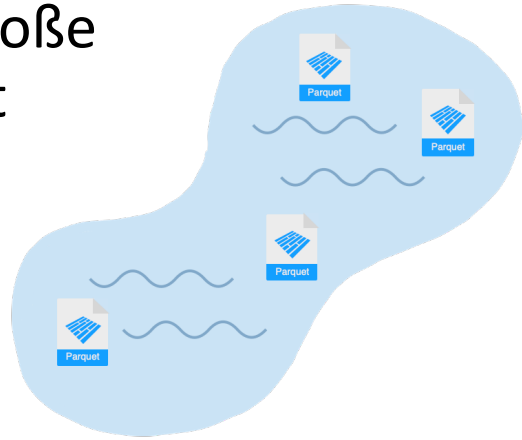
Parquet

Ein Dateiformat

Parquet

- Dateiformat zur kompakten Speicherung von großen Datenmengen
- Wird sehr häufig im Cloud-Kontext sowie in Data Lakes eingesetzt
  - Data Lakes sind zentrale Repositories / Server auf denen große Datenmengen mit unterschiedlichen Formaten gespeichert werden.
- Parquet ist ein open-source Projekt von Apache



Parquet vs. CSV

studenten.csv

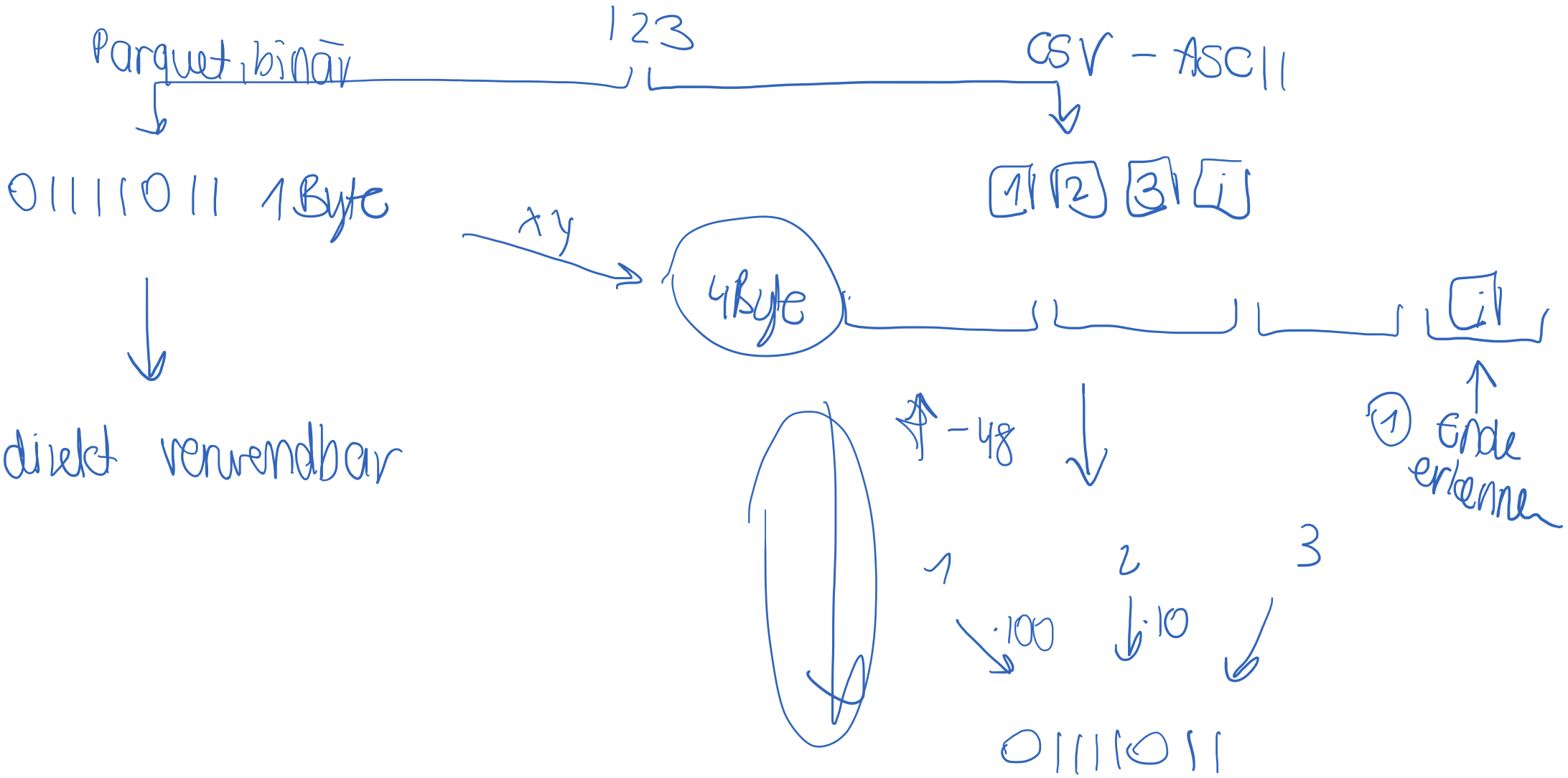
```
24002|Xenokrates|18
25403|Jonas|12
26120|Fichte|10
26830|Aristoxenos|8
27550|Schopenhauer|6
28106|Carnap|3
29120|Theophrastos|2
29555|Feuerbach|2
```

Parquet

- Parquet-Dateien sind im Binär-Format geschrieben und können nicht mit einem Text-Editor gelesen werden, aber das Binär-Format ist platzsparender
- In Parquet Dateien, werden die Daten zuerst horizontal in Blöcke und dann Spalten-Weise auf Seiten gespeichert
- Der Metadata-Footer am Ende der Parquet Datei enthält Informationen über das Schema und die Datentypen
- In Parquet können auch verschachtelte / “nested“ Datentypen wie Arrays oder JSON-Objekte gespeichert werden

CSV

- CSV-Dateien sind „human-readable“ und lassen sich dadurch händisch mit einem Text-Editor bearbeiten
- In CSV-Dateien werden die Daten Reihen-Weise geschrieben
- Das verwendete Trennzeichen sowie das Schema ist nicht in der CSV-Datei gespeichert. Wenn ein Programm CSV-Dateien lesen möchte, muss der User diese Informationen dem Programm mitteilen
- CSV Dateien kann nur Daten in 1. Normalform speichern


Parquet, binär
123
CSV - ASCII
01111011 1 Byte
1 2 3 i
direkt verwendbar
x y
4 Byte
i
-48
1 Ende erkennen
1
2
3
·100
·10
01111011

Parquet vs. CSV

Parquet

- Parquet-Dateien sind im Binär-Format geschrieben und können nicht mit einem Text-Editor gelesen werden, aber das Binär-Format ist platzsparender
- In Parquet Dateien, werden die Daten zuerst horizontal in Blöcke geteilt (sog. „**Row Groups**“) und dann Spalten-Weise (sog. “**Column Chunks**“) auf Seiten (sog. „**Pages**“) gespeichert

Matrnr	Name	Semester
24002	Xenokrates	18
25403	Jonas	12
26120	Fichte	10
26830	Aristoxenos	8
27550	Schopenhauer	6
28106	Carnap	3
29120	Theophrastos	2
29555	Feuerbach	2

Parquet vs. CSV

Parquet

- Parquet-Dateien sind im Binär-Format geschrieben und können nicht mit einem Text-Editor gelesen werden, aber das Binär-Format ist platzsparender
- In Parquet Dateien, werden die Daten zuerst horizontal in Blöcke geteilt (sog. „**Row Groups**“) und dann Spalten-Weise (sog. “**Column Chunks**“) auf Seiten (sog. „**Pages**“) gespeichert

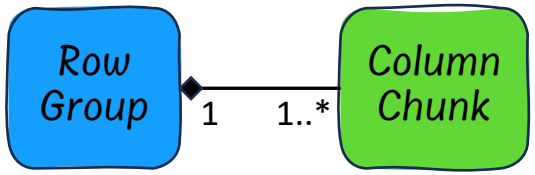
Row Group

Matrnr	Name	Semester
24002	Xenokrates	18
25403	Jonas	12
26120	Fichte	10
26830	Aristoxenos	8
27550	Schopenhauer	6
28106	Carnap	3
29120	Theophrastos	2
29555	Feuerbach	2

Parquet vs. CSV

Parquet

- Parquet-Dateien sind im Binär-Format geschrieben und können nicht mit einem Text-Editor gelesen werden, aber das Binär-Format ist platzsparender
- In Parquet Dateien, werden die Daten zuerst horizontal in Blöcke geteilt (sog. „**Row Groups**“) und dann Spalten-Weise (sog. “**Column Chunks**“) auf Seiten (sog. „**Pages**“) gespeichert

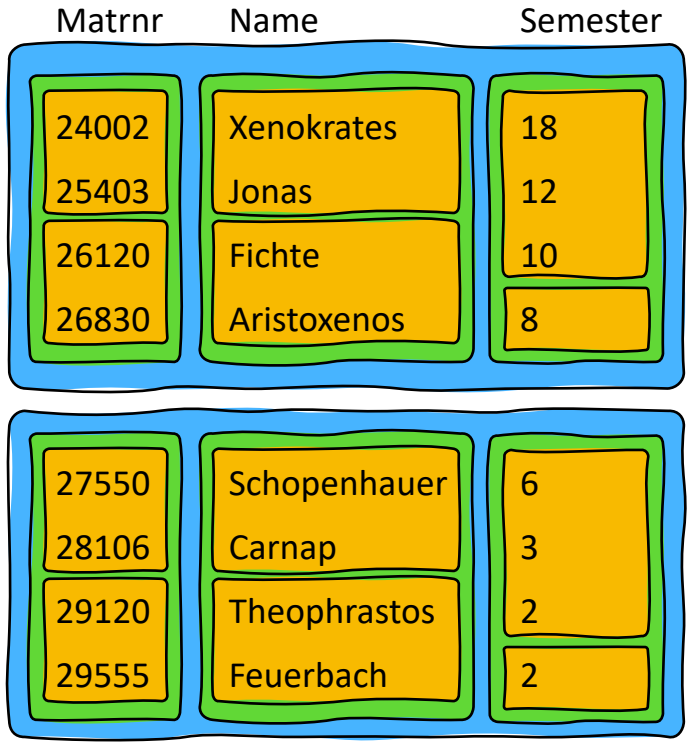


Matrnr	Name	Semester
24002	Xenokrates	18
25403	Jonas	12
26120	Fichte	10
26830	Aristoxenos	8
27550	Schopenhauer	6
28106	Carnap	3
29120	Theophrastos	2
29555	Feuerbach	2

Parquet vs. CSV

Parquet

- Parquet-Dateien sind im Binär-Format geschrieben und können nicht mit einem Text-Editor gelesen werden, aber das Binär-Format ist platzsparender
- In Parquet Dateien, werden die Daten zuerst horizontal in Blöcke geteilt (sog. „**Row Groups**“) und dann Spalten-Weise (sog. “**Column Chunks**“) auf Seiten (sog. „**Pages**“) gespeichert



Parquet vs. CSV

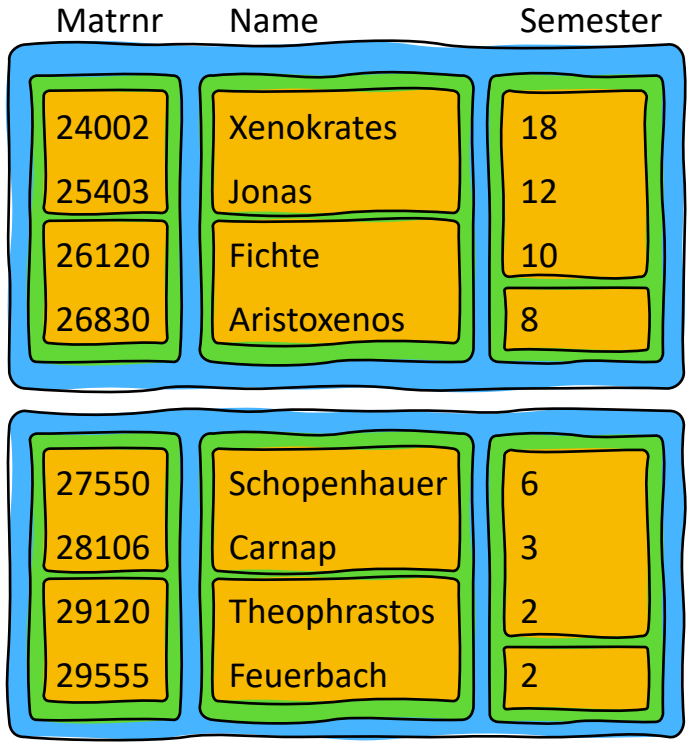
Parquet

- Da die Daten Spalten-weise gespeichert werden, können die Daten nochmals platzsparender gespeichert werden
- Auf Seiten-Ebene werden die Daten dann noch zusätzlich verkleinert durch encoding (z. B. dictionary encoding) und compression Schemas (z. B. Snappy compression)

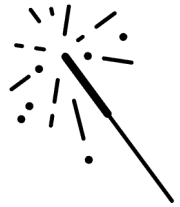
TPC-H Datengröße

	CSV	Parquet
lineitem	7.2 GB	2.1 GB

3.4×



Parquet in Spark



- Spark Dataframes können auch aus Parquet-Dateien generiert werden
- Da das Schema in den Parquet-Dateien gespeichert wird, muss das Schema nicht erneut definiert werden

```
val studenten =
spark.read.format("csv").schema(StructType(
    List(
      StructField("matrnr", IntegerType, false),
      StructField("name", StringType, false),
      StructField("semester", IntegerType, false)
    )
  )).option("delimiter", "|").load("studenten.csv")
```

```
val studenten = spark.read.format("parquet")
                     .load("studenten.parquet")
```

Parquet in Spark

Parquet vs. CSV in Spark

- Parquet Dateien sind nicht nur platzsparender als CSV Dateien, sie können durch die binäre Speicherung auch viel effizienter verarbeitet werden