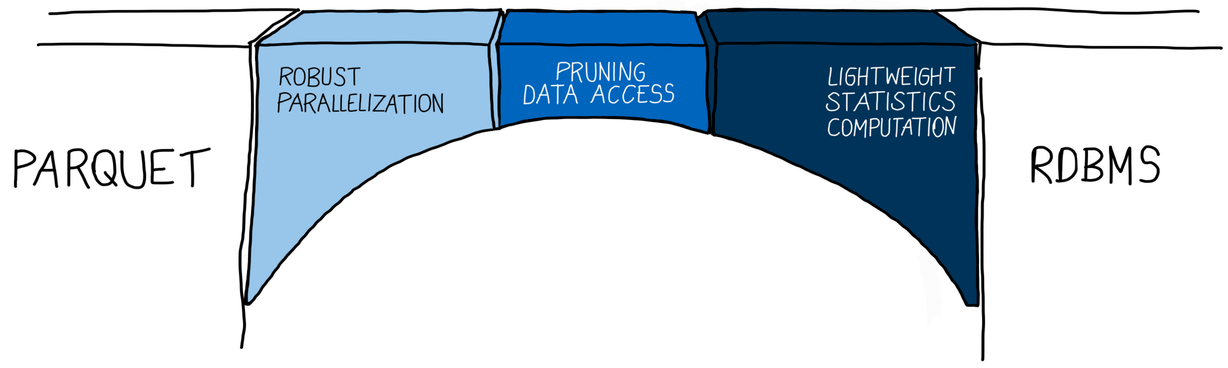
UMBRA

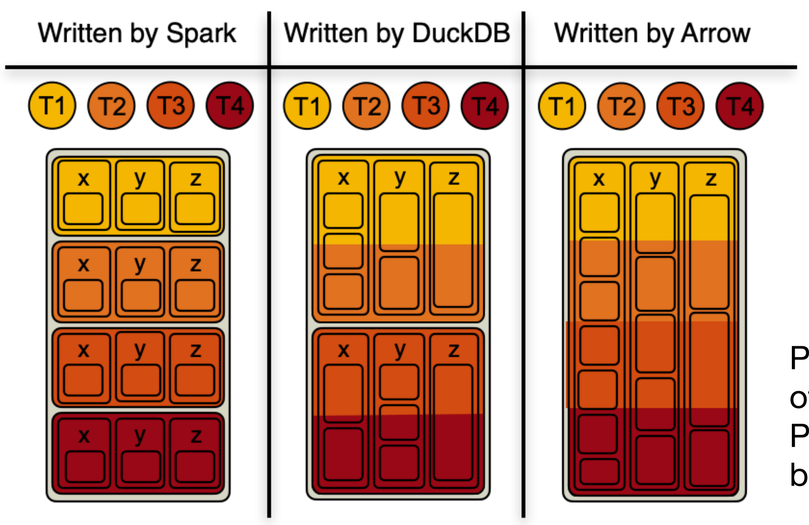
Bridging the Gap between Data Lakes and RDBMSs

Efficient Query Processing with Parquet

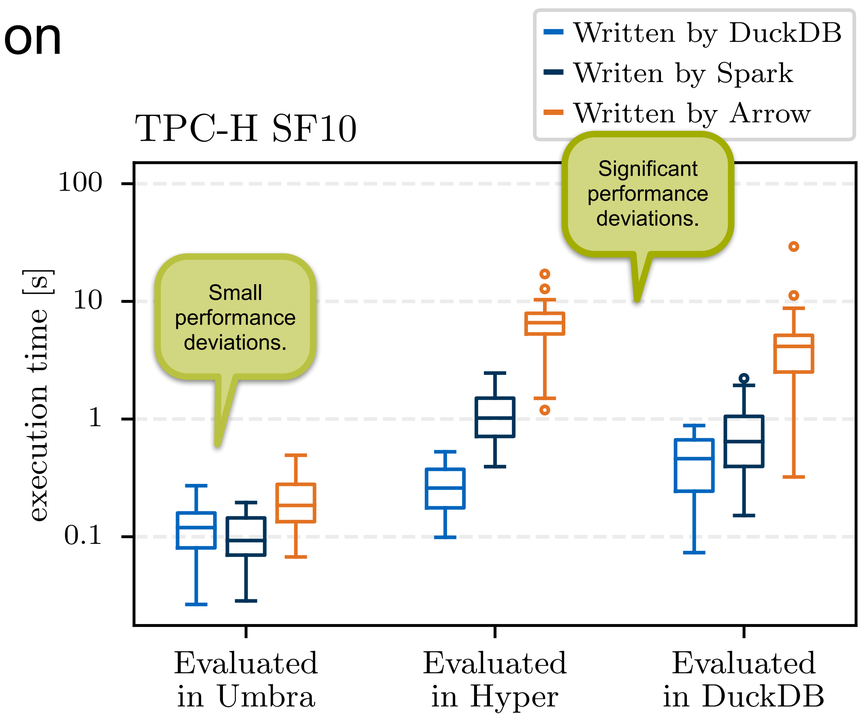


1 Robust Parallelization

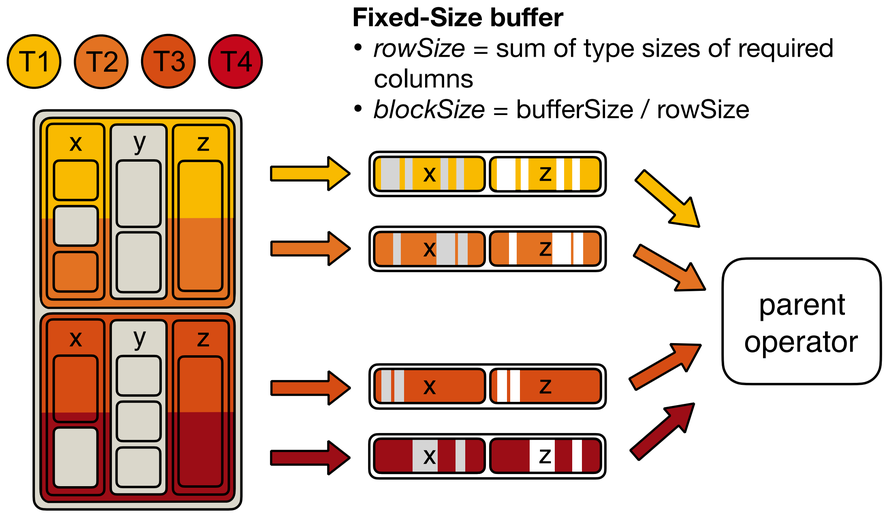
Data can be spread over the Parquet hierarchy in many different ways.



Parallelize independent of the underlying Parquet File structure beneath row group level.



2 Pruning Data Access

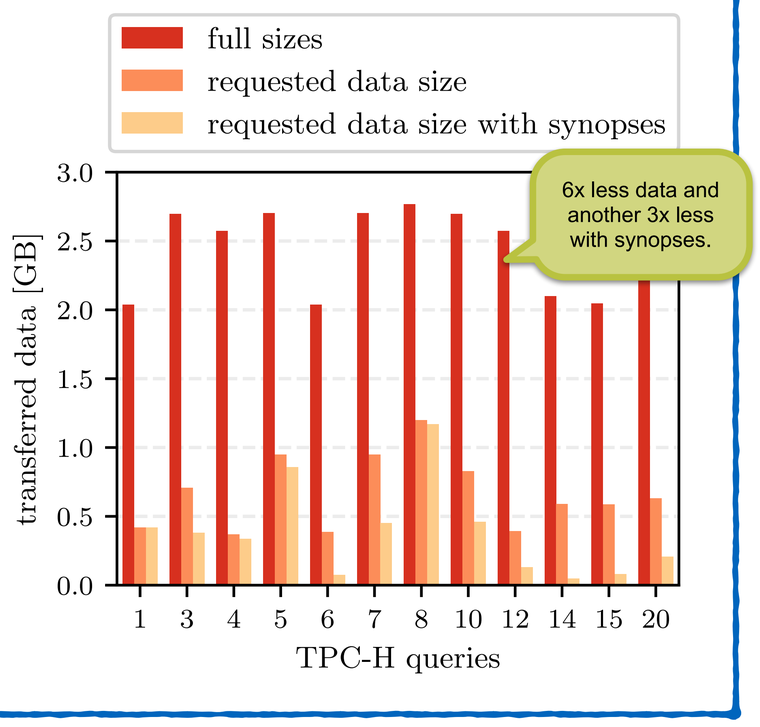


Fixed-Size buffer
- *rowSize* = sum of type sizes of required columns
- *blockSize* = bufferSize / rowSize

Focus on required columns
- Only scan column chunks of required columns

Evaluate predicates early
- Check page ranges
- Skip pages where range does not match restrictions

Vectorized functions
- Evaluate predicates with vectorized functions

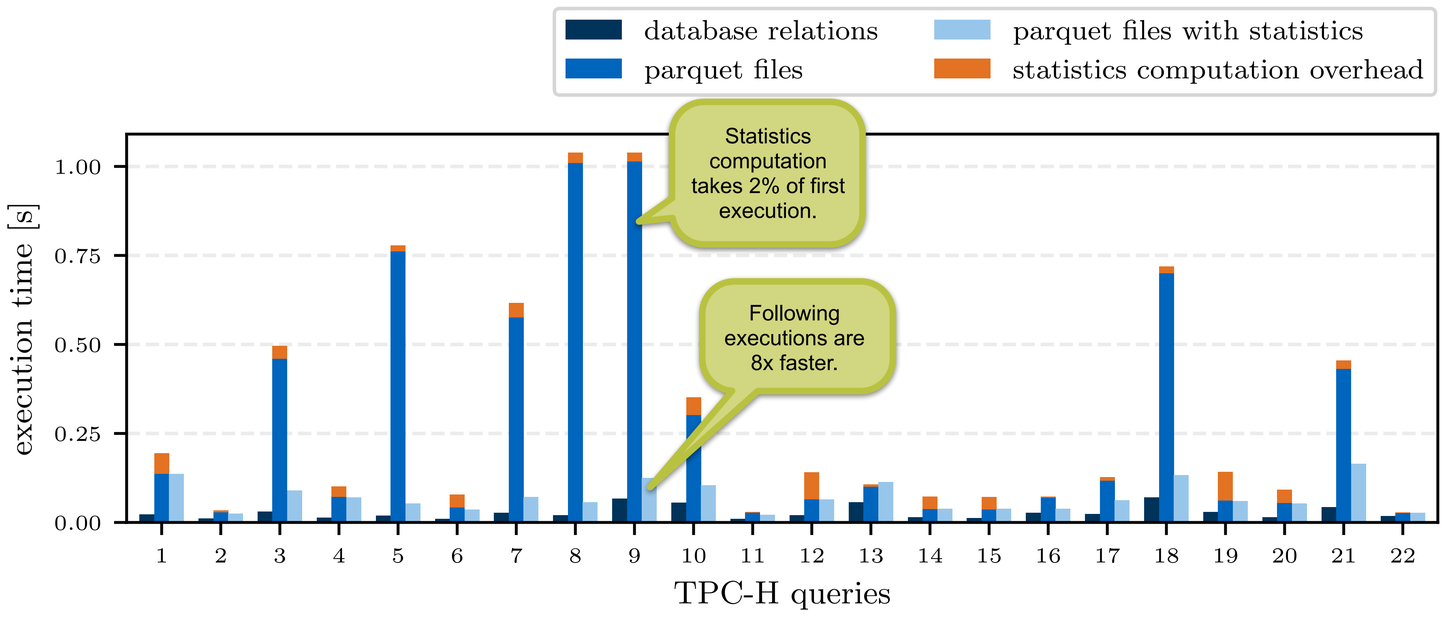


3 Lightweight Statistics Computation

Lazy Computation
- Compute statistics whenever a column is accessed for the first time
- We store samples + HyperLogLog sketches

First File Access
- Collect samples and compute HLL
- Store row numbers of samples to be able to add additional columns later

Following File Accesses
- Use statistics for query plan optimization
- If new columns get accessed, load the sample row numbers to add matching samples and compute HLL sketches



Alice Rey