

# YCSB++: Benchmarking Advanced Features of BigTable-like Stores

Swapnil Patil, Milo Polte, Kai Ren, Wittawat Tantisiriroj, Lin Xiao, Julio López and Garth Gibson (CMU)  
Adam Fuchs and Billie Rinaldi (National Security Agency)

## Motivation

Simple and lightweight scalable table stores evolve into complex, feature-rich stores

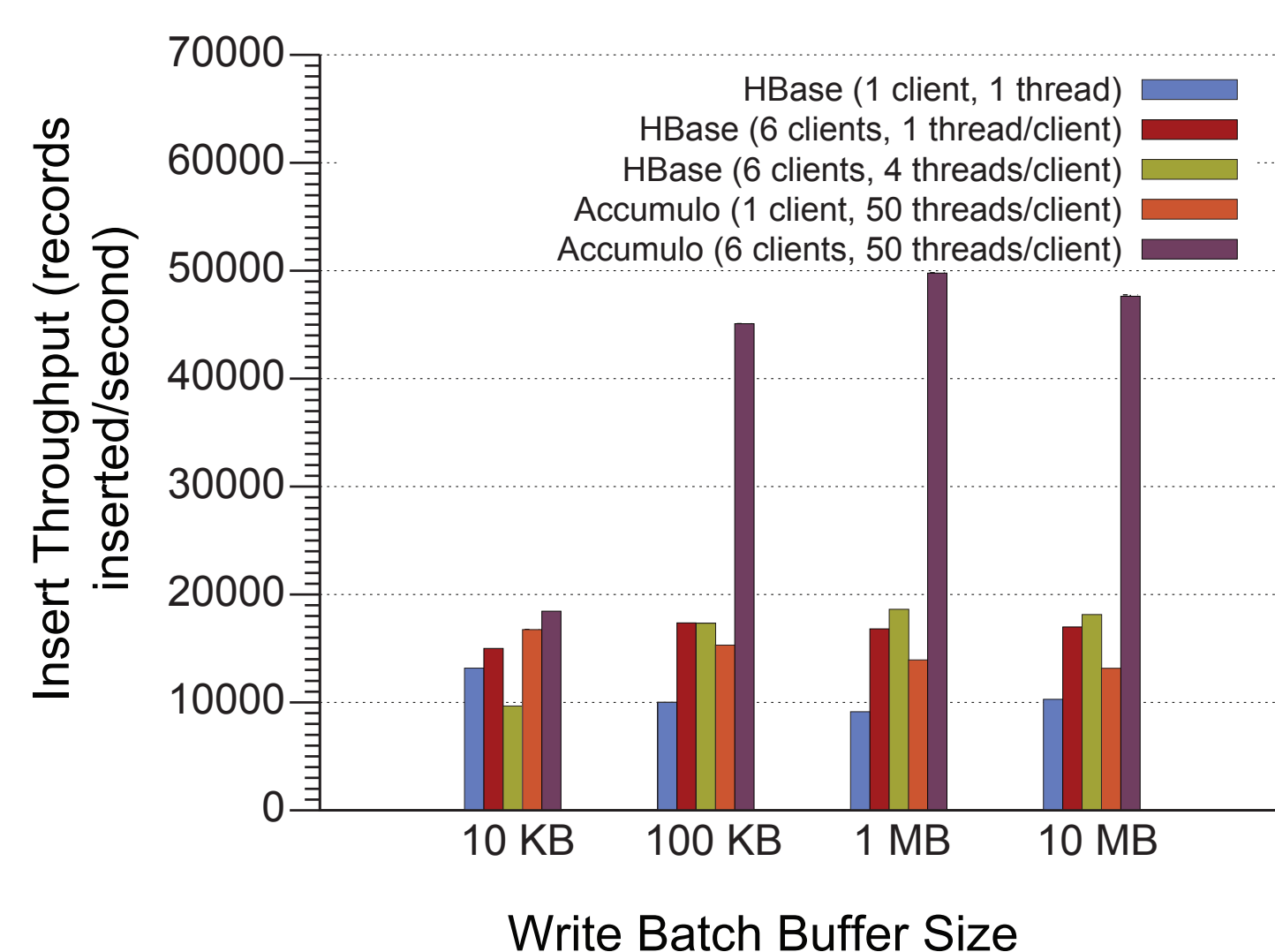
- Hard to debug performance issues & complex interactions
- Need richer tools for understanding advanced features

Case studies of BigTable-like stores

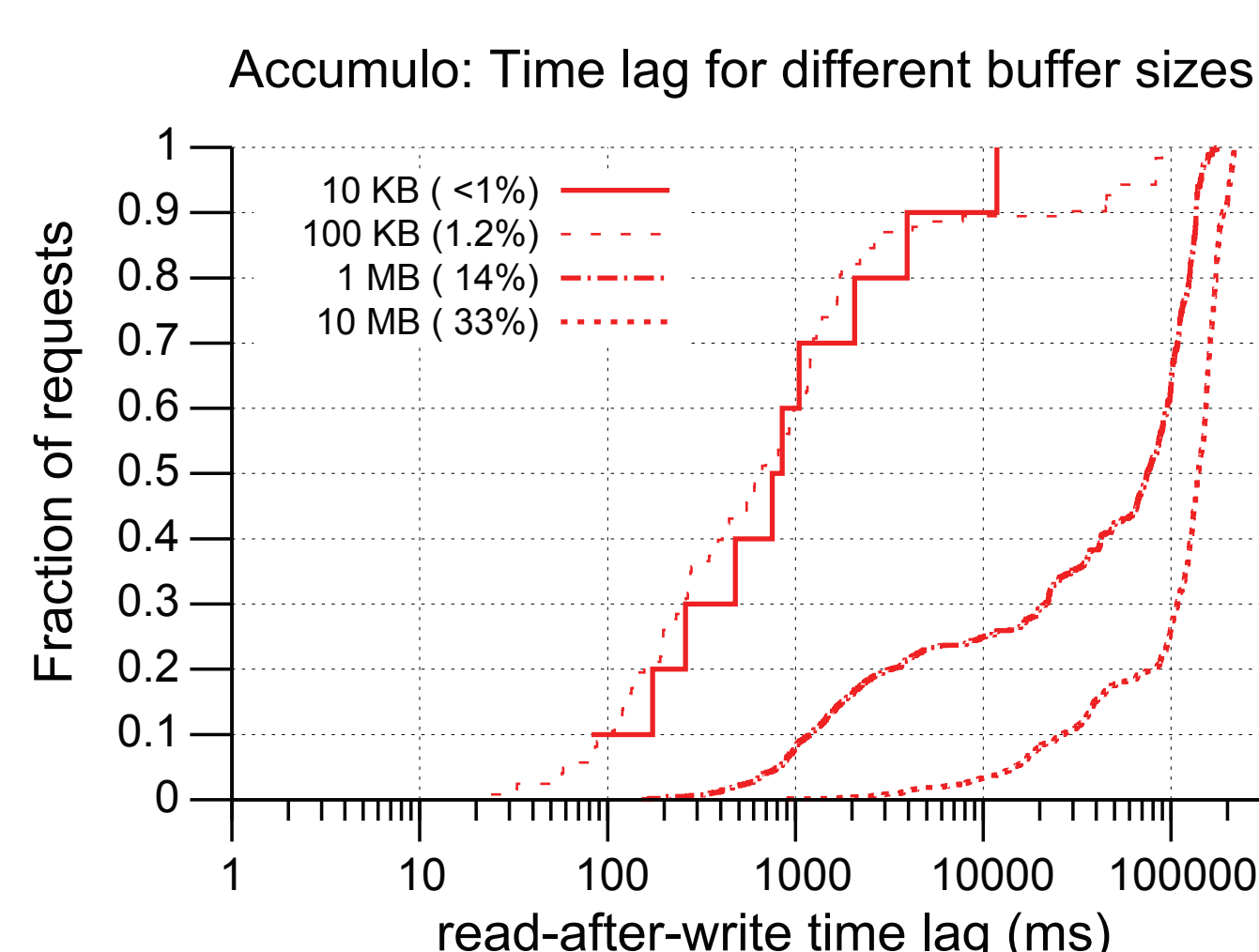
- Apache Accumulo (<http://incubator.apache.org/accumulo/>)
- Apache HBase (<http://hbase.apache.org/>)

## E.g., Batch Inserts

### THROUGHPUT VS. DATA VISIBILITY LAG



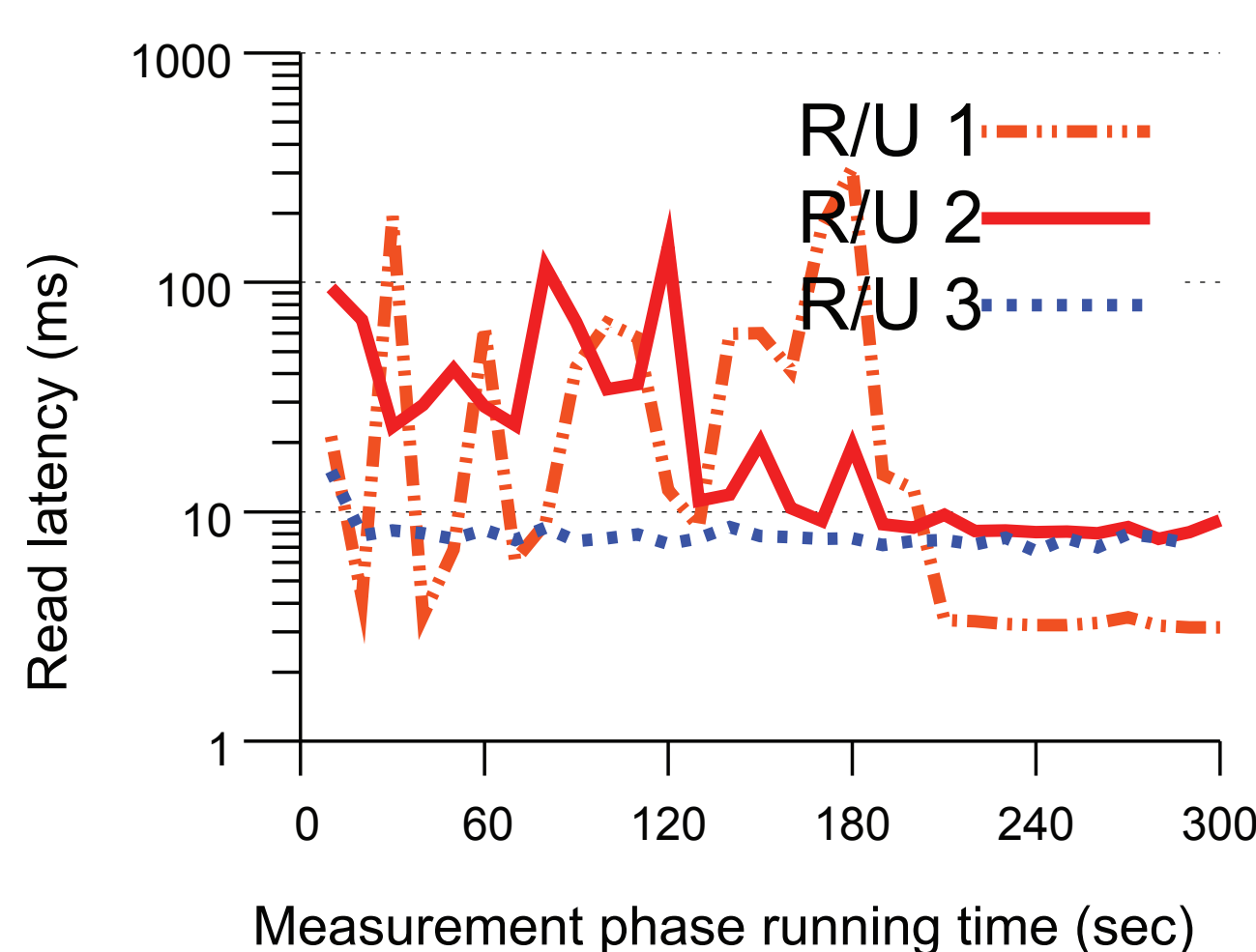
Insert Throughput as a function of DB, batch size, and workload parallelism



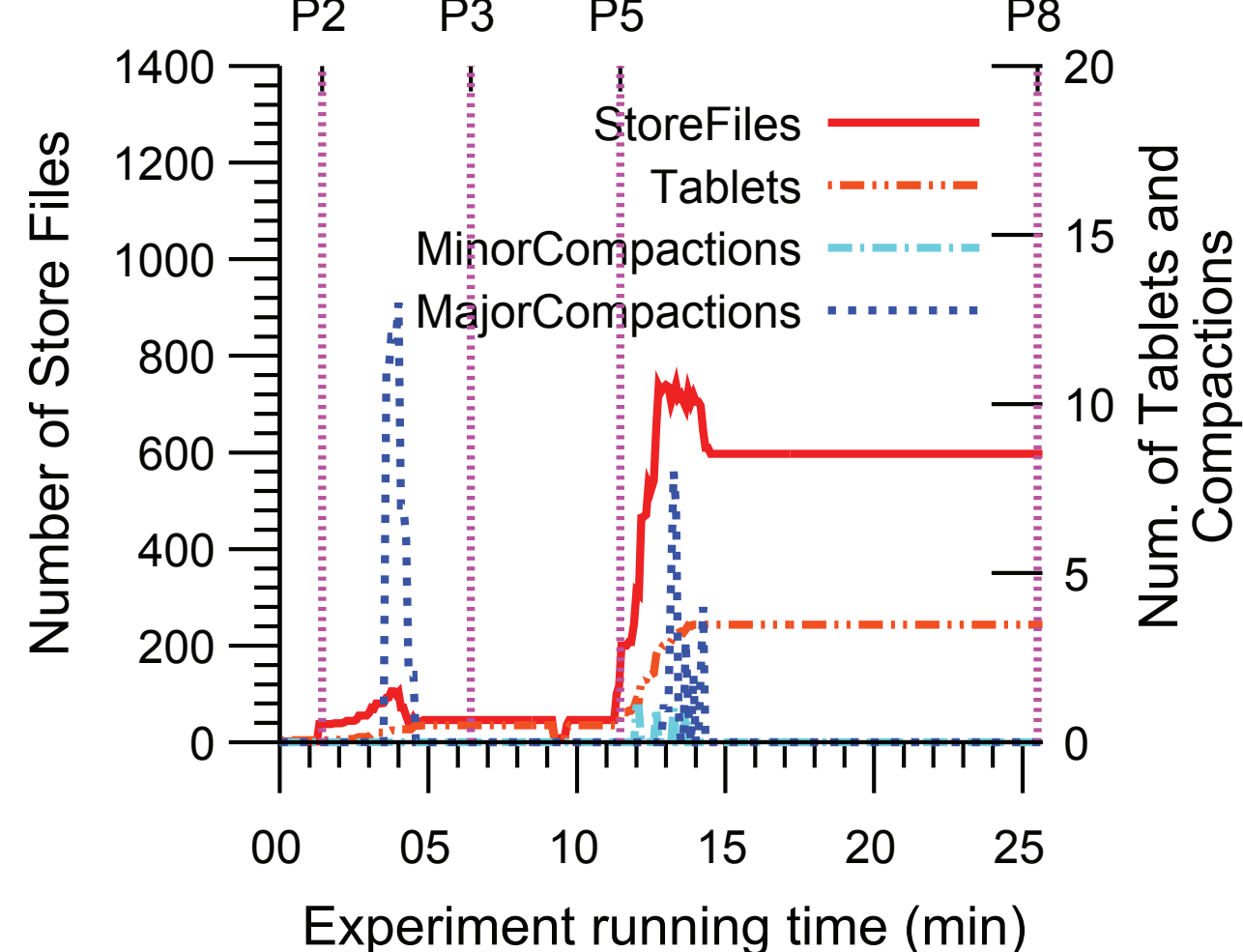
CDF of writes that suffer non-zero lag as seen by another reader

## E.g., Bulk Inserts Using Hadoop/MapReduce

Hadoop/MapReduce job directly converts the data into the "native" on-disk format used by a table store

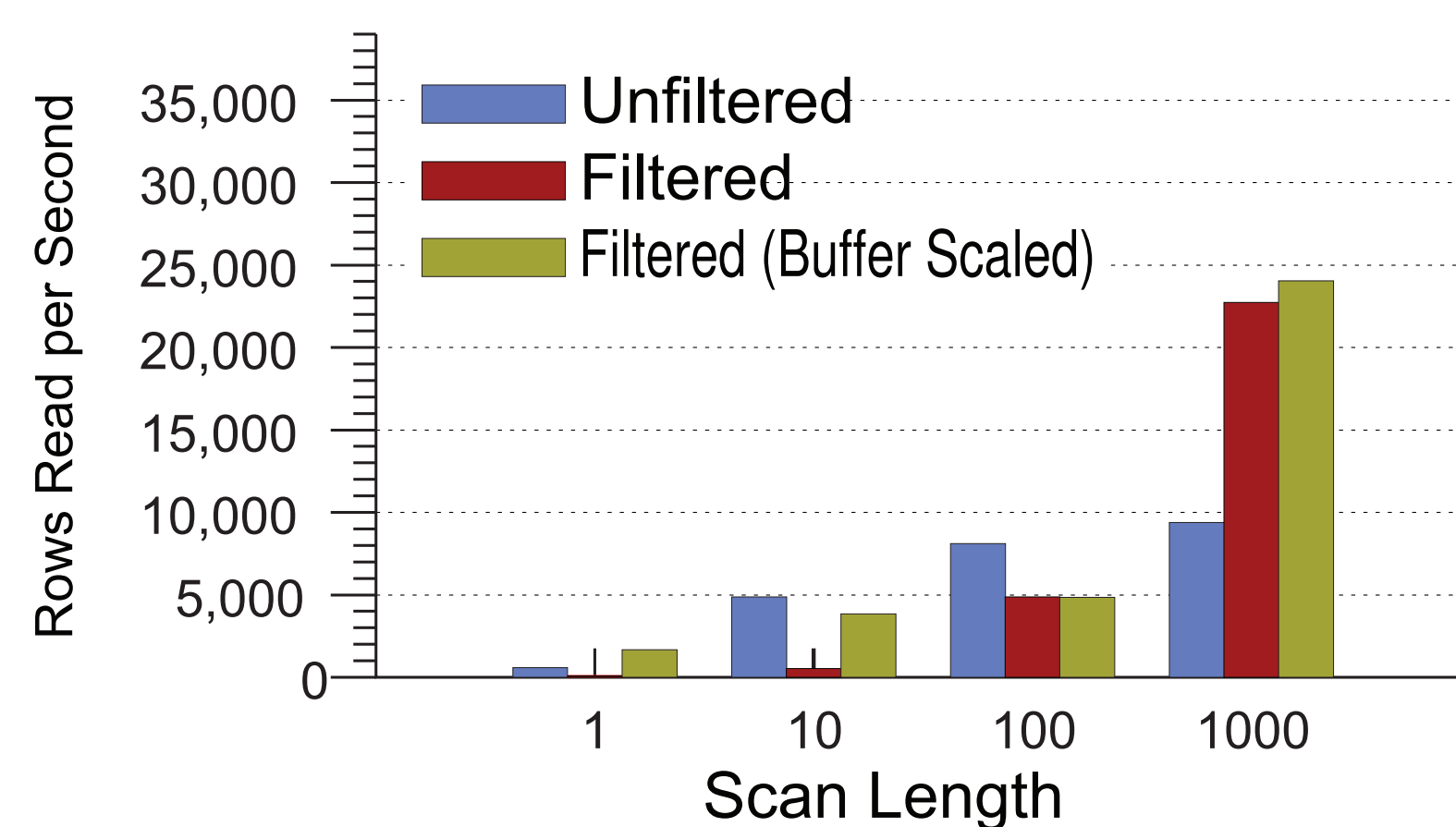


Latency of read operations measured during different phases (as described in the adjoining figure)



Number of store files, tablets, and concurrent compactions relative to different phases that load data (0-P2 and P3-P5) and read data (P2-P3 and P5-P8)

## E.g., Server-side Filtering



Efficiency of scanning different number of rows (called scan lengths) from a table with 100 column families

## New Extensions to YCSB [Cooper2010]

Distributed testing using ZooKeeper

- Synchronized multi-phase testing
- Co-ordinated multi-client testing

Ingest-intensive workload extensions

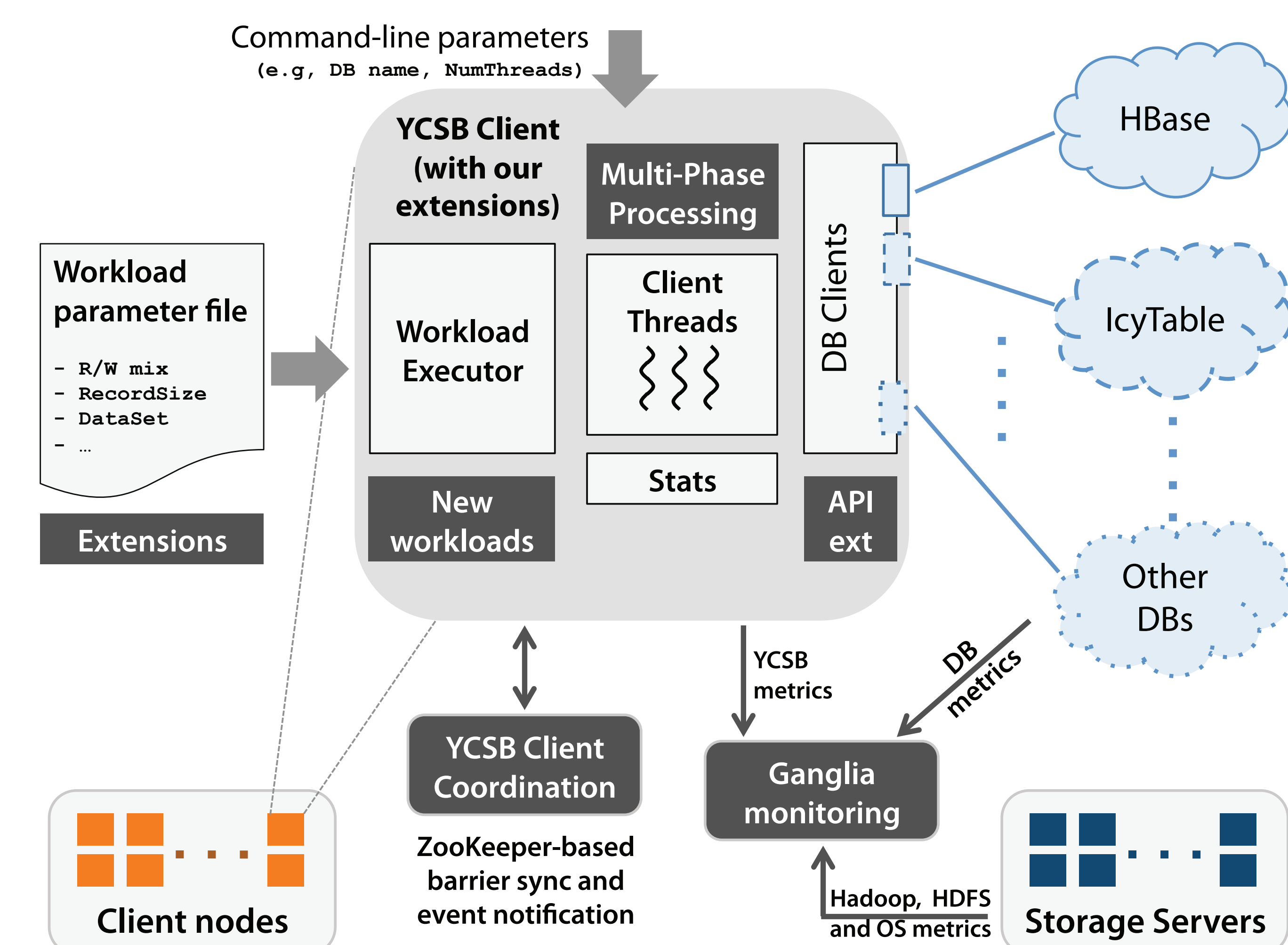
- Hadoop/MapReduce tools for bulk insertions
- Range pre-splitting for B-tree indices

Offloading functions to DB servers

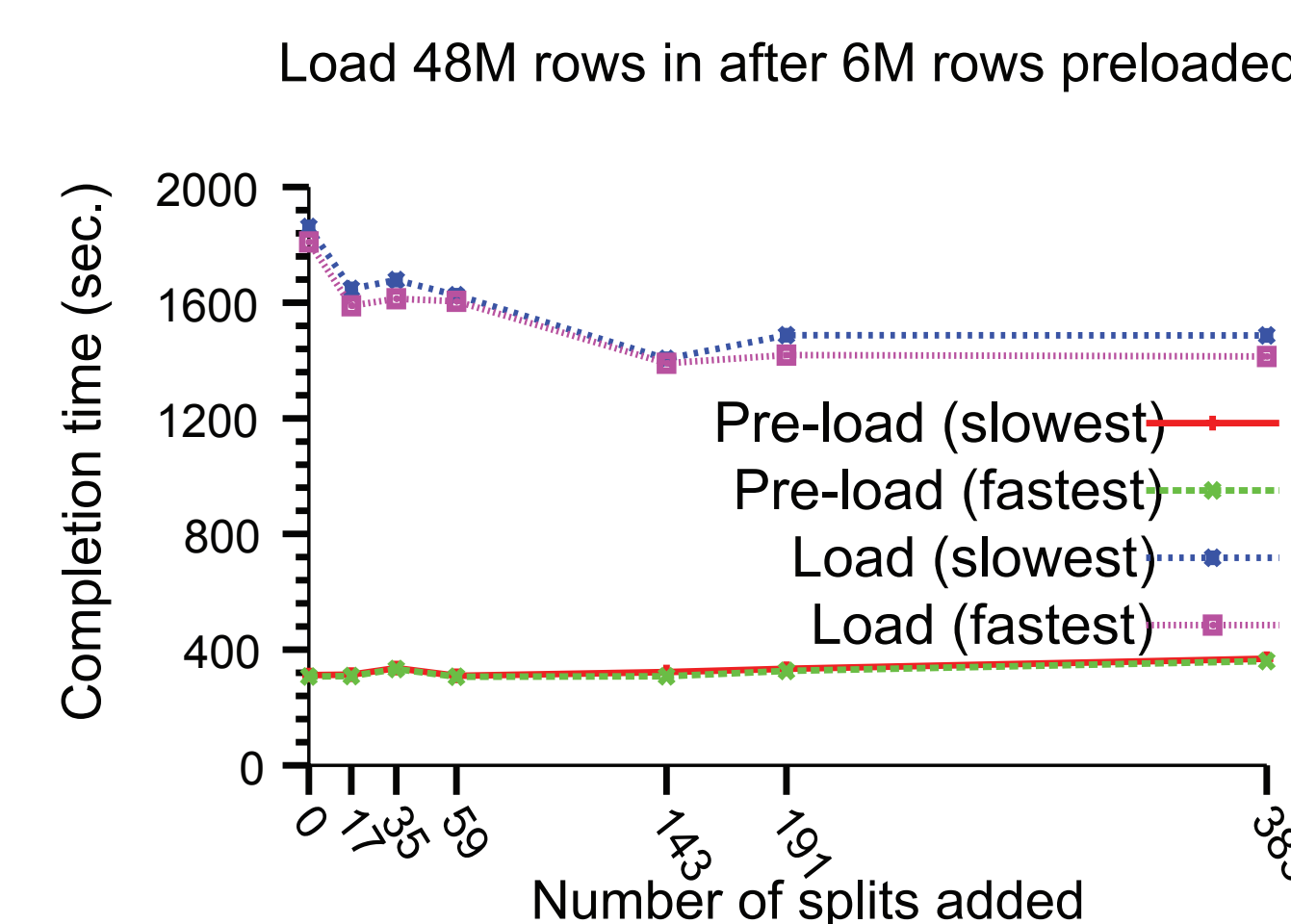
- Server-side filtering for efficient querying

Fine-grained access control

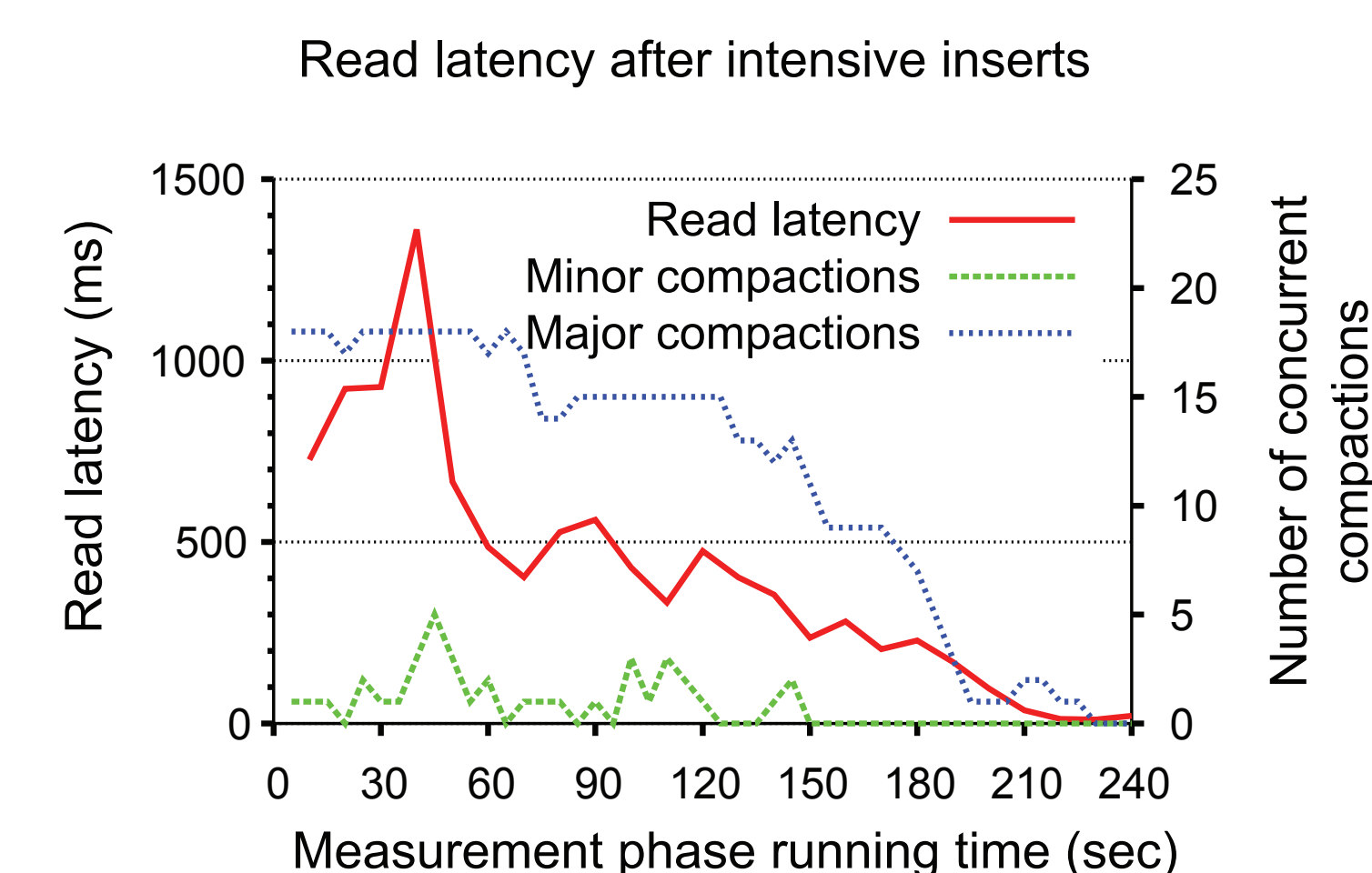
- Testing schema-level and cell-level access control



## E.g., Inserts in a Pre-split Table



Pre-splitting the key range into varying number of equal-sized ranges can speed-up ingest-intensive workloads



Correlation of latency of read operations (performed after data ingest) with compaction operations on the servers