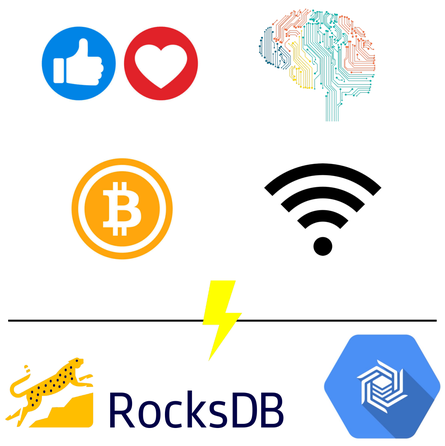
HARVARD
John A. Paulson
School of Engineering
and Applied Sciences

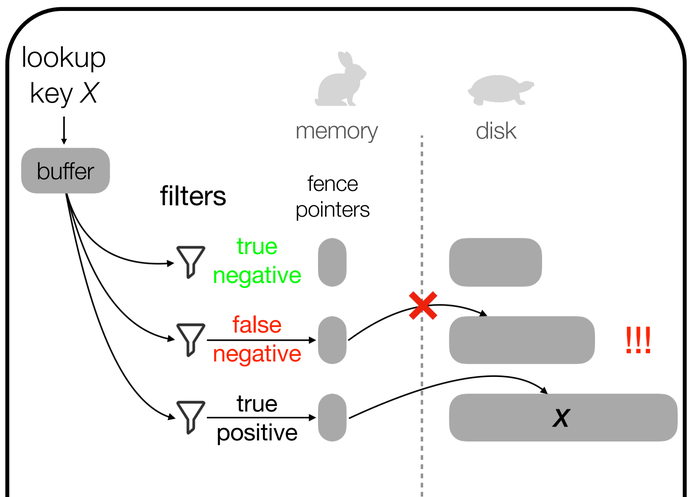
Workload-Adaptive Filtering in Storage Engines

Joshua Pan | Advised by Brian Hentschel and Stratos Idreos

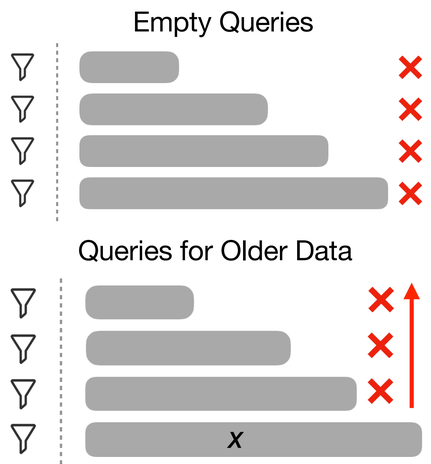
The Problem: LSM-tree based storage engines suffer with empty queries and skew workloads



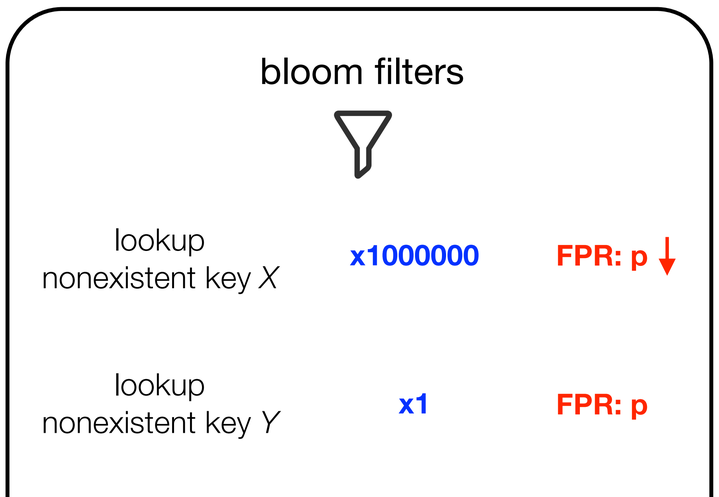
LSM-tree based storage engines are everywhere



LSM-tree performance relies on filters to prevent unnecessary disk I/Os

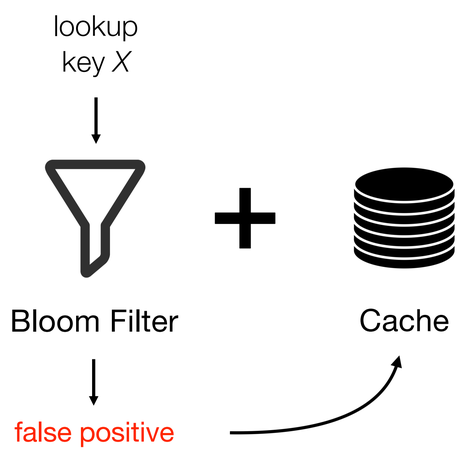


Common queries incur many unnecessary I/Os

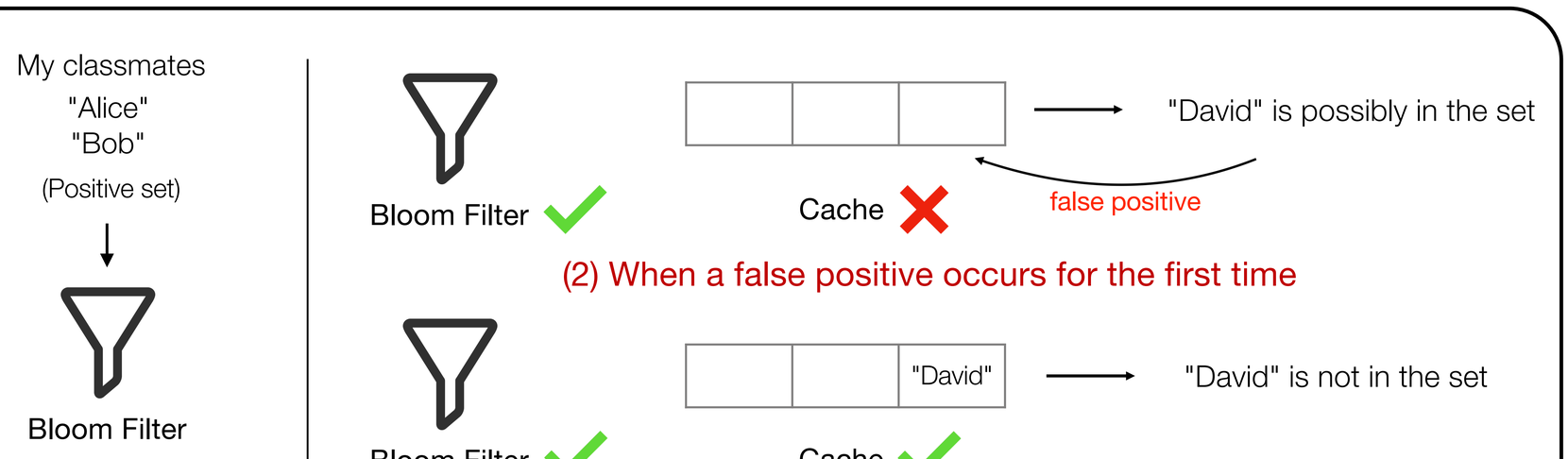


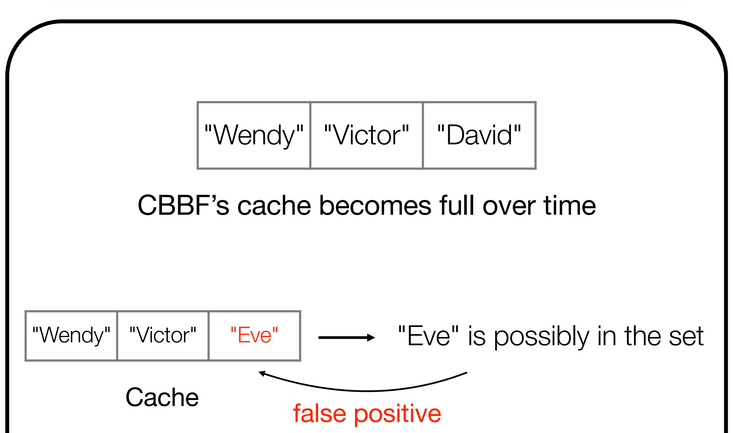
Repeated queries are an issue for current storage engines and call for adaptivity

Our Contribution: Cache-Backed Bloom Filters (CBBFs)

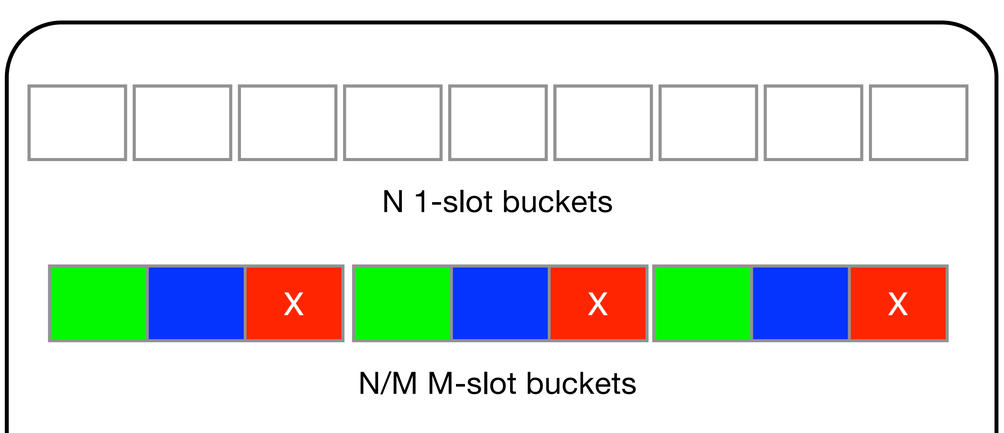


Main idea: remember bloom filter's false positives in a fixed-size cache (hashtable)

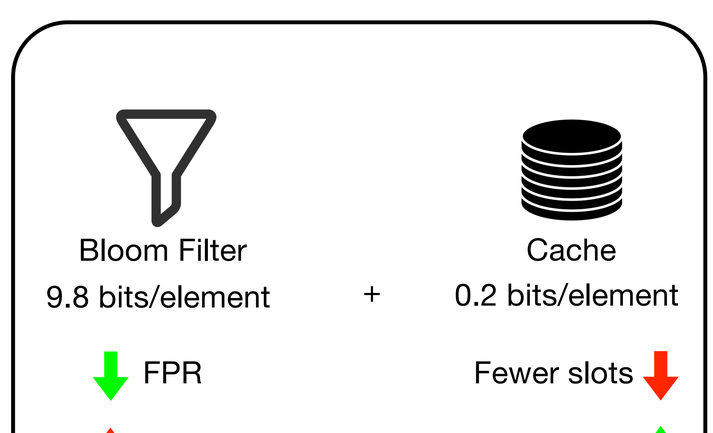




Existing false positives are evicted randomly on hash collision

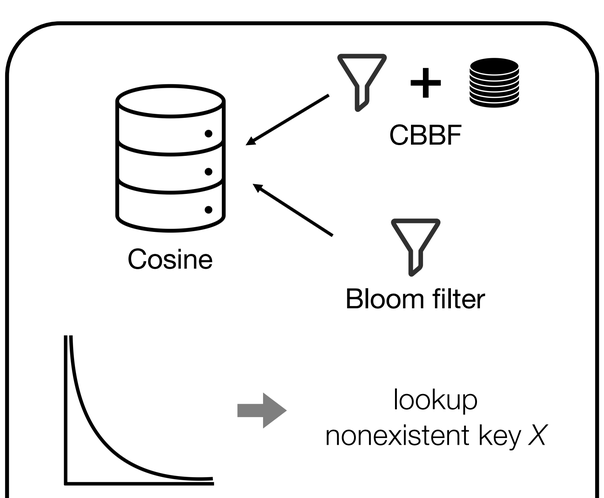


Least recently used (LRU) eviction policy to retain frequently queried false positives in the cache with high probability

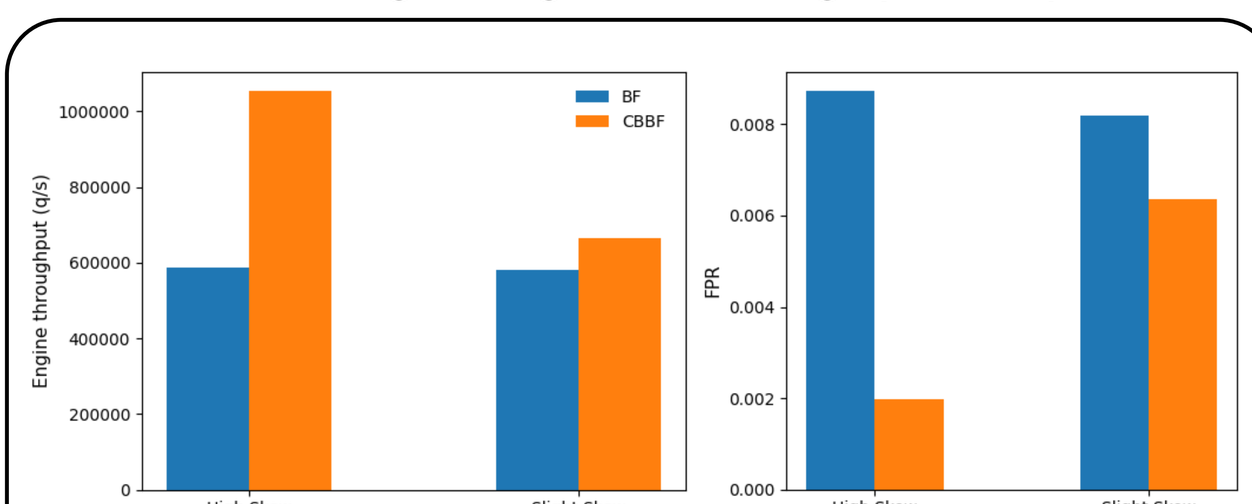


Tuning CBBFs: How to allocate memory between bloom filter and cache?

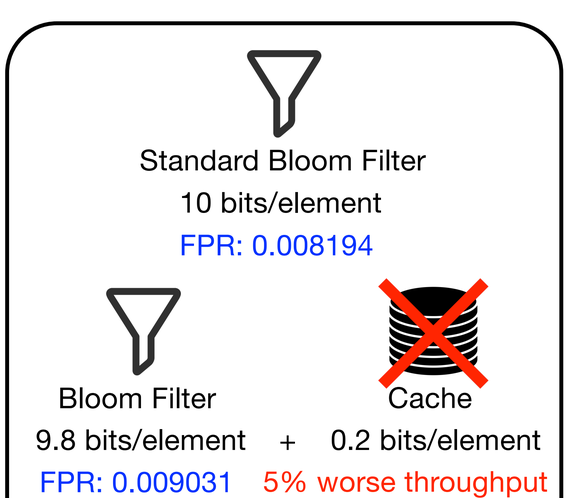
End-to-End Storage Engine Throughput Improvement



Experimental Setup



CBBFs perform better than standard bloom filters when queries repeat



Near identical performance to bloom filters in worst case