

Odissee  
DE CO-HOGESCHOOL

# Storage – SQL Databases



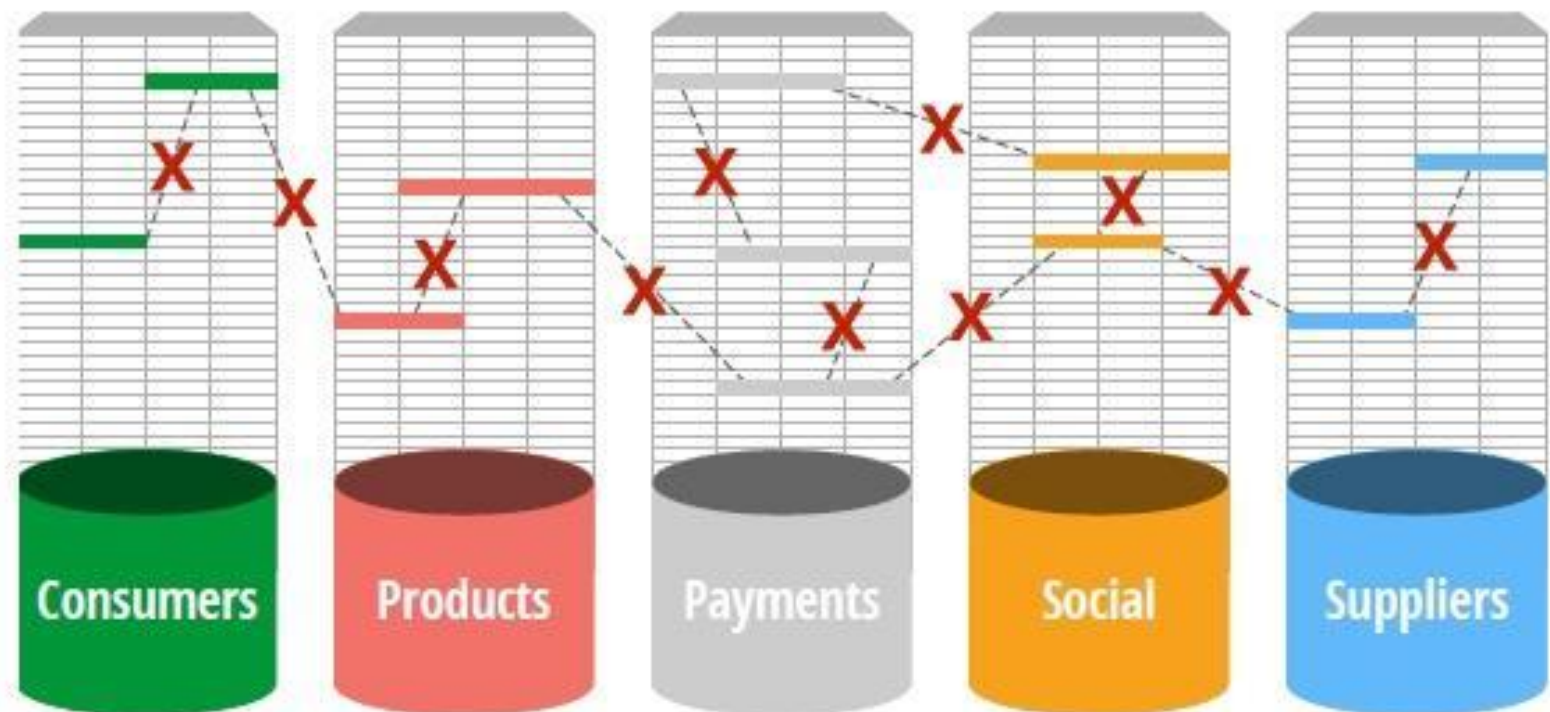
## Hoe zou je de data in een SQL-database definiëren?

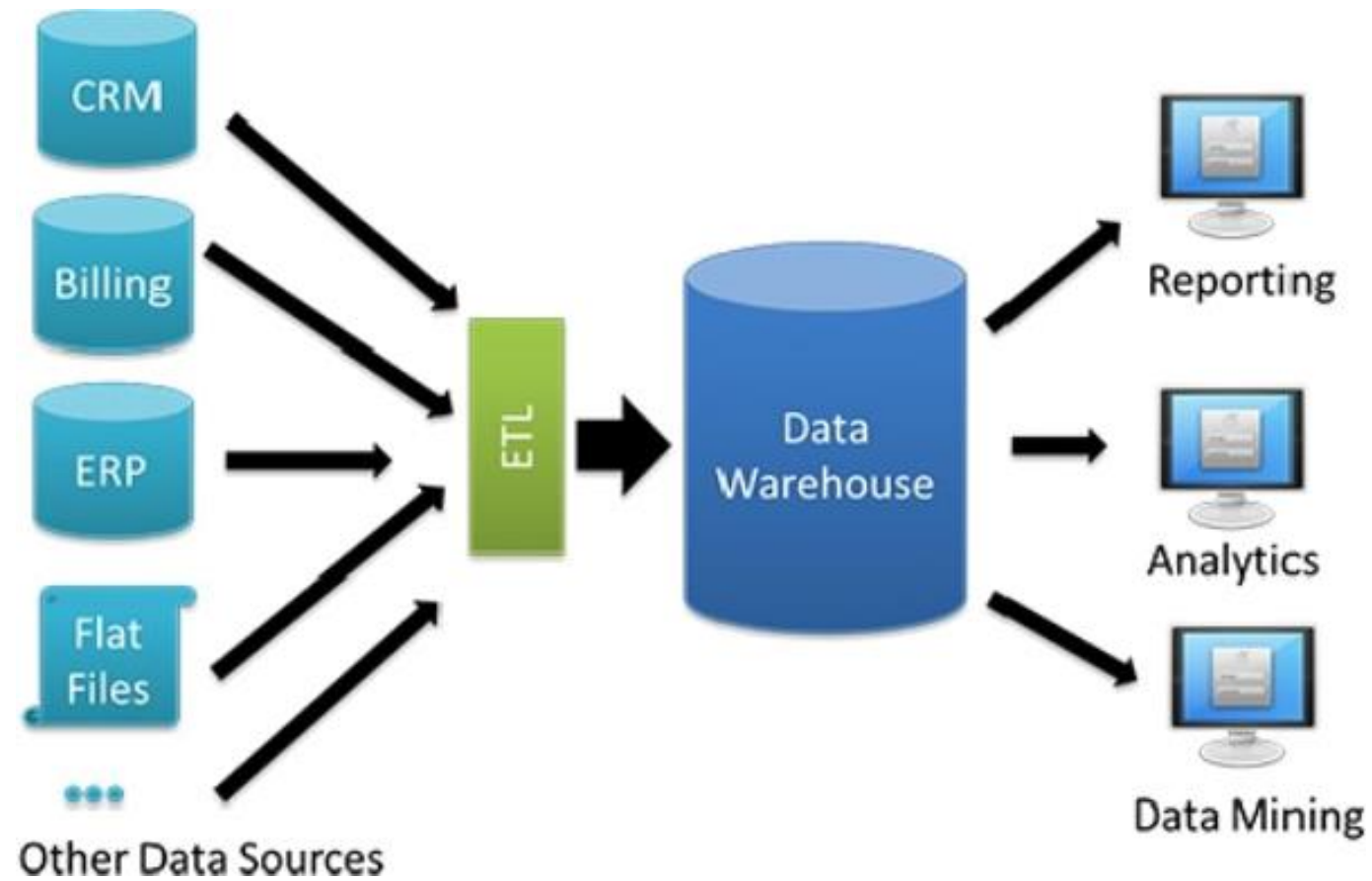


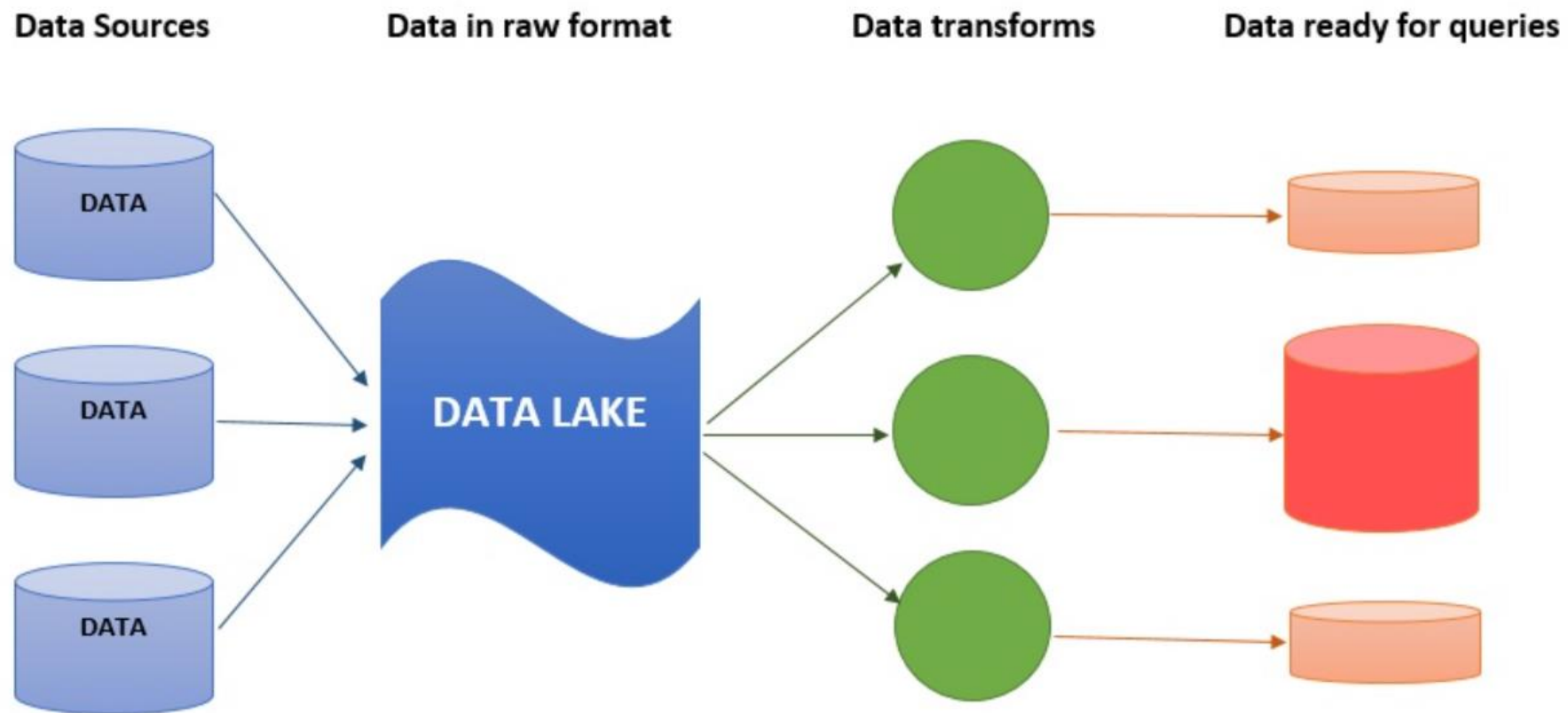
<https://www.menti.com/mvazch9s9m>

# Hoe zou je de data in een SQL-database definiëren?

- ▣ Gestructureerde data
- ▣ Relationele data-structuren
  
- ▣ Niet-gestructureerde data moeilijker bij te houden
  - Tekstbestanden
  - Beelden
  - Audio



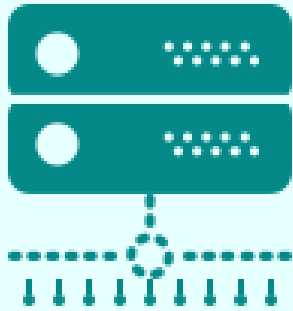




# DATA WAREHOUSE

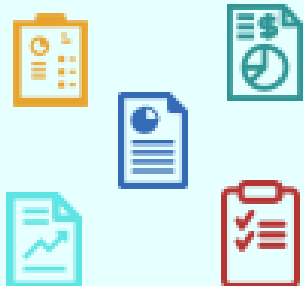
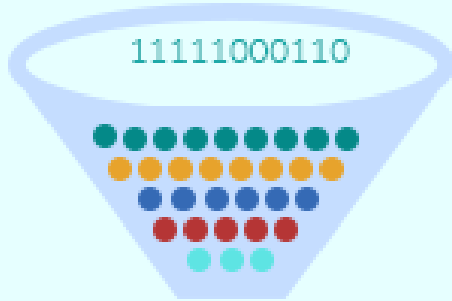
VS

# DATA LAKE

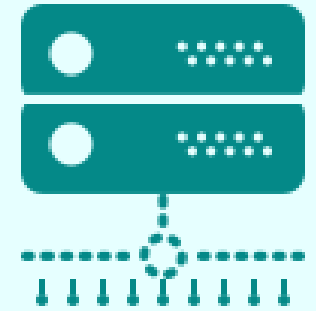


1110001101110  
011011000110  
11111000110

Data is processed and organized into a single schema before being put into the warehouse

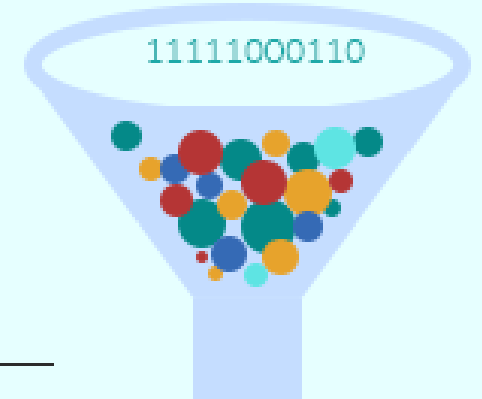


The analysis is done on the cleansed data in the warehouse



1110001101110  
011011000110  
11111000110

Raw and unstructured data goes into a data lake



Data is selected and organized as and when needed





DATA WAREHOUSE	vs.	DATA LAKE
structured, processed	DATA	structured / semi-structured / unstructured, raw
schema-on-write	PROCESSING	schema-on-read
expensive for large data volumes	STORAGE	designed for low-cost storage
less agile, fixed configuration	AGILITY	highly agile, configure and reconfigure as needed
mature	SECURITY	maturing
business professionals	USERS	data scientists et. al.

# Hadoop vs SQL?



# Hadoop Vs SQL

## #1. Data Size

Hadoop



Petabytes.

Traditional SQL



Gigabytes.

## #2. Access

Hadoop



Batch

Traditional SQL



Interactive  
& Batch.

## #3. Updates

Hadoop



Write  
once, read Multiple times.

Traditional SQL



Read  
and Write – Multiple times.

## #4. Structure

Hadoop



Dynamic  
Schema.

Traditional SQL



Static  
Schema.

## #5. Integrity

Hadoop



Low

Traditional SQL



High

## #6. Scaling

Hadoop



Linear.

Traditional SQL



Non Linear.

# Hadoop VS SQL

## Comparison Chart

Hadoop	SQL
Hadoop is an open-source distributed processing software framework for storing and processing massive influx of data across clusters.	SQL has been the ubiquitous tool to access, manipulate and store data in a database.
Hadoop is designed to work with any data type, structured, semi-structured or unstructured.	SQL can only process structured data and cannot be used for unstructured data.
Hadoop supports batch processing of data.	SQL supports real-time data processing.
Hadoop does not have the same end-user capabilities and ecosystem that SQL has.	SQL handles enforcing data quality and consistency much better than Hadoop
Hadoop is ideal for processing and storing massive amounts of data.	SQL is best for processing less amounts of data and has problems dealing with large volumes of data.

## Hoe zou SQL op Hadoop eruit zien?



<https://www.menti.com/mvazch9s9m>



**Welke software systemen kunnen hiervoor gebruikt worden?**

## Welke software systemen kunnen hiervoor gebruikt worden?

- ▣ Sqoop – import SQL data als text file (of een aantal andere formaten)
- ▣ Interessant artikel: <https://www.integrate.io/blog/integrating-relational-databases-with-apache-hadoop/>

## Welke software systemen kunnen hiervoor gebruikt worden?

- ▣ Sqoop – import SQL data als text file (of een aantal andere formaten)
- ▣ Hive – Query data on hdfs with SQL
  - ▣ Data stored in plain text (csv/tsv, ... )



# Tools voor datawarehouses on Hadoop cluster

## ■ Sqoop

- ▬ Import de resultaten van een SQL-query in een database in Hadoop
- ▬ Kan ook terug exporteren
- ▬ Resultaten op een Hadoop cluster kunnen gequeryed worden met Hive

# Tools voor datawarehouses on Hadoop cluster

## ▣ Sqoop

## ▣ Hive

- HQL (SQL-like met map-reduce, spark, ...)
- Batch processing, niet interactief
- Metadata bewaart in relationele database
- Schema voor datastructuur moet bijgehouden worden

## Tools voor datawarehouses on Hadoop cluster

- ▣ Sqoop (end of life)
- ▣ Hive
- ▣ Apache Spark Sql
  - ▬ Helpt om structured data te bevrageen/querying
  - ▬ DataFrames
  - ▬ Objecten in Memory bijgehouden
  - ▬ Resultaten kunnen bewaard worden in plain-texts/externe sql-databases



# Hive



## Hive

- ▣ Datawarehouse framework en ETL tool
  - ▣ Ontwikkeld door Facebook, later ook door Netflix
  - ▣ Kan ook gebruikt worden bij Amazon Web Services
  - ▣ SQL queries uitgevoerd via Map-reduce of Spark jobs
- 
- ▣ <https://www.tutorialspoint.com/hive/index.htm>

## Hive - Kenmerken

- ▣ Databases en tabellen eerst gemaakt, data wordt hierin ingeladen
- ▣ Gebruikt voor gestructureerde data
  - ▬ Optimalisatie en usability features zoals UDF (user defined functions)
- ▣ Verbergt complexiteit van MapReduce Programming
  - ▬ Door SQL-like dialect **HQL**, queries
- ▣ Maakt gebruik van HDFS partities om tabellen te splitsen voor optimalisatie
- ▣ Metastore voor database schema
  - ▬ Opgeslagen in relationele database
    - Derby voor single user
    - MySQL voor multi user

## Hive - Kenmerken

- ▣ Vooral interactie mogelijk via Command Line
  - Python packages om queries te sturen bestaan
- ▣ Ondersteunde fileformats
  - Textfile
  - Sequence file
  - ORC (Optimized Row Columnar)
  - RCFILE (Record Columnar File).



## Verschillen met traditionele databases

- ▣ Traditionele databases: Schema on Read and Write
  - ▬ Maakt het mogelijk om aanpassingen te doen (insertions, updates, ...)
- ▣ Hive maakt gebruik van Schema on Read
  - ▬ Omdat data op verschillende nodes bewaard wordt
  - ▬ Maakt updates onmogelijk
    - Nieuwere versies hebben wel deze functionaliteit





## Verschillen met MapReduce

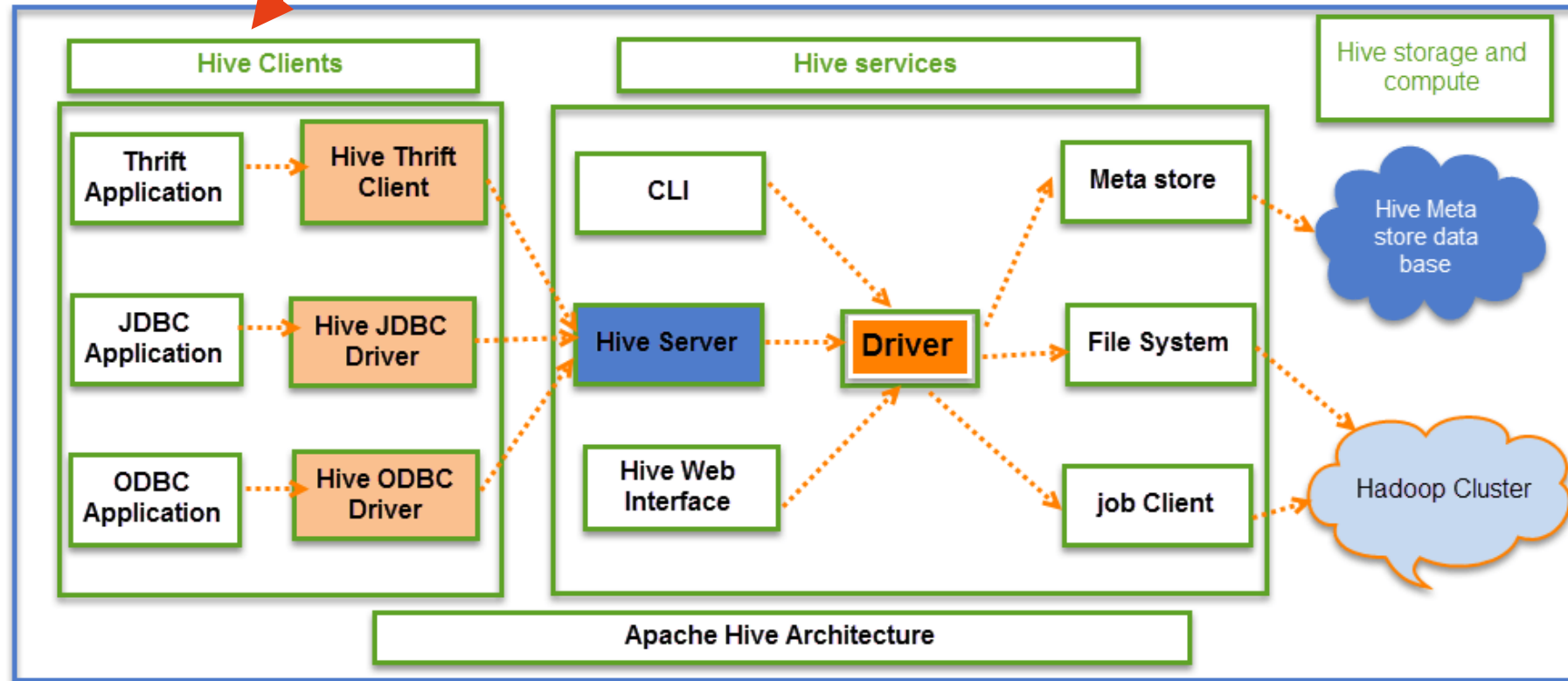
Feature	Hive	Map Reduce
Language	It Supports SQL like query language for interaction and for Data modeling	<ul style="list-style-type: none"><li>• It compiles language with two main tasks present in it. One is map task, and another one is a reducer.</li><li>• We can define these task using Java or Python</li></ul>
Level of abstraction	Higher level of Abstraction on top of HDFS	Lower level of abstraction
Efficiency in Code	Comparatively lesser than Map reduce	Provides High efficiency
Extent of code	Less number of lines code required for execution	More number of lines of codes to be defined
Type of Development work required	Less Development work required	More development work needed



# Hive Architectuur

## Hive Clients

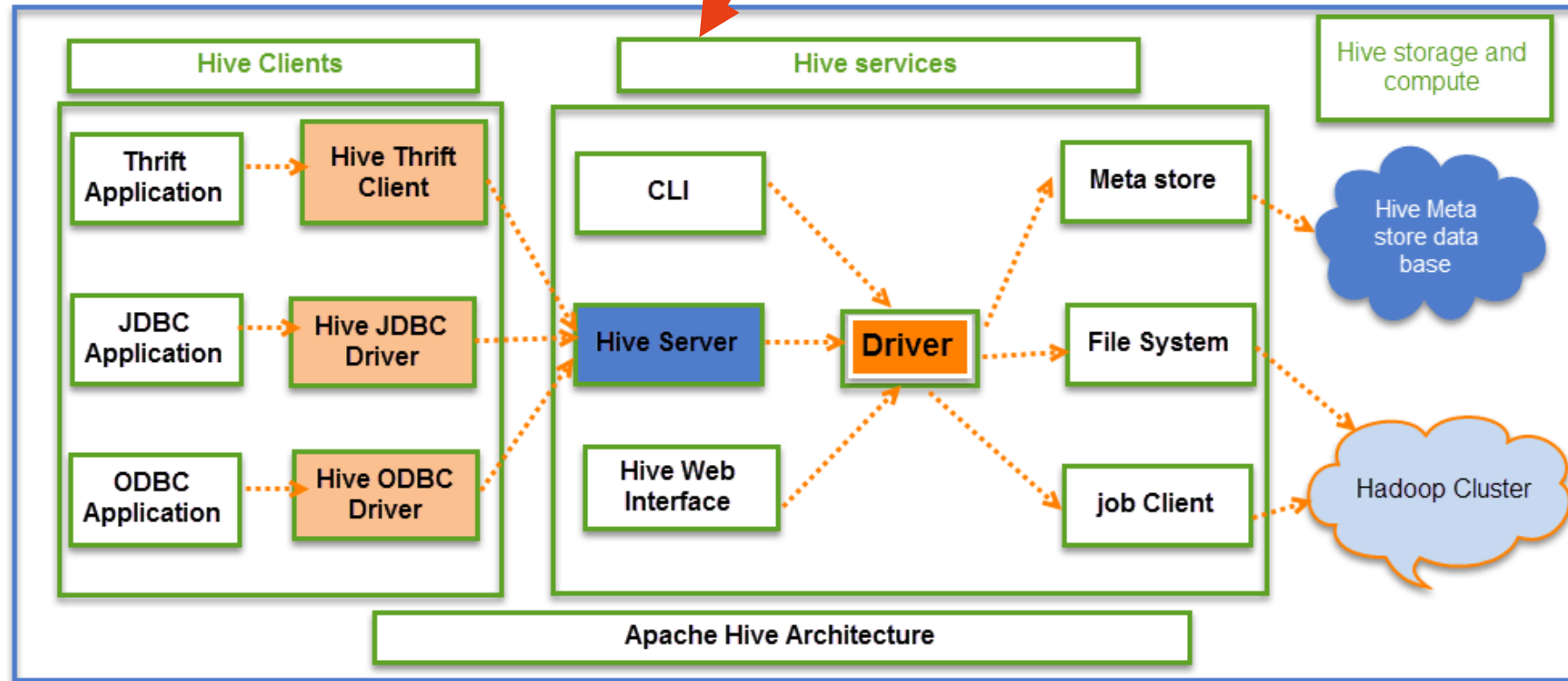
- Drivers voor communicatie met andere soorten applicaties



# Hive Architectuur

## Hive Services

- Elke uit te voeren query wordt hierdoor gecommuniceerd
- CLI is de interface voor Data Definition Language
- Driver verwerkt alle requests en communiceert met de metastore



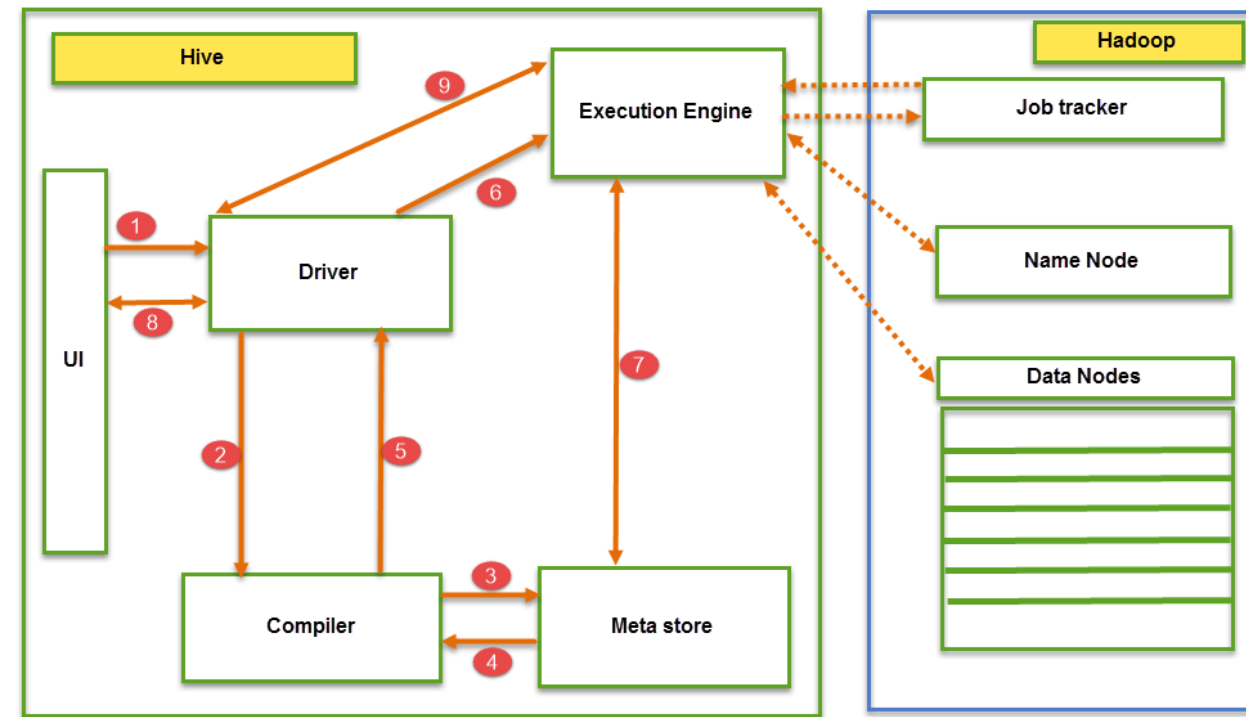
# Componenten

- ▣ Metastore:
  - ▬ Schema en locatie van alle tabellen
  - ▬ Opgeslagen in rdbms format
- ▣ Driver: create sessions, check life-cycle, ...
- ▣ Compiler: Zet query om naar stappenplan map-reduce
- ▣ Optimizer: Optimaliseer het stappenplan
- ▣ Executor: Voert het stappenplan uit
- ▣ Command-Line interface

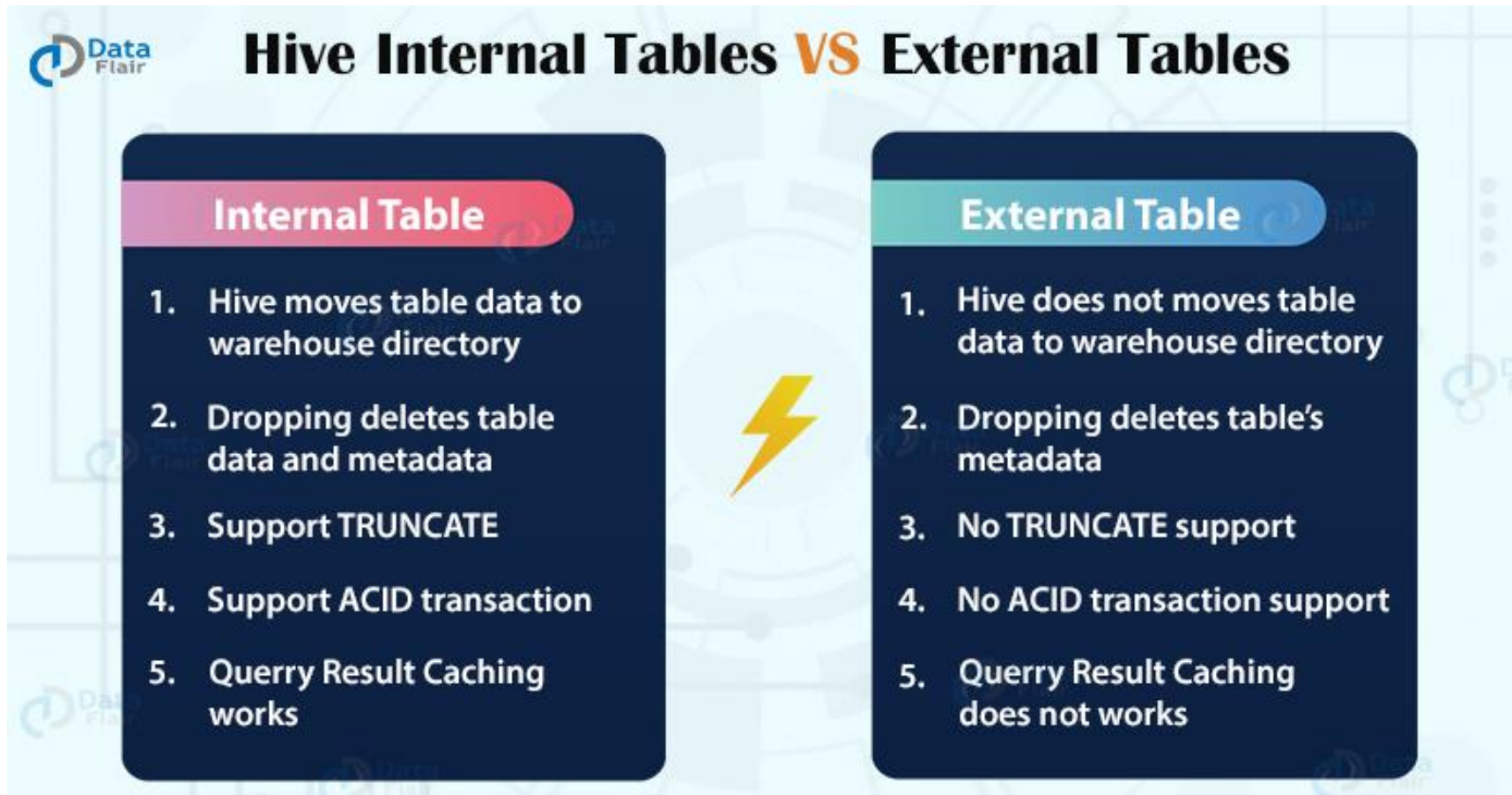


## Hive – workflow

- 1) Send Query from UI
- 2) Driver vraagt een plan en metadata aan de compiler
- 3) Compiler maakt plan vraagt metadata op
- 4) Metadata teruggestuurt naar compiler
- 5) Compiler bouwt stappenplan en stuurt het naar de driver
- 6) Driver stuurt plan naar Execution Engine
- 7) Execution Engine is brug tussen Hadoop en Hive
  - Communiceert met name- en datanodes
  - Vraag indien nodig extra data op of bewaart extra informatie in de metastore (bvb in het geval van een create table)
- 8) Fetch results from driver
- 9) Sending Results to execution engine
- 10) Stuurt het terug naar Driver en UI



# Internal vs External Tables



# Internal vs External Tables: Wanneer gebruiken

## ■ Internal

- ▬ Tijdelijke tabellen
- ▬ Hive moet de lifecycle van de data beheren
- ▬ Data moet verwijderd zijn na een drop

## ■ External

- ▬ Tabel niet aanmaken op basis van bestaande data
- ▬ Data moet buiten hive beschikbaar zijn
- ▬ Data behouden na Drop
- ▬ Data moet niet beheerd worden door Hive

# Partitioning

- Bij het aanmaken van een table kan je ook een kolom kiezen waarop gepartitioned wordt.
  - Alle elementen met dezelfde waarde in deze kolom worden bewaard in een aparte file
    - Versneld het opzoeken
    - Maakt het mogelijk om niet alle data te moeten laden





## Bucketing

- Een tweede-niveau partitioning om de data in nog gedetailleerder onder te verdelen in groepen/aparte files
  - ▬ Op basis van 1 of meerdere kolommen
  - ▬ Resultaat wordt gehashed
  - ▬ Deze hash bepaald in welke bucket het komt
- Niet standaard enabled
  - ▬ `set.hive.enforce.bucketing=true;` in `hive-site.xml`





## Oefening

- ▣ Ga naar de notebook en oefen het gebruik van Hive in

