

Handleiding: Datamodelleren voor een Datawarehouse

1. Introductie

Datamodelleren is de fundering van een succesvol datawarehouse. Het zorgt ervoor dat ruwe data wordt gestructureerd op een manier die rapportages, analyses en automatisering mogelijk maakt. In deze handleiding leer je welke modellen bestaan, hoe je een goed datamodel opzet en welke tools je kunt gebruiken.

2. Wat is een datamodel?

Een datamodel is een visuele en logische representatie van hoe data georganiseerd wordt in een database. Het laat zien hoe tabellen met elkaar verbonden zijn en welke kolommen belangrijk zijn voor analyse.

3. Soorten datamodellen

Er zijn drie hoofdsoorten datamodellen:

- Conceptueel model – beschrijft op hoog niveau de entiteiten en relaties.
- Logisch model – specificeert tabellen, kolommen en relaties zonder implementatiedetails.
- Fysiek model – bevat implementatiedetails zoals datatypes, indexen en constraints.

4. Stermodel vs. Sneeuwvlokmodel

Het stermodel is het meest gebruikt in BI. Feittabellen staan in het midden en dimensionele tabellen eromheen. Het sneeuwvlokmodel normaliseert de dimensies verder, wat complexiteit toevoegt.

5. Principes voor goed modelleren

- Kies duidelijke sleutelvelden (bijv. klant_id, product_id).
- Vermijd redundantie waar mogelijk.
- Voeg alleen data toe die echt waarde toevoegt voor rapportage.
- Zorg voor tijdsdimensies: gebruik een datumtabel.

6. Tools voor datamodelleren

Populaire tools voor het ontwerpen van datamodellen zijn onder meer:

- • dbt (data build tool) – voor modelbeheer en transformaties in code.
- • Power BI – biedt visueel modelleren binnen de tool.
- • SQL Server Management Studio – goed voor relationele modellen.
- • Lucidchart / Draw.io – voor het visueel ontwerpen van tabellen en relaties.

7. Voorbeeldmodellering in SQL & Python

Voorbeeld: een eenvoudige omzet-feittabel met twee dimensies.

SQL:

```
CREATE TABLE fact_sales (
  sale_id INT PRIMARY KEY,
  customer_id INT,
  product_id INT,
  sale_date DATE,
  amount DECIMAL(10,2)
);
```

Python met pandas:

```
import pandas as pd
```

```
fact_sales = pd.DataFrame({
  'sale_id': [1, 2],
  'customer_id': [101, 102],
  'product_id': [1001, 1002],
  'sale_date': ['2024-01-01', '2024-01-02'],
  'amount': [500.00, 300.00]
})
```

8. Testen en valideren

Controleer:

- • Zijn alle sleutels uniek?
- • Zijn joins correct en leiden ze niet tot duplicatie?
- • Wordt de data correct weergegeven in rapportages?

9. Best practices

- • Gebruik duidelijke naamgeving.
- • Zorg voor documentatie van tabellen en velden.
- • Bouw iteratief – begin simpel, breid uit op basis van behoeften.

10. Conclusie

Een goed datamodel maakt het verschil tussen verwarring en inzicht. Met de juiste structuur en tools kun je snel waarde halen uit je data. Begin klein, documenteer je keuzes, en werk iteratief naar een volwassen model.