, en misschien nog wel op 6/11)
- In het tweede trimester worden de kennis en  
vaardigheden verdiept in OGO 1.2.





## organisatie / contact

### ■ docent:

Paul De Bra

Sectie Informatiesystemen

HG 7.72

<http://wwwis.win.tue.nl/~debra/>

[debra@win.tue.nl](mailto:debra@win.tue.nl)

### ■ ISO site:

<http://wwwis.win.tue.nl/iso/>

...of contacteer je instructeur!



## waar komen informatiesystemen voor?

### ■ overall waar gegevens/informatie moeten worden opgeslagen, gebruikt en/of veranderd:

- *fabriek, winkel, belastingdienst, gemeente, bank, verzekeringsmaatschappij, ziekenhuis, reisburo...*  
organisaties met **administratieve** gegevens
- *krant en nieuws-site, museum site, digitale bibliotheek, zoekmachine, ...*  
organisaties met **documentaire** gegevens
- *telefoon en telefooncentrale, vliegtuig, space-shuttle, televisie en video, (wekker)radio, kamerthermostaat, ...*  
apparaten met **technische** gegevens

### ■ in ISO bestuderen we **administratieve IS**



## informatiesysteem en objectsysteem

- objectsysteem:  
de “real world” van een bedrijf of organisatie:  
mensen, machines, producten, warenhuizen
- informatiesysteem:  
een *representatie* van de werkelijkheid in de  
computer, zoals getallen die de voorraad  
beschrijven van producten die met een code  
of nummer worden beschreven



## waarom gebruikt een organisatie een IS?

- een IS ondersteunt de bedrijfsprocessen;  
voorbeeld informatiebehoefte in autobedrijf:
  - verkopen nieuwe auto's
  - verkopen occasions
  - inkopen nieuwe auto's
  - inkopen occasions
  - uitvoeren onderhoud
  - beheren onderdelen (inkoop, voorraad, verkoop)
  - betalen salaris personeel
  - onderhoud bedrijfsruimte en -kleding
  - etc.





## IS als strategisch hulpmiddel

- Met een goed IS (en on-line toegang) kan een organisatie zichzelf aantrekkelijk maken:
  - on-line configureren en prijs-berekenen van auto
  - automatische berichten over onderhoud, apk, vervanging, etc.
  - anticiperen van benodigde onderdelen voor onderhoud
  - tracking van bestellingen en levering: post-, pakket- en koerierbedrijven lokken klanten met tracking/tracing faciliteiten
  - on-line checken van prijzen en beschikbaarheid van producten in winkels
  - directe en indirecte communicatie tussen klanten



## modellieren

- welke informatie
  - wat is de *structuur* van de gegevens
  - wat zijn de *verbanden* tussen de gegevens
  - welke *beperkingen* zijn er op de gegevens

### DATAMODELLEREN

- hoe wordt informatie verwerkt
  - hoe en wanneer komt de informatie tot stand
  - hoe wordt informatie *bewerkt*
  - hoe wordt informatie *gecommuniceerd* met andere delen van de organisatie of andere partijen

### PROCESMODELLEREN





## waarom database systemen?

- de eerste “informatiesystemen” gebruikten “gewone” bestanden; nadelen:
  - data redundantie en inconsistenties
    - ▶ verschillende bestandsformaten, duplicatie van informatie in verschillende bestanden
  - moeilijke toegankelijkheid van informatie
    - ▶ voor elke taak is een nieuw programma nodig
  - data beperkingen zijn moeilijk te bewaken
    - ▶ elk programma moet afzonderlijk de gemodelleerde beperkingen bewaken
    - ▶ het is moeilijk om beperkingen toe te voegen of te wijzigen



## waarom database systemen (cont.)

- nadelen van gewone bestanden (cont.):
  - bewerkingen moeten *atomair* zijn
    - ▶ elke bewerking moet helemaal of helemaal niet gebeuren: voorbeeld: overboeking tussen bankrekeningen
  - gelijktijdig gebruik door verschillende gebruikers
    - ▶ vele mensen moeten dezelfde bestanden tegelijk kunnen gebruiken: vb: klanten moeten tegelijk kunnen winkelen
    - ▶ gelijktijdig gebruik van bestanden kan tot fouten leiden:
      - vb: twee transacties die op eenzelfde bankrekening geld bijschrijven (lezen oud saldo, schrijven nieuw saldo)
  - beveiliging van gegevens
    - ▶ moeilijk om gebruikers toegang te geven tot sommige informatie uit een bestand en niet tot de rest van het bestand
- database systemen lossen al deze problemen op





## instantie en schema

- **schema** – de structuur van de database
  - vb: een universiteitsdatabase bevat informatie over studenten, docenten en colleges en de verbanden daartussen
  - het “programma” om de namen van de studenten te tonen die het college 2R290 volgen is altijd hetzelfde, onafhankelijk van welke studenten dat zijn
- **instantie** – de werkelijke inhoud van de database op een bepaald moment
  - vb: de studenten die dit jaar 2R290 volgen is een instantie, en is verschillend van de instantie van vorig jaar.
- we beschrijven bewerkingen op een database altijd op het niveau van het schema
  - je kunt de correctheid van een bewerking nooit testen op een instantie!



## data manipulatie talen

- we bewerken gegevens met speciale data manipulatie talen
  - ze zijn ook bekend als “query talen”
- twee klassen van query talen
  - **Procedureel** – is als programmeren: je beschrijft wat de computer moet *doen* om het gewenste resultaat te bereiken
  - **Declaratief (niet-procedureel)** – je specificeert *wat het gewenste resultaat is* maar niet hoe dat moet worden berekend



## fysiek / logisch / view niveau

- een database systeem scheidt het fysieke en logische niveau
  - elk database management system (dbms) vertaalt logische structuren verschillend naar fysieke opslag op harde schijven
  - query talen werken op het hogere *logische niveau* van het relationele database model
- applicaties bieden *views* op de data aan
  - eindgebruikers krijgen meestal een *beperkt* en *vereenvoudigd* beeld op de database structuur

