Parquet file format — brief introduction

Wojciech Muła, 0x80.pl December 2021

Major features:

- Fixed data schema
- Columnar storage (by design)
- Ability to split multiple rows into "row groups" (file creator decision)
- Designed to parallel processing: at the level of row groups and columns
- Per-column & per-row group data compression
- Rich data types: numbers, strings, timestamps, bool, arrays, structs, dictionaries, unions, etc.
- Uses **Dremel** encoding to describe nested data structures

Algorithm backing the format

- **Dremel** invented by Google
- Assigns values only to `leaves` of schema tree
- ... inner nodes may be only set to `null`
- Allow to repeat any part of schema subtree
- ...but does not support arrays directly

Sample schema with five columns

```
customer
+- id          -- customer.id
+- location
|  + city      -- customer.location.city
|  + country   -- customer.location.country
|
+- name
   |
   + first     -- customer.name.first
   + last      -- customer.name.last
```

Column data — overview

Single column data consists **two** or **three** arrays of values

- Always present **definition levels** — which part of path is "defined"
- Optional **repetition levels** — at which part of schema we repeat values
- **Actual** data, present if there are not-null values in column
- Definition & repetition levels are sufficient to reconstruct arbitrary record structure

Column data — levels

- Definition and repetition levels are arrays of unsigned numbers
- Their sizes are `equal` to the number of rows
- Repetition level is present only if some part of path may repeats
- . . . this property is set in the data schema

Column data — definition levels

Let's assume path `customer.location.city`. Other possible paths are `customer.location` and `customer`.

row	definition level	value
#1	0	null
#2	3 <defined>	"London"
#3	1	null
#4	3 <defined>	"New York"
#5	2	null

Corresponding JSON

```
{"customer": null}
{"customer": {"location": {"city": "London"}}}
{"customer": {"location": null}}
{"customer": {"location": {"city": "New York"}}}
{"customer": {"location": {"city": null}}}
```

Column data — definition levels continued

- Definition level equals to "max definition level" means the values is present
- ... in the example we have only two values in the data array
- Definition level less than max says at which part of path we set `null`
- It's easy to do queries like "SELECT COUNT(*) ... WHERE column IS NOT NULL"

Column data — repetition levels

- similarly to definition level — decides which part of path repeats
- if value is defined it means "append" to the tree
- ... it doesn't apply to JSON, XML is better

The same definition levels, but different repetition levels

```
<customer>
    <location>
        <city>London</city>
        <city>New York</city>
    </location>
</customer>
```

```
<customer>
    <location>
        <city>London</city>
    </location>
    <location>
        <city>New York</city>
    </location>
</customer>
```

Parquet format — part 1

- Parquet uses the Dremel algorithm underneath
- It efficiently encodes definition and repetition levels (RLE, compression)
- ... but exposes them as plain arrays of `uint16`
- Parquet uses seven physical types: `boolean`, `int32`, `int64h` `float32`, `float64`, `variable-length bytes`, `fixed-length bytes`
- It may collect some statistics regarding the column, like: nulls count, distinct count, min value, max value
- Columnar data is usually compressed

Parquet format — part 2

- Physical types are mapped into logical types, like UTF-8 strings, timestamps, date time, decimal
- Logical types are also complex ones: structures, arrays, maps, unions, dictionaries
- Arrays can't be directly represented in Dremel algorithm
- ... as it lets repeat only "key-value" pairs
- ... plain arrays are done by adding artificial nodes to schema and then process them in a special way
- Unions and dictionaries are data types designed for reduce memory usage

Further reading

- Parquet format — a lot of details: data layout, used encodings, design considerations, etc.
- Dremel made simple with Parquet — detailed overview of interpretation definition and repetition levels
- Dremel: Interactive Analysis of Web-Scale Datasets – the original paper