Faster Columns: Adding Indices to Modern Data Stores
Michael S. Kester, Zezhou Liu, Wilson Qin, and Stratos Idreos
Harvard University School of Engineering and Applied Science

Modern data stores are fast if we add indices = are they even faster?

SELECT: scan vs. index
We compare the query throughput of a system that relies only on scans to one that uses indices.

Our Index includes:
- A cache-sensitive b+tree
- Optimized Fanout
- Optimized node searching

Our scan uses:
- Single Instruction Multiple Data (SIMD)
- Shared Scanning
- Multicore Processing
- Compression

Examples of shared scanning
- Basic Scan
  a) Q0
  b) Q1
- Batched Scan
  c) Q1, Q2
  Intercitable Scan
  - Q1 Consumed Data
  - Q2 Consumed Data

Choosing Fanout
- Inner (SIMD)
- Inner & Index
- Index & outer (SIMD)
- Index & outer

Results 10M Values
- An index should be used when S < ~7%
- Memory: 1.65x / 2.65x
- Build time: 81.31 vs 1.35