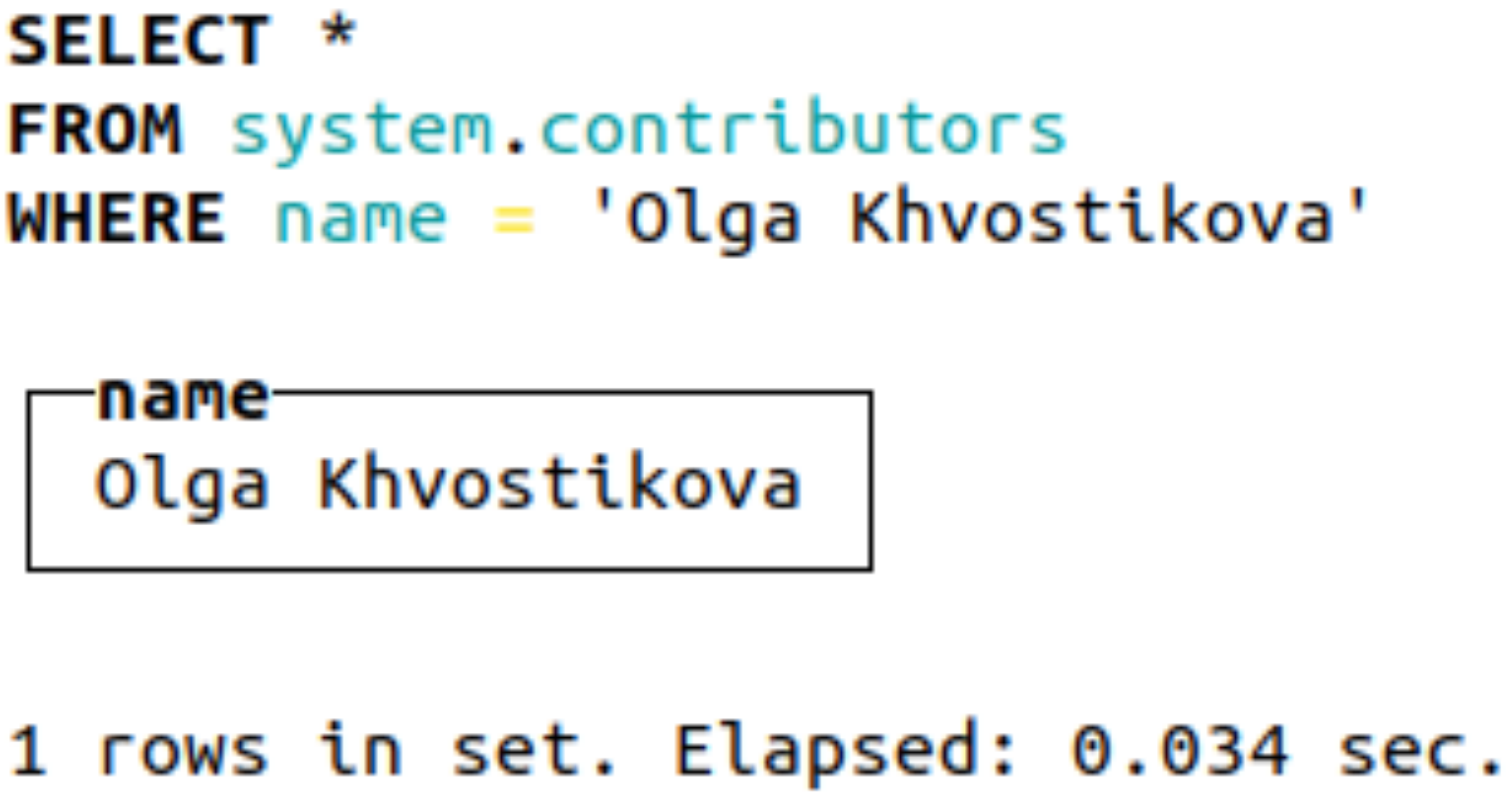
Yandex

Best way to introduce



```
SELECT *
FROM system.contributors
WHERE name = 'Olga Khvostikova'
```

```
┌─name─────────────┐
│ Olga Khvostikova │
└──────────────────┘

1 rows in set. Elapsed: 0.034 sec.
```

How many languages do you speak?

Three: Russian, English and SQL

ClickHouse

Using ClickHouse to process data stored in files or Hadoop

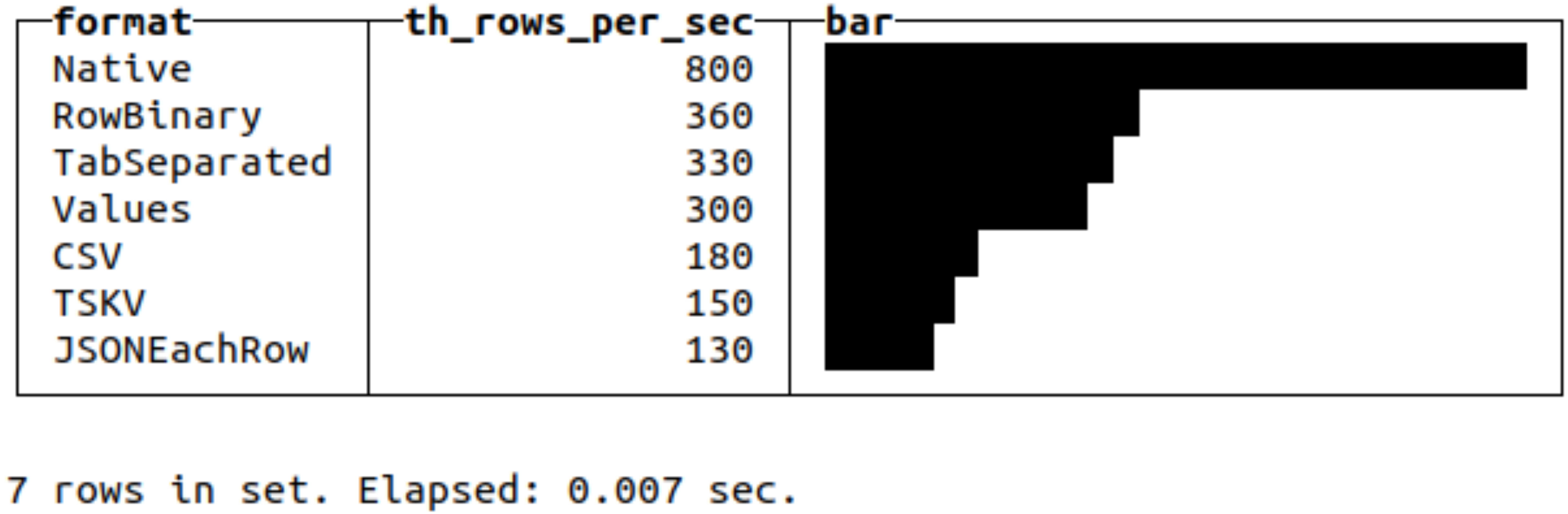
Olga Khvostikova, ClickHouse core developer

Table Engine Families

- MergeTree (subsequent background data processing)
- Log (write many small tables and read them later as a whole)
- Integration (communicating with other data storage systems)
- Special:
  - Distributed (distributed query processing on multiple servers)
  - Dictionary (external map)
  - File
  - ...

File as a source

Parsing performance



```
┌─format───────┬─th_rows_per_sec─┬─bar────────────────────────────┐
│ Native       │             800 │ ██████████████████████████████ │
│ RowBinary    │             360 │ █████████████▌                 │
│ TabSeparated │             330 │ ████████████▍                  │
│ Values       │             300 │ ███████████▎                   │
│ CSV          │             180 │ ██████▋                        │
│ TSKV         │             150 │ █████▋                         │
│ JSONEachRow  │             130 │ ████▊                          │
└──────────────┴─────────────────┴────────────────────────────────┘

7 rows in set. Elapsed: 0.007 sec.
```

The 'bar' function is the best data visualization tool

Also ClickHouse can parse

- Parquet
- Protobuf
- Cap'n Proto
- ORC (since 19.14!)
- Template (since 19.14!)

The clickhouse-local program

- Processing local files
- No deploy and server configuration
- Use the same code as ClickHouse server

SELECT count() FROM table

	clickhouse-	Spark with local file	Linux tools (ws)
real	32s 910ms	3m 8s 397ms	3s 564ms
user	29s 379ms		1s 507ms
sys	2s 773ms		2s 56ms

local file

single thread

no compression

TabSeparated

SELECT symbol, count() FROM table GROUP BY symbol order by symbol

	clickhouse-	Spark with local file	Linux tools
real	33s 384ms	3m 7s 962ms	Terminated
user	30s 934ms		too
sys	2s 392ms		slow

local file

single thread

no compression

TabSeparated

Coming soon

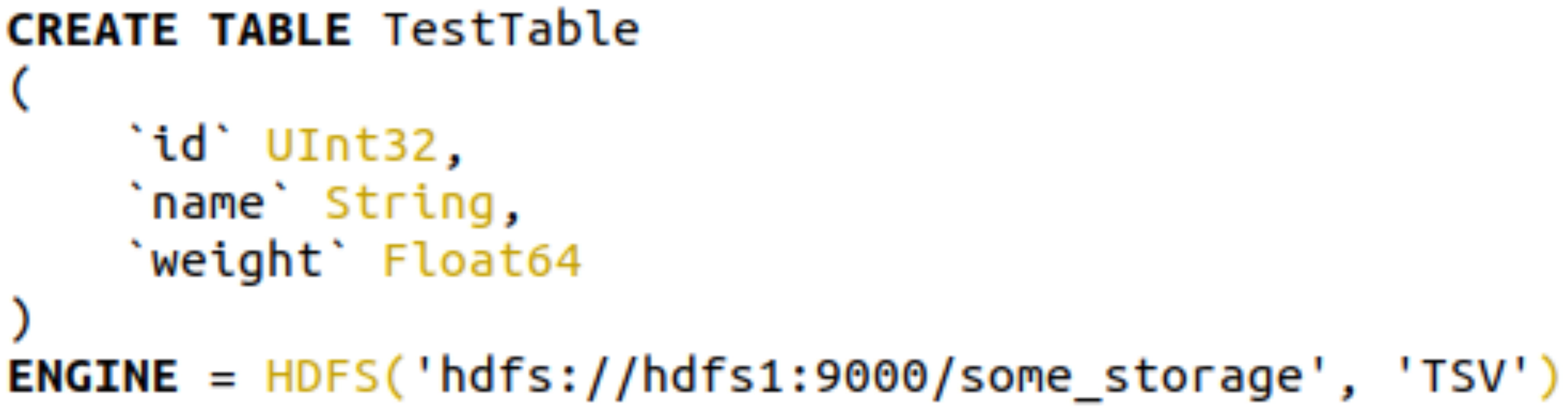
- Parallel parsing of data formats
- (https://github.com/yandex/ClickHouse/pull/5372)
- Avro support
- (https://github.com/yandex/ClickHouse/issues/5601)

Some source code to read in the long autumn evenings...

Or to write...

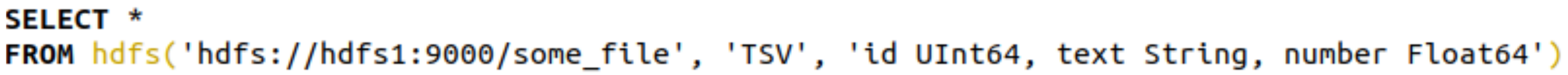
Hadoop Distributed File System as a source

- Engine



```
CREATE TABLE TestTable
(
    `id` UInt32,
    `name` String,
    `weight` Float64
)
ENGINE = HDFS('hdfs://hdfs1:9000/some_storage', 'TSV')
```

- Table function



```
SELECT *
FROM hdfs('hdfs://hdfs1:9000/some_file', 'TSV', 'id UInt64, text String, number Float64')
```

Globs in path (since 19.14)

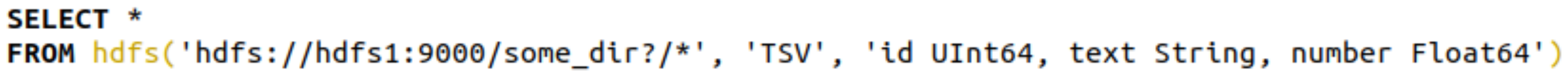
- * - any number of any characters including none
- ? - any single character
- {N..M} - any number in range from N to M
- {aba,caba,bac} - any of strings 'aba', 'caba', 'bac'

similar to remote table function

parallel reading

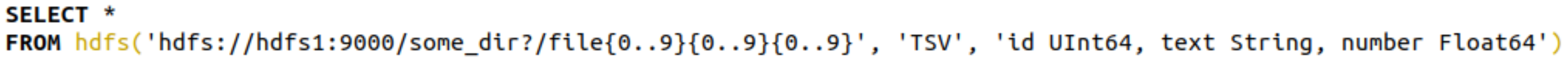
also in file table function

Examples



```
SELECT *
FROM hdfs('hdfs://hdfs1:9000/some_dir?/*', 'TSV', 'id UInt64, text String, number Float64')
```

Multiple path components can have globs



```
SELECT *
FROM hdfs('hdfs://hdfs1:9000/some_dir?/file{0..9}{0..9}{0..9}', 'TSV', 'id UInt64, text String, number Float64')
```

Thank you!

ClickHouse

- YouTube: https://www.youtube.com/c/ClickHouseDB
- Twitter: https://twitter.com/ClickHouseDB
- Ask anything: clickhouse-feedback@yandex-team.com
- GitHub: https://github.com/yandex/ClickHouse
- Telegram: https://t.me/clickhouse_en
- More info: https://clickhouse.yandex