

H1: Databanken inleiding

Basisbegrippen.

In dit hoofdstuk

- Wie gebruikt databank(technologie)?
- Van bestand naar databank
- (R)DBMS
- Delen van een databanksysteem
- Belangrijke begrippen

**Wie gebruikt
databank(technologie)?**

Wie gebruikt databank(technologie)?

- Traditionele bedrijfsapplicaties (loonberekening, tijdsregistratie, ...)
- Biometrische applicaties (vingerafdrukken, resultaten scans)
- Sensor-applicaties (in kerncentrales, ...)
- GIS applicaties (geografische informatie systemen (Google Maps, ...))
- Big Data applicaties (Walmart, ...)
- ‘Internet of Things (IoT)’ applicaties (Telematics, ...)

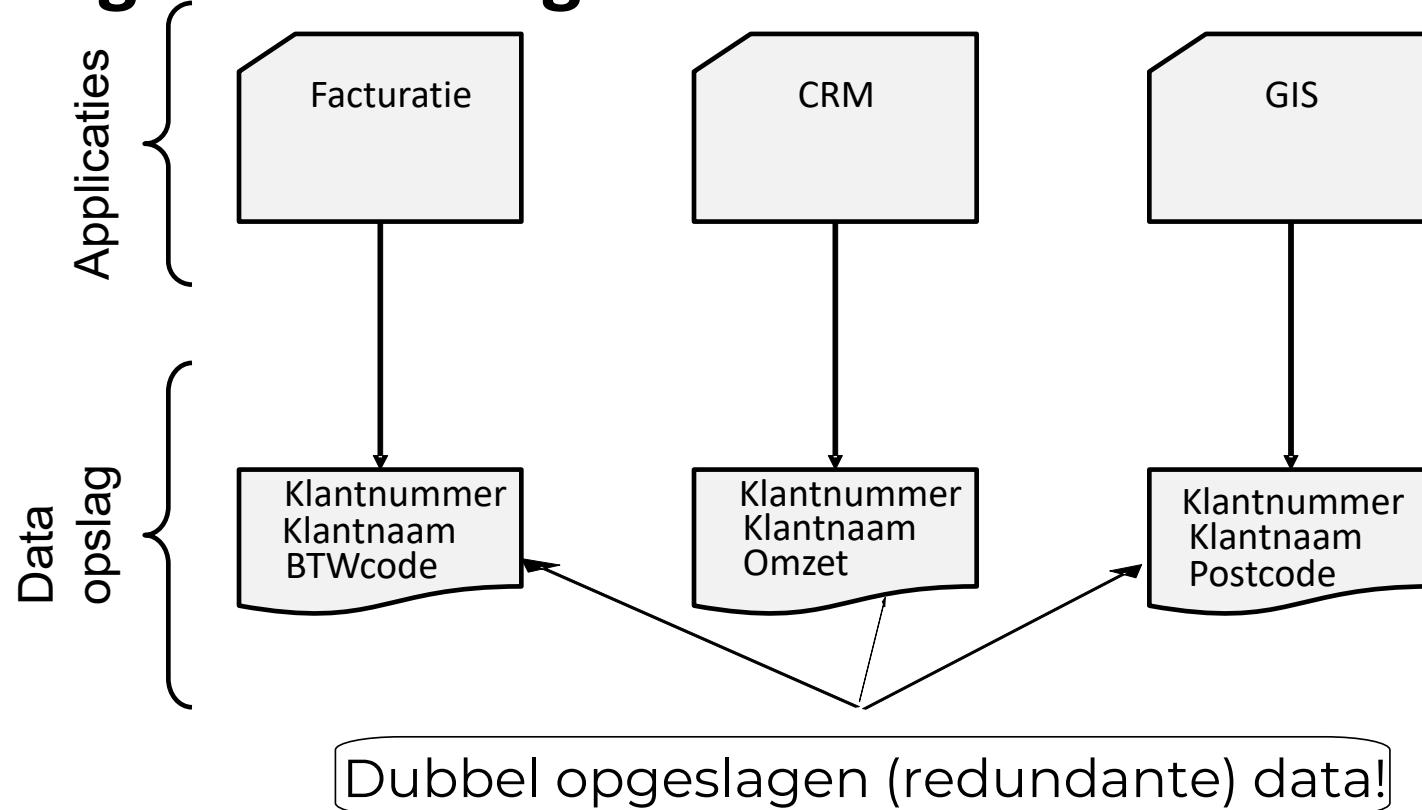
→ opslag en terug ophalen van informatie (data)

Van bestand naar databank

Gegevensmanagement via bestanden

- Bestandsgebaseerde oplossing → elke toepassing definieert zijn eigen bestanden.
- => Er wordt dus gebruik gemaakt van verschillende bestanden zonder relaties tussen de bestanden.

Gegevensmanagement via bestanden



Nadelen gegevensmanagement via bestanden

Waarom zijn de rijen 9 en 10 leeg?	Hoe kan men een alfabetische lijst van de werknemers creëren?	Hoe kan je het aantal werknemers met het certificaat 'Basic Database Manipulation' tellen?	Is Basic Database Manipulation hetzelfde als Basic DB Manipulation?	Was als een werknemer een vierde certificaat verwerft?						
ID	Enum	Name	Title	HireDate	Skill1	Skill1Date	Skill2	Skill2Date	Skill3	Skill3Date
1	12345	Biran Oates	DBA	2/14/2005	Basic Database Management	2/14/2012	Advanced Database Management	2/14/2015	Basic Web Design	8/09/2018
2	18273	Marco Bienz	Analyst	7/28/2016	Basic Web Design	3/08/2019	Advance Process Modeling	8/19/2022		
3	16234	Jasmine Patel	Programmer	8/10/2015	Basic Web Design	8/10/2017	Advanced C# programming	8/10/2017	Basic DB manipulation	1/29/2022
4	13373	Franklin Johnson, Jr.	Purchasing Agent	3/15/2012	Advanced Spreadsheets	6/20/2021				
5	13567	Almond, Robert	Analyst	9/30/2022	Basic Process Modeling	9/30/2023	Basic Database Design	5/23/2024		
6	10282	Richardson, Amanda	Clerk	4/11/2021						
7	19382	Susan Mathis	Database Programmer	8/02/2020	Basic DB Design	8/02/2022	Basic Database Manipulation	8/02/2022	Advanced DB Manipulation	5/01/2023
8	14311	Duong, Lee	Programmer	9/01/2024	Basic Web Design	9/02/2024				
9					Master Database Programming					
10					Basic Spreadsheets					
11	19002	Wade Gaither	Clerk	5/20/2020	Advanced Spreadsheets	5/16/2023	Basic Web Design	5/16/2023		
12	13383	Raymond F. Matthews	Programmer	3/12/2022	Basic C# Programming	3/12/2024				
13	19283	Chavez, Juan	Clerk	7/04/2020						
14	14893	Patricia Richards	DBA	6/11/2014	Advanced Database Management	6/11/2016	Advanced Database Manipulation	9/20/2022		
15	13932	Lee, Megan	Programmer	9/29/2023						

Nadelen gegevensmanagement via bestanden

- Nadeel 1: Verspreiding en isolatie van gegevens
- Nadeel 2: Gegevensredundantie

HR – afdeling

Payroll - afdeling

Bestand met
data van
werknemers

Bestand met
data van
werknemers

- Nadeel 3: Data afhankelijkheid

id	voornaam	familienaam	adres	pc	woonplaats	tel	statuut	loon	
0	5	35	90	145	155	200	215	260	280

id	voornaam	tweede naam	familienaam	adres	pc	woonplaats	tel	statuut	loon	
0	5	35	65	120	175	185	230	245	290	310

Nadelen gegevensmanagement via bestanden

- Nadeel 4: Incompatibiliteit



Payroll – afdeling

COBOL -
bestand met
data van
werknemers



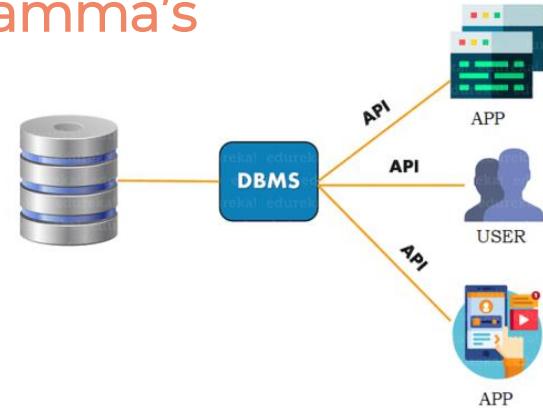
- Nadeel 5: Fixed queries

Basisdefinities: databank

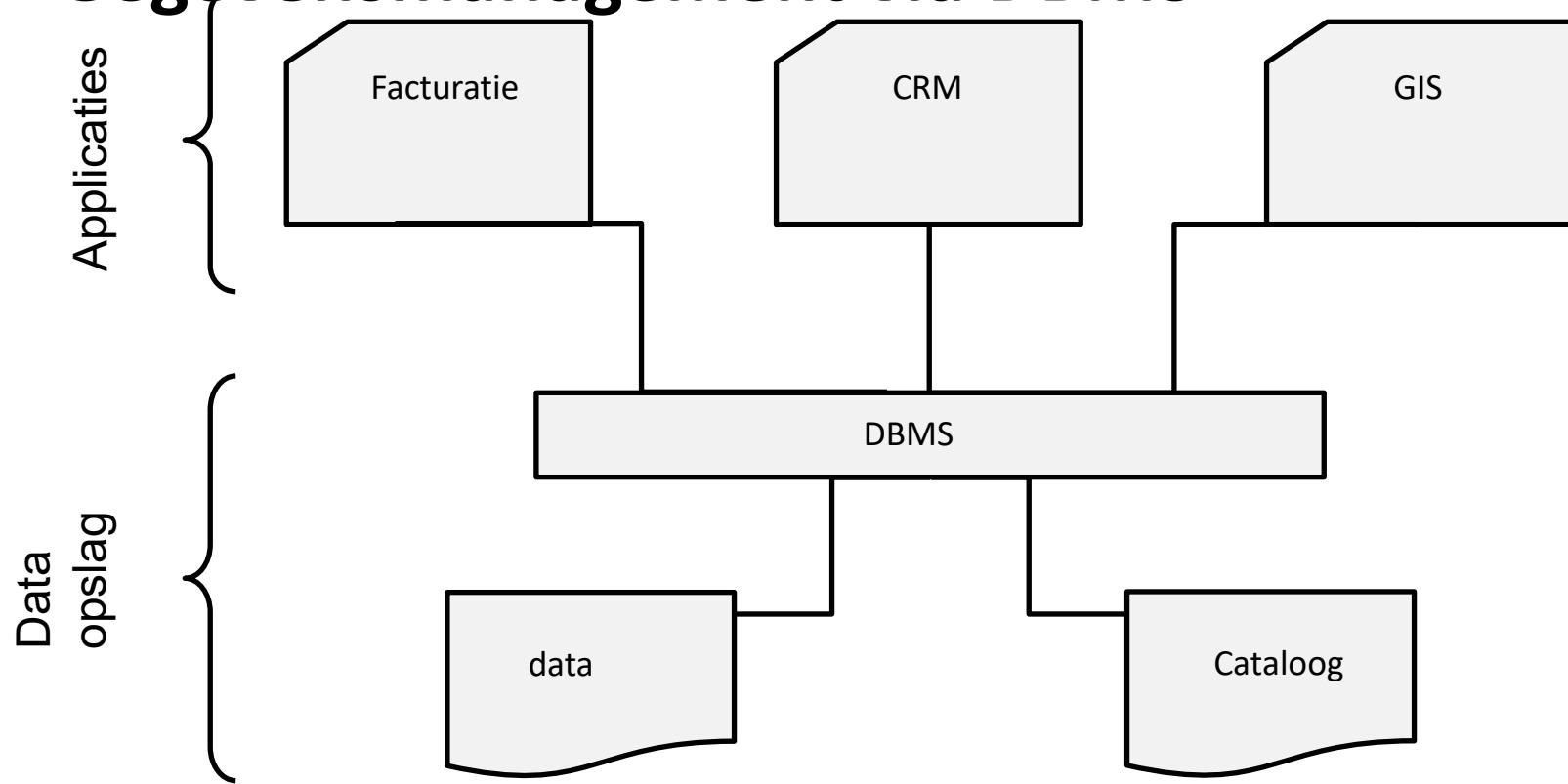
- Een **gedeelde** verzameling van **logisch** met elkaar **verbonden** gegevens en hun **beschrijving**, ontworpen om aan de **informatienoden** van een organisatie te voldoen (T. Connolly)
 - digitaal opgeslagen
 - specifiek bedrijfsproces
 - specifieke groep (gebruikers en applicaties)

Basisdefinities: DBMS

- Een Database Management System (DBMS)
 - Een **verzameling computerprogramma's** (softwaremodules)
 - Nodig om een databank te ..
 - definiëren
 - creëren
 - wijzigen
 - beheren (backups, monitoring, optimalisatie)
 - gebruiken (gegevens invoeren en 'lezen')
- Databanksysteem: databank + DBMS



Gegevensmanagement via DBMS



Basisdefinities: RDBMS

- RDBMS = DBMS voor **relationele** databanken
 - Maakt gebruik van het relationeel gegevensmodel (met tabellen, rijen, kolommen, ...)
 - Momenteel het meest frequent gebruikt in de bedrijfswereld
 - SQL (beschrijvend en gebaseerd op resultset)

Voorbeelden van RDBMS

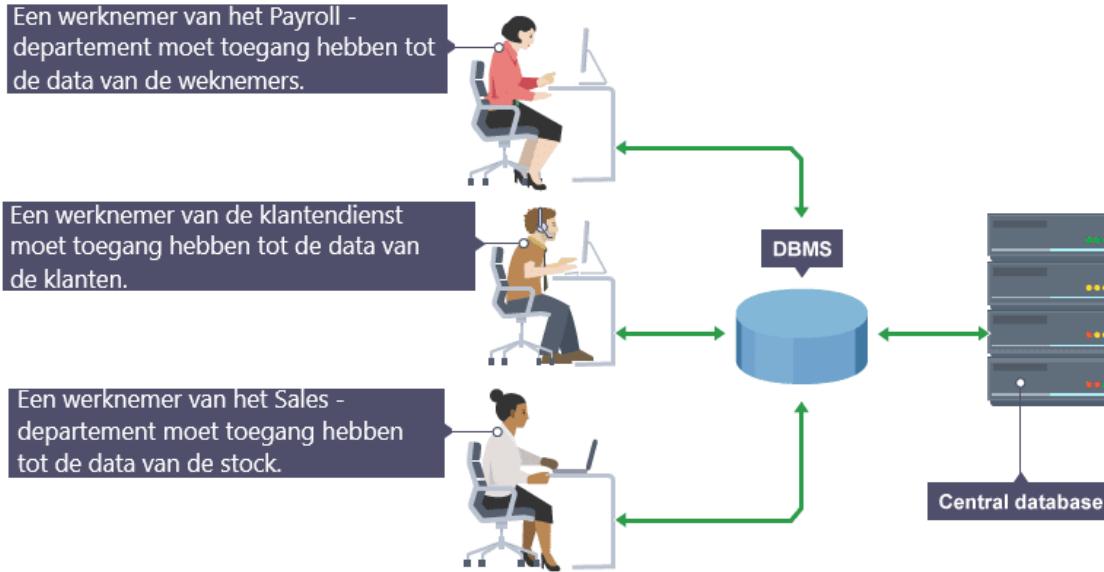
	Het grootste en eerste commerciële RDBMS. Wordt gebruikt in veel van 's werelds grootste bedrijven.
	RDBMS-product van Microsoft. Leverbaar in vele versies voor verschillende bedrijfsbehoeften.
	Het populairste open source RDBMS. Sinds 2010 wordt het ontwikkeld, gedistribueerd en ondersteund door Oracle Corporation.
	Ook een gratis, open source RDBMS. Sommigen zouden zeggen krachtiger dan MySQL

In één dag 101 miljard rijker...



Basisdefinities: Toepassingsprogramma's.

- Toepassingssoftware zorgt voor de verbinding met het DBMS. DBMS verzorgt de toegang tot de relevante



Basisdefinities: Databank gebruikers

- De data-administrators (DA) zijn in een onderneming centraal verantwoordelijk zijn voor de data
- Dbontwerper vertaalt conceptueel model naar logisch en intern model
- DBA (databankbeheerder of database administrator) implementeert en monitort DB
- Applicatieontwikkelaar schrijft databankprogramma's/databankapplicaties

Basisdefinities: Databank gebruikers

- Eindgebruikers gebruikt databankapplicaties en voert **op die manier** databankacties uit
 - Sommige eindgebruikers zijn zich niet bewust van de databank
 - Sommige geavanceerde eindgebruikers kennen de structuur van de databank

Delen van een databanksysteem

Delen van een databanksysteem

- Databankmodel versus instances
- Catalog
- Databankmodel
- 3-lagen architectuur

Databankmodel versus instances

- Databankmodel = databankschema
 - bevat
 - beschrijving van de databankstructuur
 - specificaties v/d elementen, hun eigenschappen, relaties, beperkingen, ...
 - Opgesteld tijdens databankontwerp
 - Wijzigt niet om de haverklap
 - Opgeslagen in de catalogus
- Toestand van een databank
 - Op dat ogenblik aanwezige data
 - Wijzigt voortdurend

Databankmodel versus instances

- Voorbeeld databankmodel :

Kunstenaar (naam, geboorteplaats, geboortedatum)

Kunstwerk (naam, museum, jaar)

Museum (naam, stad)

Databankmodel versus instances

- Toestand van de databank :

MUSEUM

naam	stad
Sint-Pietersbasiliek	Rome
Museo Reina Sofia	Madrid
Gemäldeoalerie van de Staatliche Museen	Berlin

KUNSTENAAR

naam	geboorteplaats	geboortedatum
Michelangelo	Caprese	06/03/1475
Rembrandt	Leiden	15/07/1606
Picasso	Malaga	25/08/1881

KUNSTWERK

naam	museum	jaar
Guernica	Museo Reina Sofia	1937
Christuskoop	Gemäldeoalerie van de Staatliche Museen	1648
Pieta	Sint-Pietersbasiliek	1499

Datamodel

- **Databankmodel**: bestaat uit verschillende datamodellen
- **Datamodel**: weergave van de gegevens met hun kenmerken en hun relaties
- In de cursus komen er 3 datamodellen aan bod:
 - Conceptueel datamodel
 - Logisch datamodel
 - Fysiek datamodel

Datamodel

- **Conceptueel datamodel :**
 - perfecte weergave van de gegevensvereisten van de ‘business’
 - algemene beschrijving gegevenselementen, kenmerken en relaties
 - Gebruikt door ‘IT’ en ‘business’
 - Weergave hoe ‘de business’ de gegevens ziet
 - Voorstelling: (E)ERD diagram
- **!!Veronderstellingen en ontbrekende informatie duidelijk vermelden!!**

Datamodel

- **Logisch datamodel**
 - vertaling conceptueel gegevensmodel naar het type databankmodel
 - Relatieel, hiërarchisch, OO, XML, NoSQL
 - nog altijd verstaanbaar voor niet IT-ers, maar leunt al dichter aan bij hoe de data fysiek zal opgeslagen worden

Datamodel

- **Fysiek datamodel**
 - Geeft informatie over fysieke opslag:
 - waar worden welke gegevens opgeslagen
 - Wat is de grootte van de datavelden
 - Indexen die het ophalen versnellen
 - Zeer DBMS afhankelijk



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.

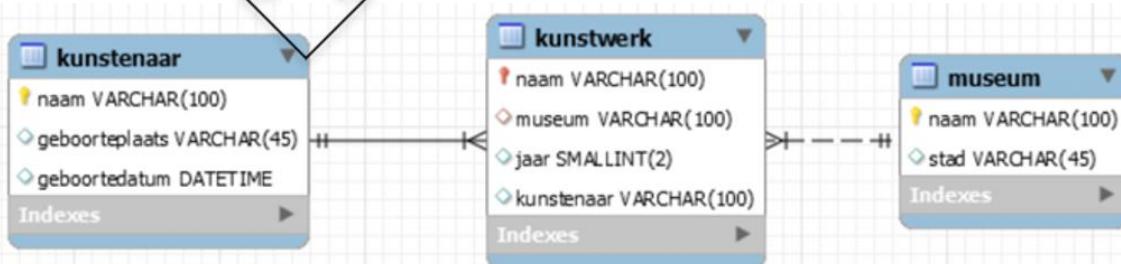
Kunstenaar(naam, geboorteplaats, geboortedatum)

Kunstwerk(naam, jaar, naam kunstenaar, naam museum)

Museum(naam, stad)

Logisch model

We voorzien scripts om de databank fysisch te creëren.



Fysisch model

```

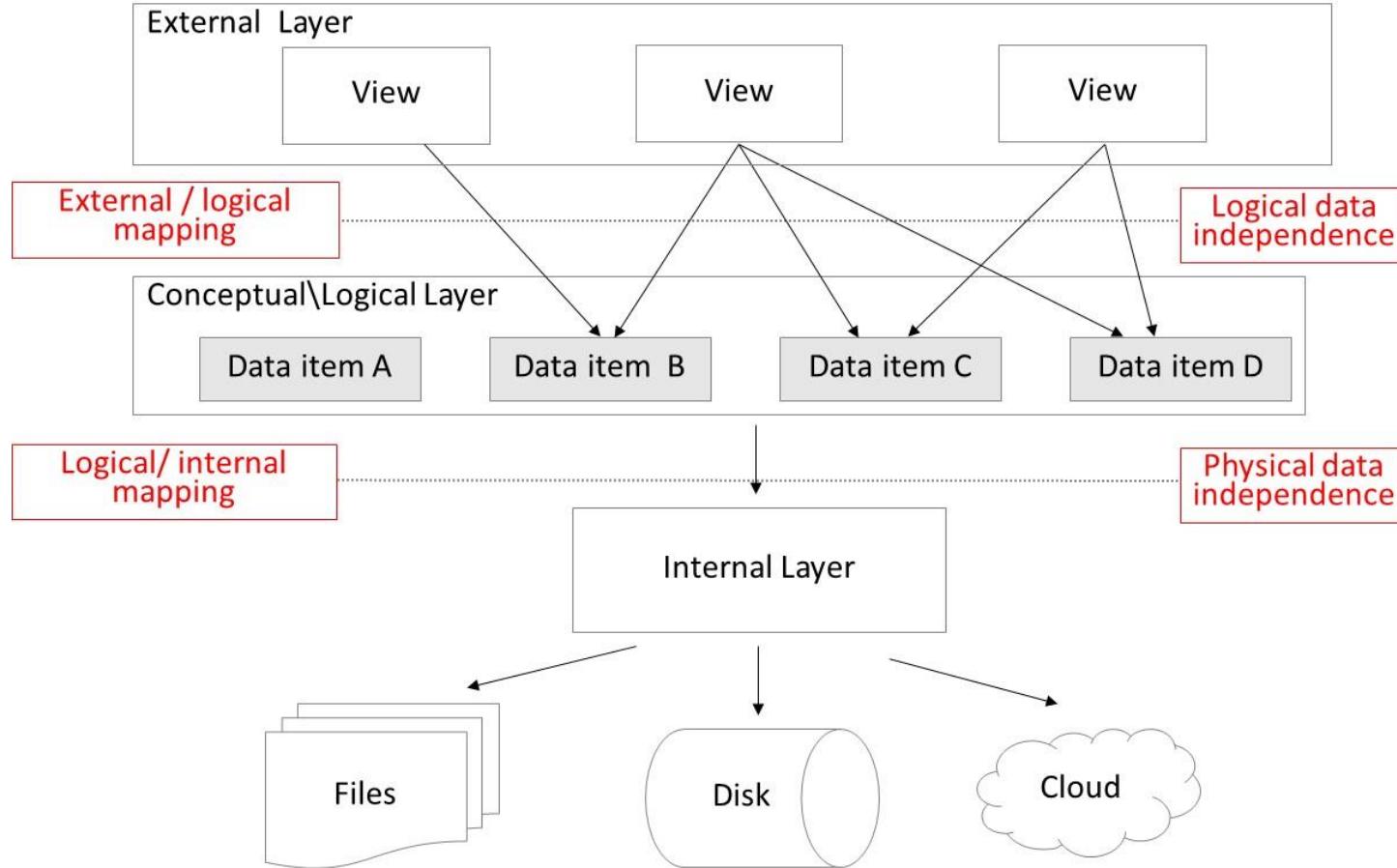
CREATE TABLE `kunstenaar` (
  `naam` varchar(100) NOT NULL,
  `geboorteplaats` varchar(45) DEFAULT NULL,
  `geboortedatum` datetime DEFAULT NULL,
  PRIMARY KEY (`naam`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8

```

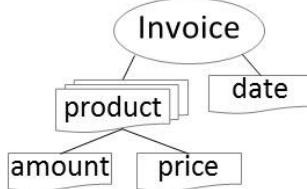
drielagen architectuur

- Doel: verkrijgen van dataonafhankelijkheid
 - Wijzigingen in de ene laag leiden tot minimale wijzigingen in de andere laag
- Externe laag → externe datamodel, bevat **views**
 - Deelverzameling van het logisch model
 - Voor een specifieke doelgroep.
 - Wordt gebruikt om de toegang tot gegevens te controleren en beveiliging af te dwingen
- Middelste laag → conceptueel / logisch model
- Interne laag → fysieke datamodel, legt vast hoe data fysisch georganiseerd en opgeslagen wordt

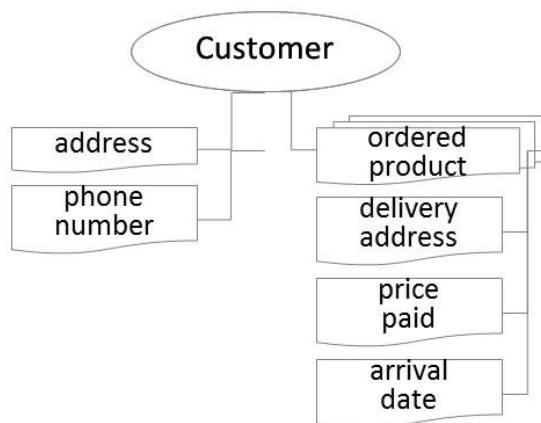
Inleiding



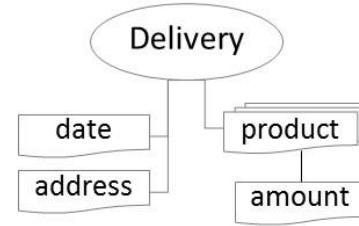
Finance department



Customer service



Logistics department



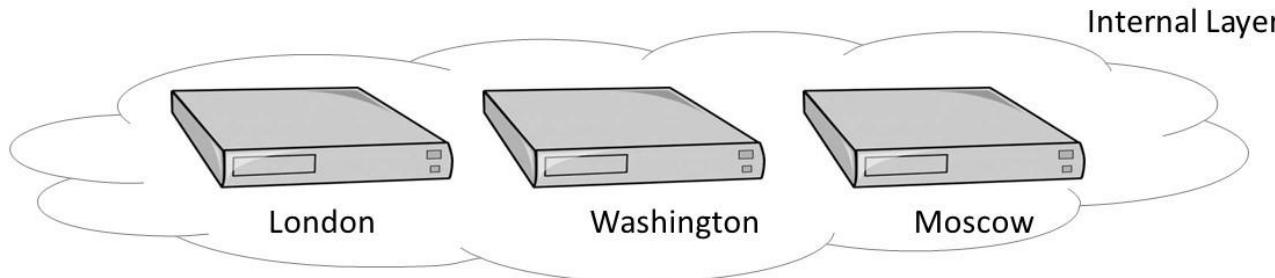
View = een deel van de databank waarin een bepaalde applicatie of gebruikersgroep geïnteresseerd is + verbergt de rest van de databank.

Product
Customer
Invoice
Delivery

name, description, cost, ...
name, phone, address, ...
customer, date, products (with price and amount), ...
invoice, address, date, ...

Conceptual\Logical Layer

Conceptuele / logische datamodel



interne laag bevat het interne datamodel

Belangrijke begrippen

Nog enkele belangrijke begrippen

- Gegevensonafhankelijkheid
- Gestruktureerde en ongestruktureerde gegevens
- Redundante gegevens
- Integriteitsregels
- Catalogus

Gegevensonafhankelijkheid

- = wijzigingen aan de gegevensbeschrijving hebben weinig tot geen impact op de applicaties
 - Fysieke gegevensonafhankelijkheid: wijzigingen van de opslagspecificaties hebben geen invloed op het logisch model noch op de applicatie
→ wordt opgevangen door het DBMS
 - Logische gegevensonafhankelijkheid: minimale aanpassingen aan de applicaties bij wijzigingen aan het logisch model

(On)gestructureerde gegevens

- **Gestructureerde gegevens**
 - Kunnen in een logisch datamodel voorgesteld worden
 - Integriteitsregels kunnen opgesteld en afgedwongen worden
 - Vereenvoudigen, opzoeken, verwerken en analyseren
 - Voorbeelden: naam, geboortejaar, geboorteplaats van een kunstenaar

(On)gestructureerde gegevens

- **Ongestructureerde gegevens**
 - kunnen niet op een zinvolle manier worden geïnterpreteerd door een applicatie
 - Voorbeelden: gesprekken op social media, e-mails
 - Let op: er bestaat veel meer ongestructureerde data dan gestructureerde data

(On)gestructureerde gegevens

- **Semi-gestructureerde gegevens**
 - De structuur van de gegevens is zeer onregelmatig of zeer wisselend.
 - Voorbeelden: webpagina's van individuele gebruikers op een social media platform, cv-documenten in een personeelsdatabank

Redundante gegevens

- Databank = centrale en unieke opslag gegevens
- Soms worden databanken geduplicateerd uit veiligheidsoverwegingen of omwille van performantie
→**redundantie**
- DBMS is verantwoordelijk voor de synchronisatie en garandeert de juistheid van de gegevens

Integriteitsregels

- **Integriteitsregels** worden gedefinieerd op basis van het conceptueel model en opgeslagen in de catalogus
 - Worden afgedwongen door het DBMS
- Vastleggen hoe gegevens worden opgeslagen (syntactische regel)
 - Bvb: customerID is een geheel getal (100, 200, niet 2.A)
- Vastleggen wanneer gegevens correct zijn (semantische regel)
 - Bvb: Eenheidsprijs > 0; geboortedatum niet > vandaag
- Integriteitsregels worden gespecificeerd als onderdeel van het conceptuele/logische datamodel + centraal opgeslagen in de catalogus

Catalogoog

- Schatkist van DBMS
- Definities en beschrijving van de elementen in de DB (= metadata)
- Definities logisch gegevensmodel en intern gegevensmodel
- Zorgt voor synchronisatie en consistentie van de gegevensmodellen
- Opslagplaats voor integriteitsregels, en andere informatie zoals gebruikers, ...

Geschiedenis

Inleiding

