Hive Optimization

File Formats and Compression

Record Columnar File (RCFile): Design Goals

[Facebook] 2011 - RCFile: A Fast and Space-efficient Data Placement Structure in MapReduce-based Warehouse Systems

Record Columnar File (RCFile): Design Goals

1. **fast data loading**

[Facebook] 2011 - RCFile: A Fast and Space-efficient Data Placement Structure in MapReduce-based Warehouse Systems

Record Columnar File (RCFile): Design Goals

1. fast data loading
2. **fast query processing**

[Facebook] 2011 - RCFile: A Fast and Space-efficient Data Placement Structure in MapReduce-based Warehouse Systems

Record Columnar File (RCFile): Design Goals

1. fast data loading
2. fast query processing
3. **highly efficient storage space utilization**

[Facebook] 2011 - RCFile: A Fast and Space-efficient Data Placement Structure in MapReduce-based Warehouse Systems

Record Columnar File (RCFile): Design Goals

1. fast data loading
2. fast query processing
3. highly efficient storage space utilization
4. **strong adaptivity to highly dynamic workload patterns**

[Facebook] 2011 - RCFile: A Fast and Space-efficient Data Placement Structure in MapReduce-based Warehouse Systems

Record Columnar File (RCFile): Design Goals

Logical table

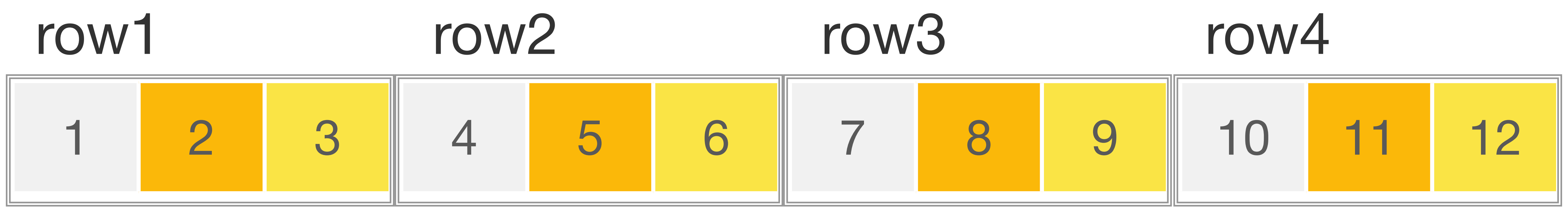
	col1	col2	col3
row1	1	2	3
row2	4	5	6
row3	7	8	9
row4	10	11	12

Record Columnar File (RCFile): Design Goals

Logical table

	col1	col2	col3
row1	1	2	3
row2	4	5	6
row3	7	8	9
row4	10	11	12

Row-oriented layout



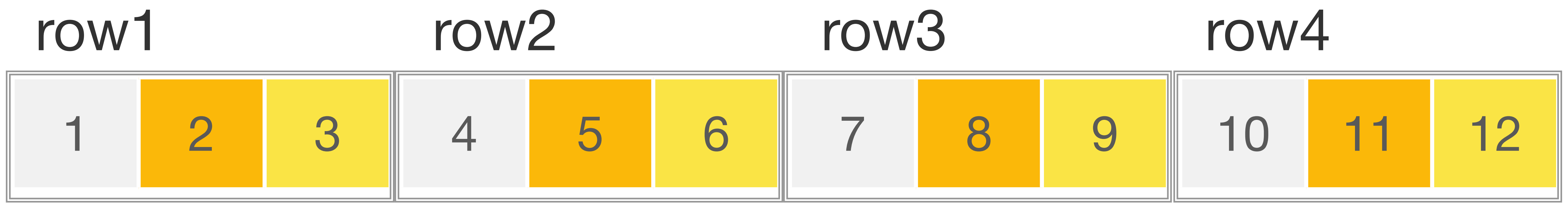
row1			row2			row3			row4		
1	2	3	4	5	6	7	8	9	10	11	12

Record Columnar File (RCFile): Design Goals

Logical table

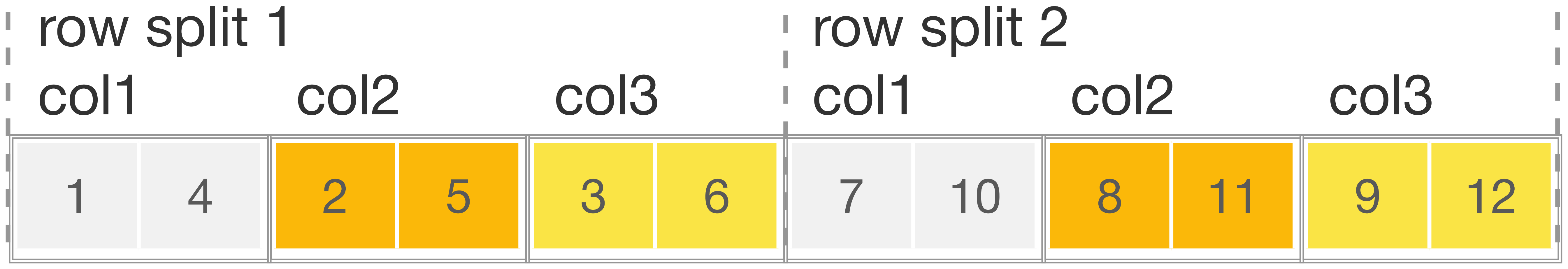
	col1	col2	col3
row1	1	2	3
row2	4	5	6
row3	7	8	9
row4	10	11	12

Row-oriented layout



row1			row2			row3			row4		
1	2	3	4	5	6	7	8	9	10	11	12

Column-oriented layout (RCFile)



row split 1						row split 2					
col1		col2		col3		col1		col2		col3	
1	4	2	5	3	6	7	10	8	11	9	12

Record Columnar File (RCFile): Design Goals

Logical table

	col1	col2	col3
row1	1	2	3
row2	4	5	6
row3	7	8	9
row4	10	11	12

Row-oriented layout

row1			row2			row3			row4		
1	2	3	4	5	6	7	8	9	10	11	12

Column-oriented layout (RCFile)

row split 1						row split 2					
col1		col2		col3		col1		col2		col3	
1	4	2	5	3	6	7	10	8	11	9	12

date	user	order
2017-05-19 17:53	John	100
2017-05-19 17:59	Jane	200
2017-05-19 18:02	Alex	50
2017-05-20 10:27	Emeli	350

Record Columnar File (RCFile): Design Goals

Logical table

	col1	col2	col3
row1	1	2	3
row2	4	5	6
row3	7	8	9
row4	10	11	12

Row-oriented layout

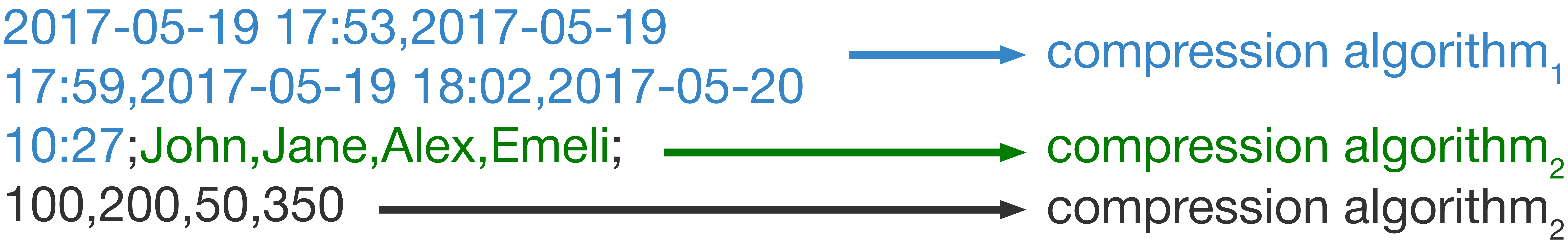
row1			row2			row3			row4		
1	2	3	4	5	6	7	8	9	10	11	12

Column-oriented layout (RCFile)

row split 1						row split 2					
col1		col2		col3		col1		col2		col3	
1	4	2	5	3	6	7	10	8	11	9	12

date	user	order
2017-05-19 17:53	John	100
2017-05-19 17:59	Jane	200
2017-05-19 18:02	Alex	50
2017-05-20 10:27	Emeli	350

2017-05-19 17:53,2017-05-19 17:59,2017-05-19 18:02,2017-05-20 10:27;John,Jane,Alex,Emeli; 100,200,50,350

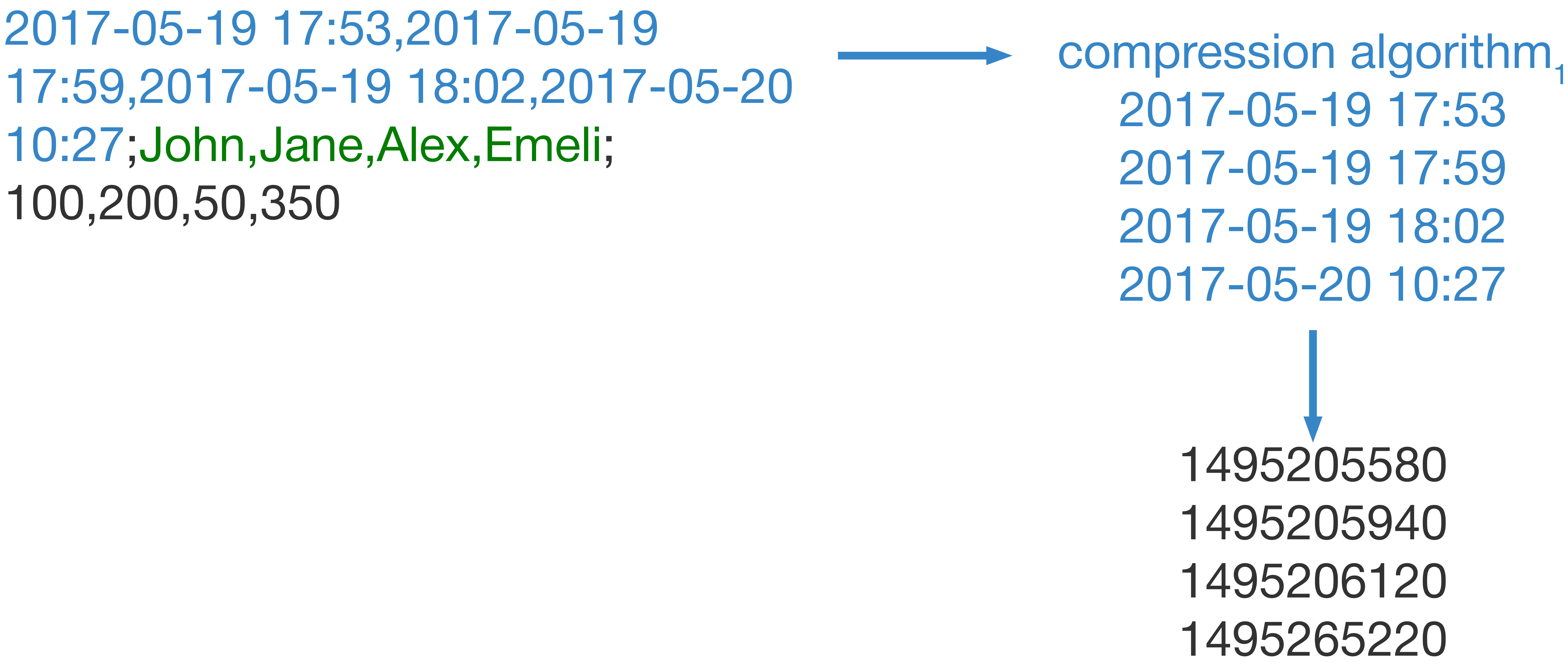


2017-05-19 17:53,2017-05-19
17:59,2017-05-19 18:02,2017-05-20
10:27;John,Jane,Alex,Emeli;
100,200,50,350

compression algorithm$_1$

compression algorithm$_2$

compression algorithm$_2$



2017-05-19 17:53,2017-05-19 17:59,2017-05-19 18:02,2017-05-20 10:27;John,Jane,Alex,Emeli;
100,200,50,350

compression algorithm$_1$

2017-05-19 17:53
2017-05-19 17:59
2017-05-19 18:02
2017-05-20 10:27

1495205580
1495205940
1495206120
1495265220

2017-05-19 17:53,2017-05-19 17:59,2017-05-19 18:02,2017-05-20 10:27;John,Jane,Alex,Emeli;
100,200,50,350

compression algorithm$_1$

2017-05-19 17:53
2017-05-19 17:59
2017-05-19 18:02
2017-05-20 10:27

1495205580
1495205940
1495206120
1495265220

2017-05-19 17:53,2017-05-19 17:59,2017-05-19 18:02,2017-05-20 10:27;John,Jane,Alex,Emeli;
100,200,50,350

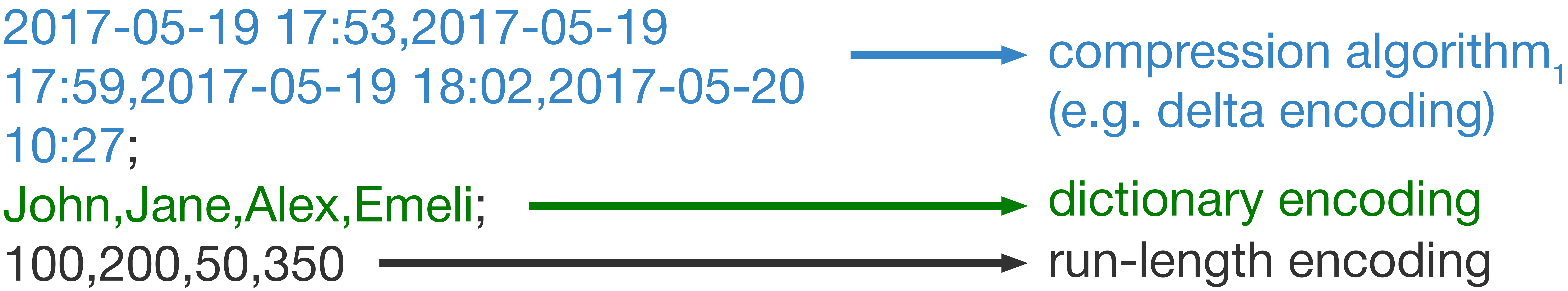
compression algorithm$_1$

2017-05-19 17:53
2017-05-19 17:59
2017-05-19 18:02
2017-05-20 10:27

1495205580
1495205940
1495206120
1495265220

δ-encoding

1495205580
+360
+180
+59100



2017-05-19 17:53,2017-05-19 17:59,2017-05-19 18:02,2017-05-20 10:27;

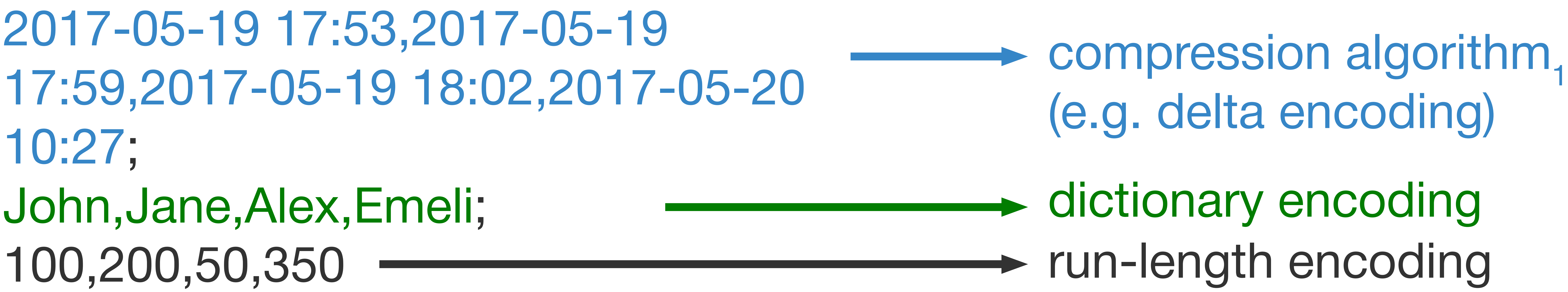
John,Jane,Alex,Emeli;

100,200,50,350

compression algorithm$_1$ (e.g. delta encoding)

dictionary encoding

run-length encoding



2017-05-19 17:53,2017-05-19 17:59,2017-05-19 18:02,2017-05-20 10:27; → compression algorithm[1] (e.g. delta encoding)

John,Jane,Alex,Emeli; → dictionary encoding

100,200,50,350 → run-length encoding

see: https://cwiki.apache.org/confluence/display/Hive/RCFile
see: https://en.wikipedia.org/wiki/Dictionary_coder
see: https://en.wikipedia.org/wiki/Run-length_encoding

Record Columnar File (RCFile): Design Goals

1. fast data loading
2. fast query processing
3. highly efficient storage space utilization
4. strong adaptivity to highly dynamic workload patterns

Record Columnar File (RCFile): Design Goals

1. fast data loading
2. ✓ fast query processing
3. highly efficient storage space utilization
4. strong adaptivity to highly dynamic workload patterns

Record Columnar File (RCFile): Design Goals

1. fast data loading

✓ 2. fast query processing

✓ 3. highly efficient storage space utilization

4. strong adaptivity to highly dynamic workload patterns

Record Columnar File (RCFile): Design Goals

± 1. fast data loading

✓ 2. fast query processing

✓ 3. highly efficient storage space utilization

4. strong adaptivity to highly dynamic workload patterns

Record Columnar File (RCFile): Design Goals

± 1. fast data loading

✓ 2. fast query processing

✓ 3. highly efficient storage space utilization

✗ 4. strong adaptivity to highly dynamic workload patterns

↓

✓ ORC = Optimised Row Columnar (file format)   Apache orc

Record Columnar File (RCFile): Design Goals

± 1. fast data loading

✓ 2. fast query processing

✓ 3. highly efficient storage space utilization

✗ 4. strong adaptivity to highly dynamic workload patterns

↓

✓ ORC = Optimised Row Columnar (file format) Apache orc

facebook + HORTONWORKS

Record Columnar File (RCFile): Design Goals

± 1. fast data loading

✓ 2. fast query processing

✓ 3. highly efficient storage space utilization

✗ 4. strong adaptivity to highly dynamic workload patterns

↓

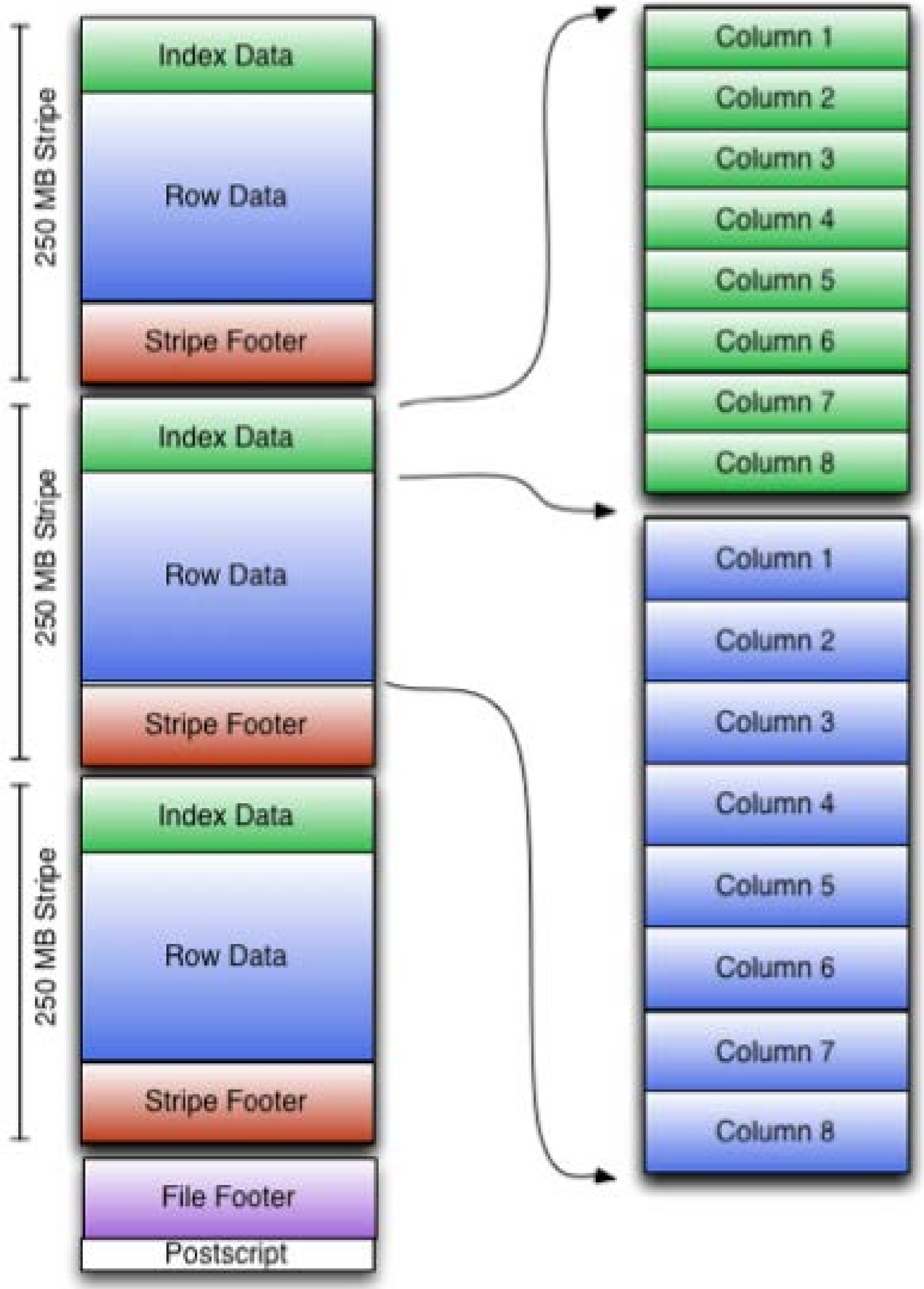
✓ ORC = Optimised Row Columnar (file format) Apache orc

facebook + HORTONWORKS

✓ Parquet (especially for nested structures) Parquet

twitter + cloudera

ORC

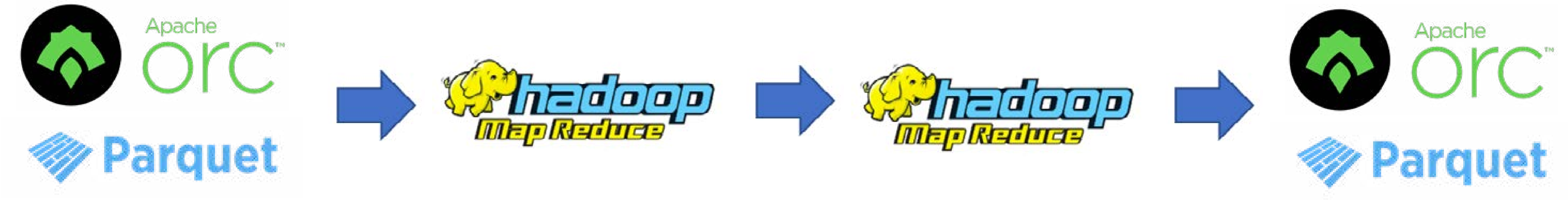


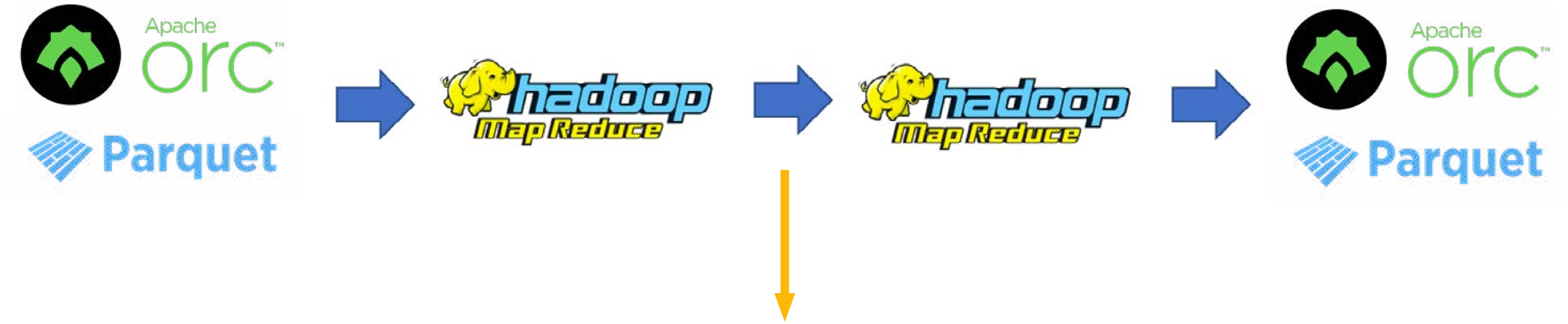
see: https://cwiki.apache.org/confluence/display/Hive/LanguageManual+ORC

ORC

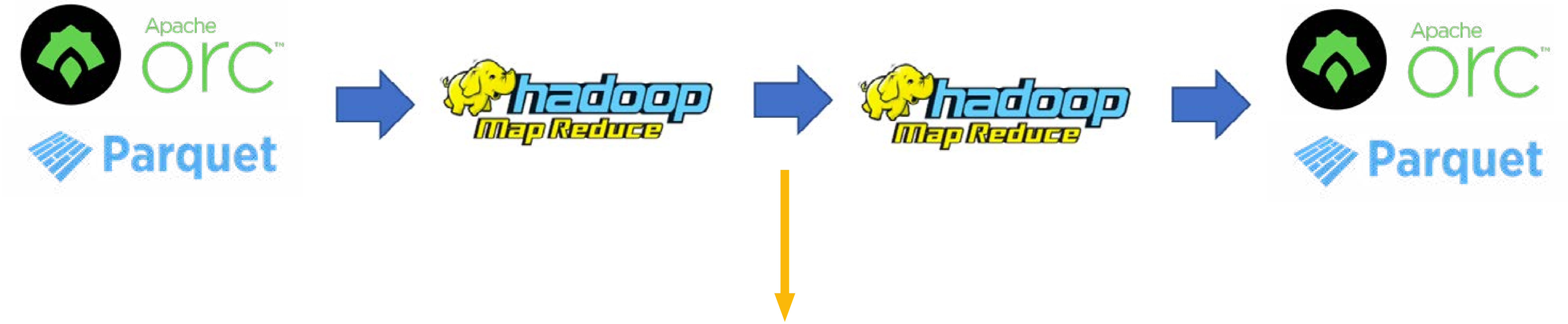
Key	Default	Notes
orc.compress	ZLIB	high level compression (one of NONE, ZLIB, SNAPPY)
orc.compress.size	262,144	number of bytes in each compression chunk
orc.stripe.size	67,108,864	number of bytes in each stripe

```
CREATE TABLE my_orc_table (
    …
)
STORED AS orc
    TBLPROPERTIES ("orc.compress"="NONE")
…;
```

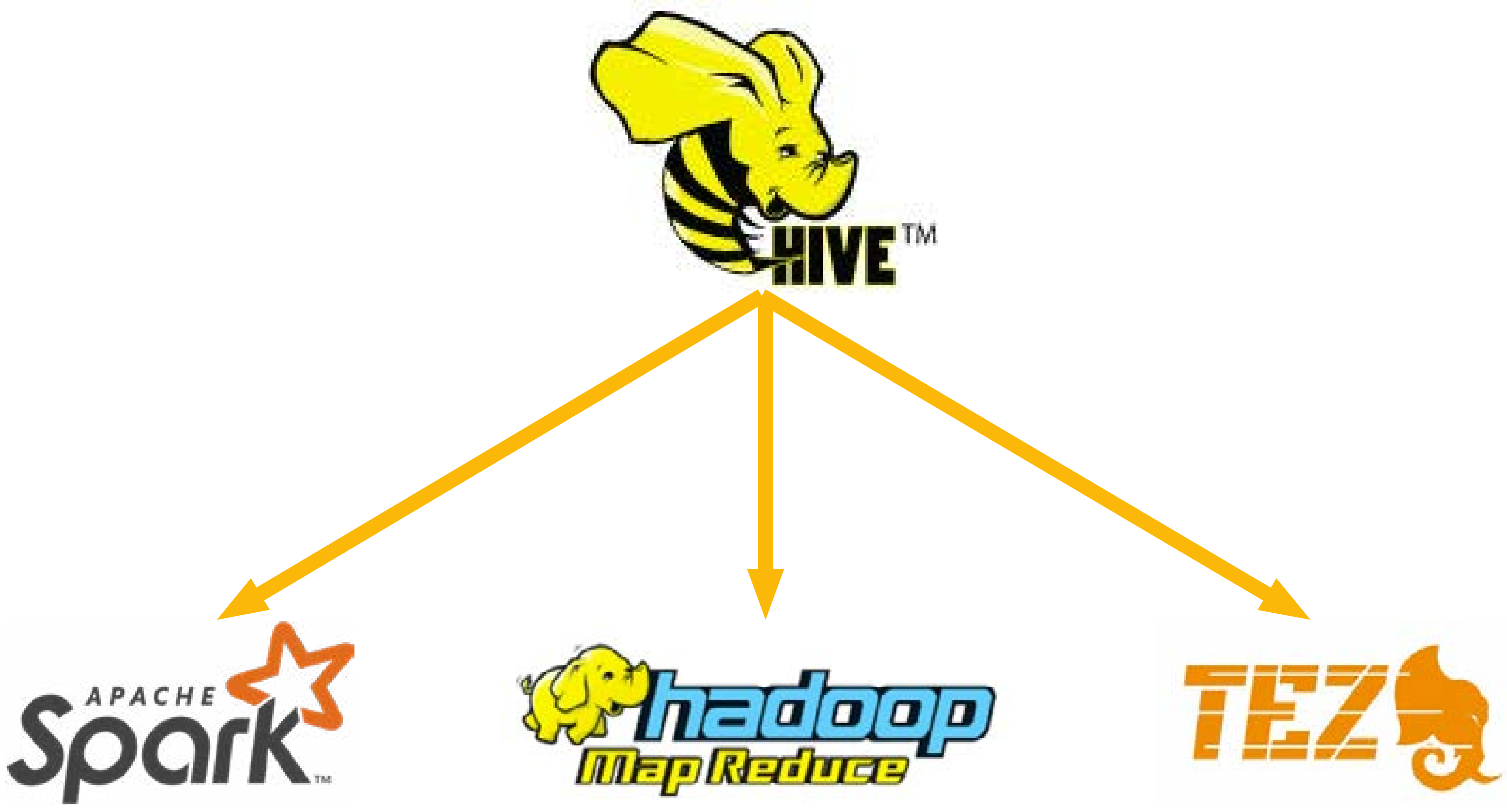

Apache orc™
Parquet
hadoop Map Reduce
hadoop Map Reduce
Apache orc™
Parquet



```
SET hive.exec.compress.intermediate=true;
SET mapreduce.map.output.compress=true;
SET mapreduce.map.output.compress.codec=…;
```



```
SET hive.exec.compress.intermediate=true;
SET mapreduce.map.output.compress=true;
SET mapreduce.map.output.compress.codec=…;
```



Summary

Summary

- You can **list 4** design goals for **row-column oriented** file formats for Big Data applications

Summary

- You can **list 4** design goals for **row-column oriented** file formats for Big Data applications
- You can **configure** Hive table metadata to use **ORC / Parquet** file format

Summary

- You can **list 4** design goals for **row-column oriented** file formats for Big Data applications
- You can **configure** Hive table metadata to use **ORC / Parquet** file format
- You can **list** supported Hive execution engines and explain how to **configure** intermediate compression for MapReduce engine

Summary

- You can **list 4** design goals for **row-column oriented** file formats for Big Data applications
- You can **configure** Hive table metadata to use **ORC / Parquet** file format
- You can **list** supported Hive execution engines and explain how to **configure** intermediate compression for MapReduce engine

`nice article: Scaling the Facebook data warehouse to 300 PB`