Parquet Files

Apache Parquet is a *columnar* file format originally developed by engineers at Cloudera and Twitter. Parquet was inspired by a project at Google called Dremel.

Columnar, or column-oriented, file formats organize data by column, rather than by row. This makes them more efficient when you need to process only one or a few columns. For example, see Figure 1 below. The rows (1, 2, 3) and columns (A, B, C) of the data are shown in a tabular structure on the left side. The images on the right side represent how this data could be stored in a row-oriented file format (top) and a column-oriented file format (bottom).

When the data is stored in a row-oriented format, the file organizes the data sequentially first by row (Row 1 consists of A1, B1, C1). When the data is stored like this, Hive and Impala must read each full row even if the query requires processing only one column.

When the data is stored in a column-oriented format, the file organizes the data sequentially first by column (Column A has values A1, A2, A3). When the data is stored like this, Hive and Impala can skip over the columns that are not part of the query, and quickly read only the values for the columns that are part of the query. This improves query performance, especially for tables with dozens or hundreds of columns.
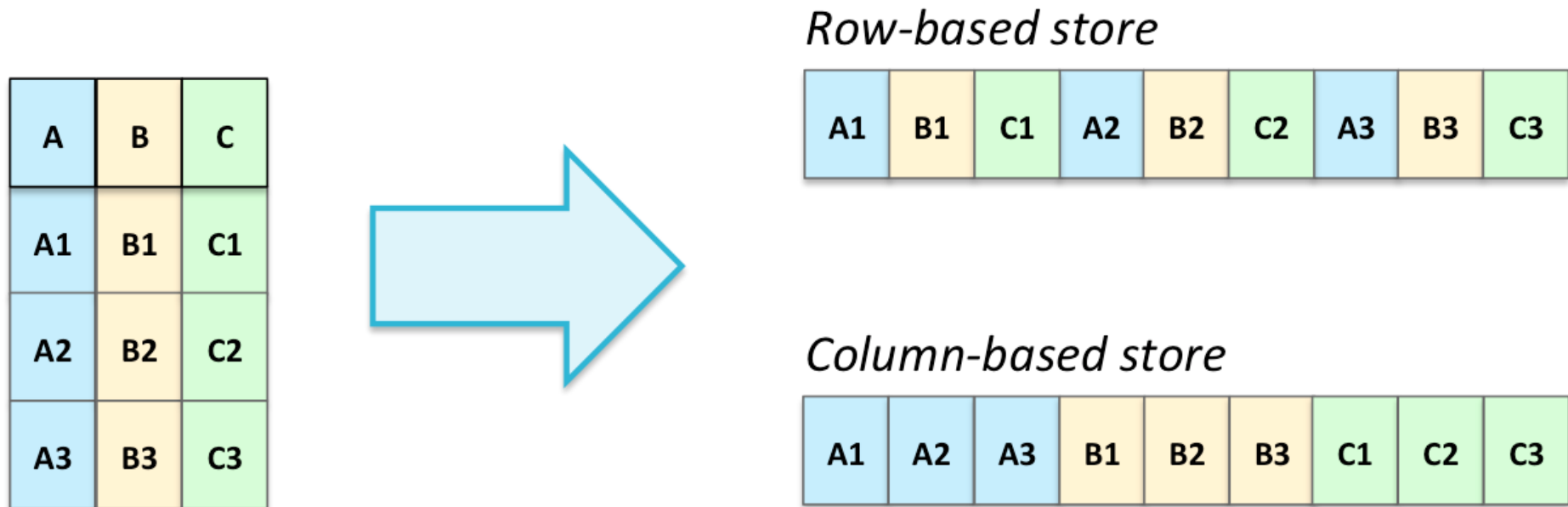
Figure 1

Parquet is widely supported by big data tools, including Hive and Impala, and it's designed to work across different programming languages. Like Avro, Parquet embeds a schema definition in the file, and it supports schema evolution.

Parquet uses advanced optimizations to store data more compactly and to speed up queries. It is most efficient when data is loaded all at once or in large batches; this enables Parquet to take advantage of repeated patterns in the data to store it more efficiently. (See Figure 2 below.)

id	name	city

1	Alice	Palo Alto
2	Bob	Sunnyvale
3	Bob	Palo Alto
4	Bob	Berkeley
5	Carol	Berkeley

Alice Bob (×3) Carol … Palo Alto

Figure 2

Parquet also uses *compression*. Compression encodes data in a way to take up less storage than an uncompressed file will, but there is a time cost when you read or write the file. That is, encoding for less storage means more time needed to compress (encode) the file before writing it, or to decompress (decode) the file after reading it.

Overall, Parquet offers excellent interoperability and excellent performance, making it a popular choice for columnar data storage.