

Access Path Selection in Modern Columnar DBMSs



Background

Performance of the analytical DBMSs depends on the efficient data access.

State-of-the-art solutions:

- Zone maps (min/max pruning)
- Column sketches (SIMD / AVX-512)
- Bitmap indexing

Storage evolution:

Traditional analytical systems → pure columnar layout

Modern systems → hybrid layout (PAX-based)

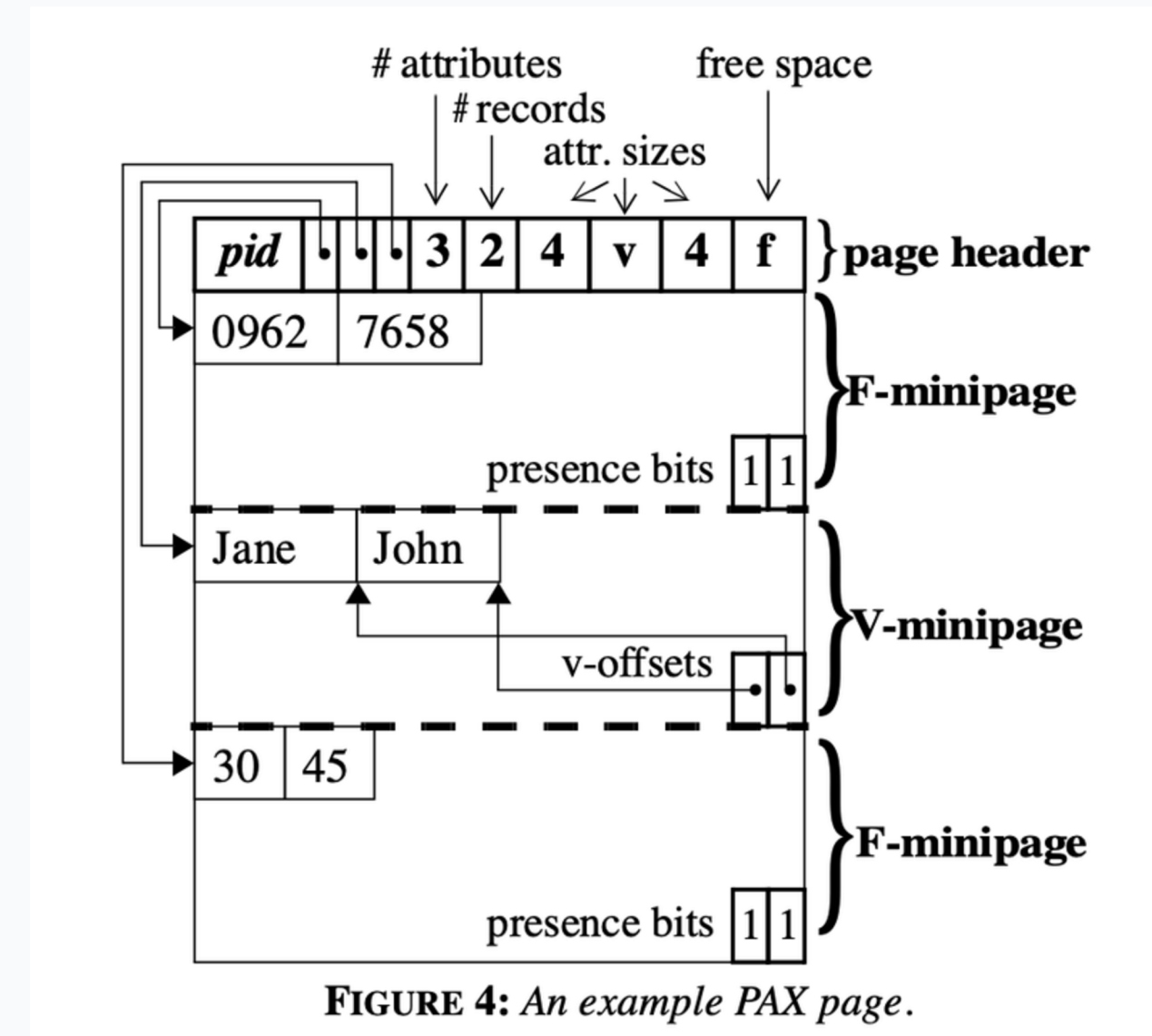
Background

PAX (Partition Attributes Across):

- Data organized by pages → minipages per column
- Combines:
 - Row locality (within page)
 - Columnar access (within minipage)

Key Benefits:

- Better CPU cache utilization
- Reduced unnecessary data access



Source: Ailamaki et al., VLDB'01

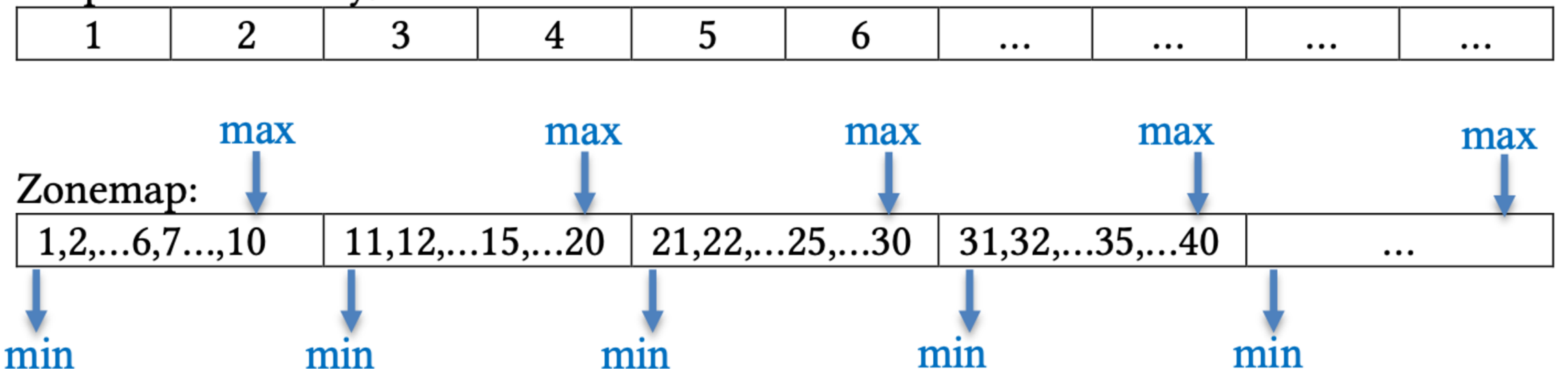
Motivation

- Existing research on retrieval approaches mainly focuses on pure columnar DBMSs
- Modern analytical systems adopt hybrid layouts with hardware-aware and advanced indexing techniques (AVX-512 vectorized sketch evaluation and CUBIT's concurrent bitmap indexing)
- Optimal access path selection in such modern environments remains an open question

Zonemaps

- Metadata structure storing min/max values per block/page of a column
- During a scan, blocks are skipped entirely if the predicate falls outside the min/max range
- Most effective when data is sorted or clustered (more blocks can be pruned)

Simple sorted array/vector:



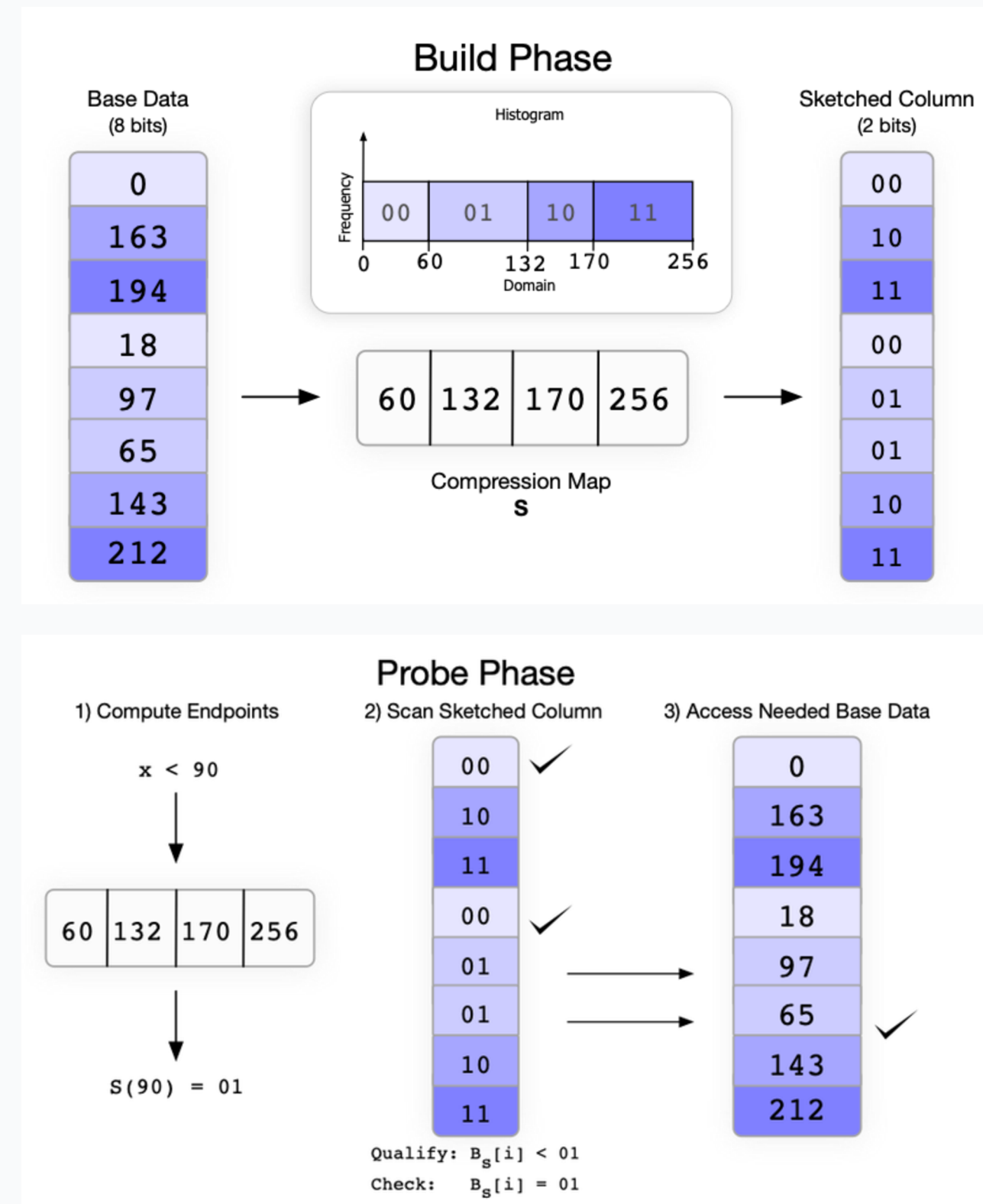
Bit Maps

- Maps each distinct column value to a bit vector over all rows
- Each bit indicates whether that row contains the value
- Queries use fast bitwise AND/OR/NOT operations to combine predicates
- Very efficient for low-cardinality columns
- CUBIT and RABIT extend this to support concurrent updates and range queries

Column A	A=10	A=20	A=30
30	0	0	1
20	0	1	0
30	0	0	1
10	1	0	0
20	0	1	0
10	1	0	0
30	0	0	1
20	0	1	0

Column Sketches

- A lossy compressed summary of a column using fewer bits per value
- Maps original values to a smaller domain
- Predicates are evaluated on the sketch first
- False positives are possible but no false negatives, correctness is guaranteed
- Accelerated with AVX-512 SIMD instructions for vectorized sketch evaluation



Source: Hentschel et al., SIGMOD 2018

Research Objectives

01 Evaluate performance of data retrieval methods

02 Analyze impact of selectivity & data distribution

03 Study behavior under modern PAX-based storage

04 Build access path selection model

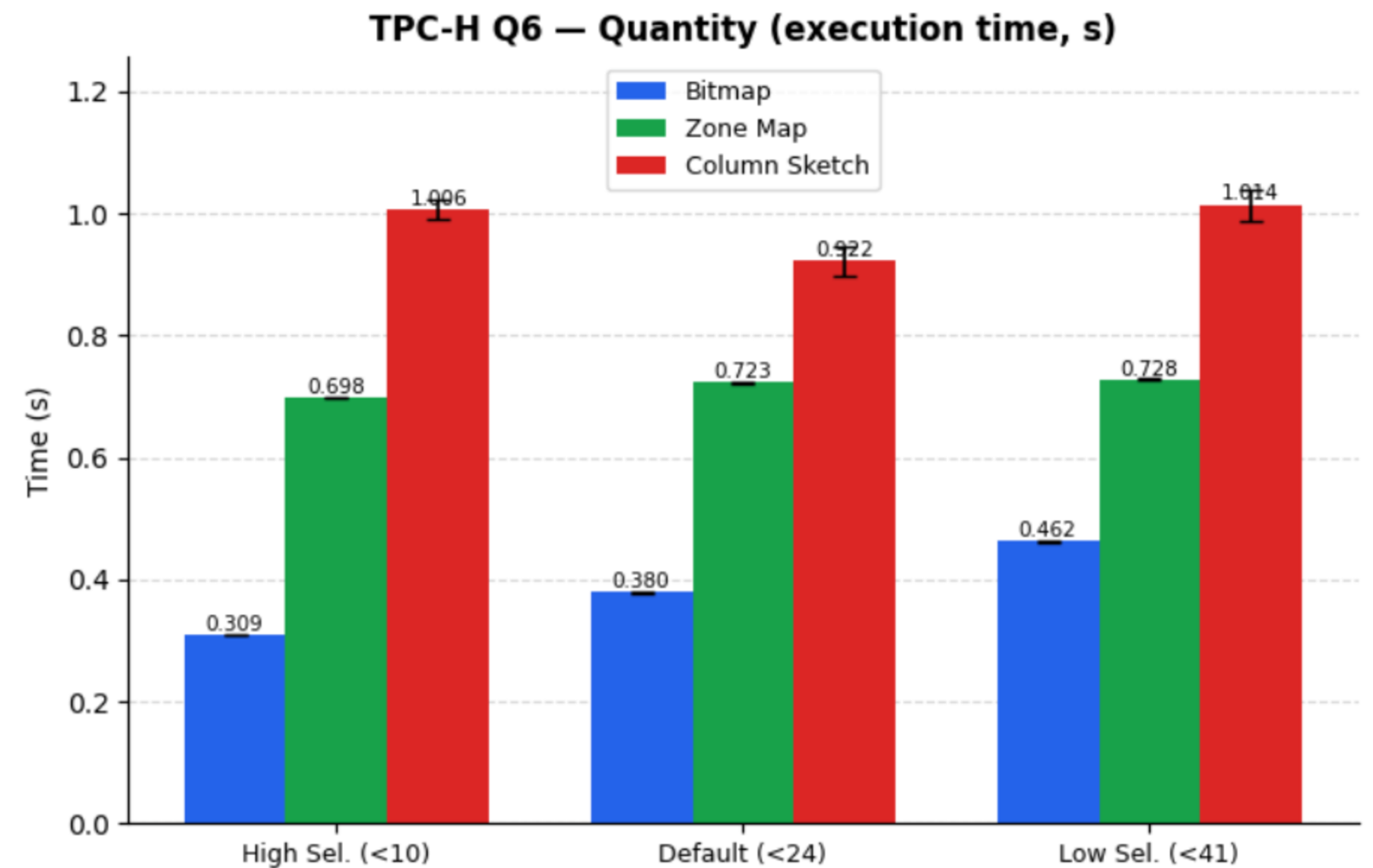
Experimental Setup

- Infrastructure: Intel Xeon Platinum 8576C, 16 logical CPUs @ 2.50 GHz, AVX-512 support
- Database: DuckDB (PAX-based, vectorized execution)
- Dataset: TPC-H, SF = 10 (10 GB)
- Queries: TPC-H Q1, Q6, and a custom query with predicates on *shipdate*, *quantity*, *extended price*, and *discount*
- Workload: Varying selectivity (low / medium / high), across different column types and predicate operators
- Metrics: Execution time and pruning effectiveness

TPC-H 6 Results (10 gb)

- Ranges from: 1-50

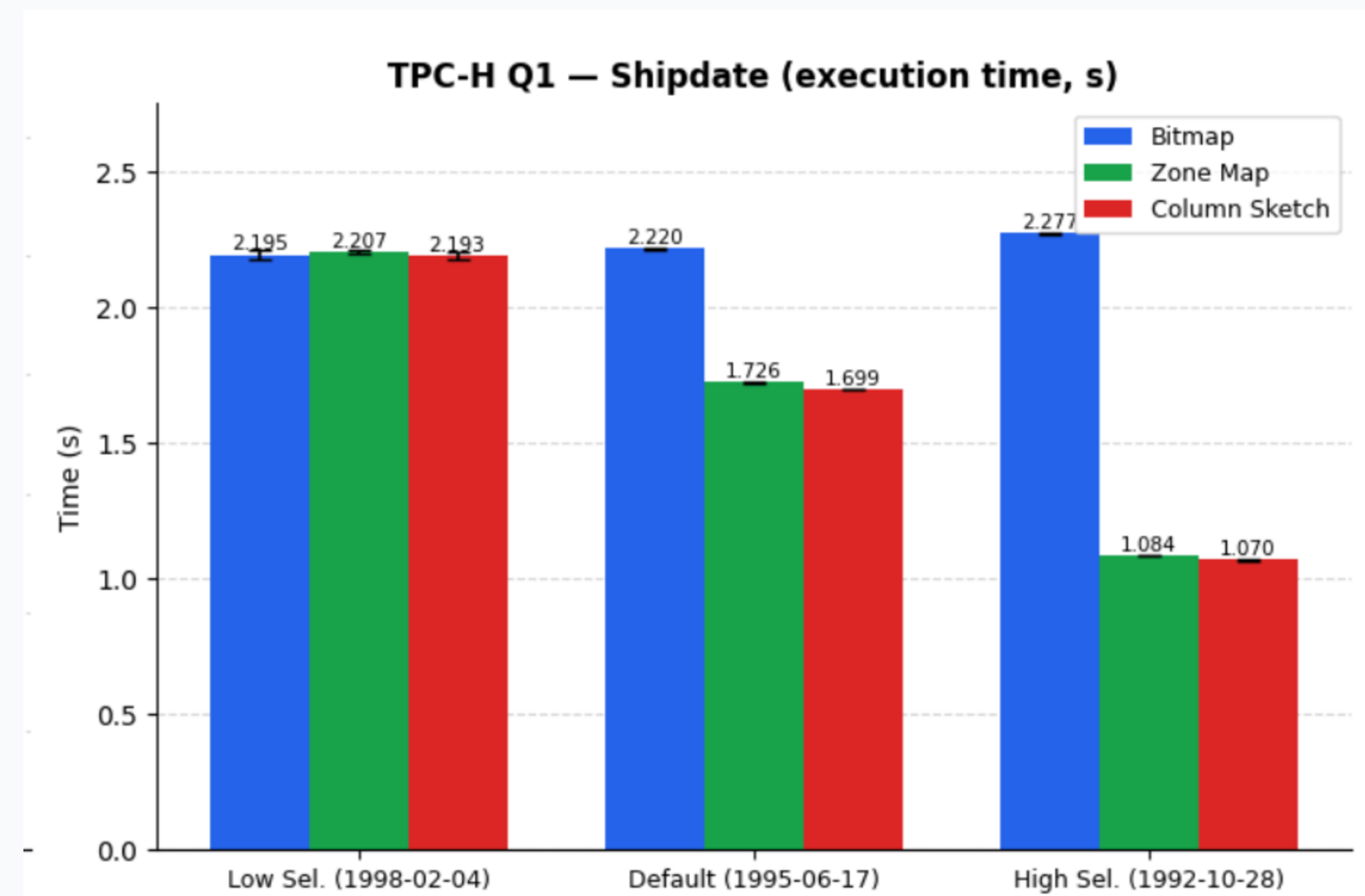
- Bitmaps thrive under columns with low cardinality distributions (Quantity ranges from 1-50)
- Zone maps beat out column sketches in similar workloads.



TPC-H 1 (Shipdate) Results (10 gb)

- Ranges from: 1992-01-02 to 1998-12-01

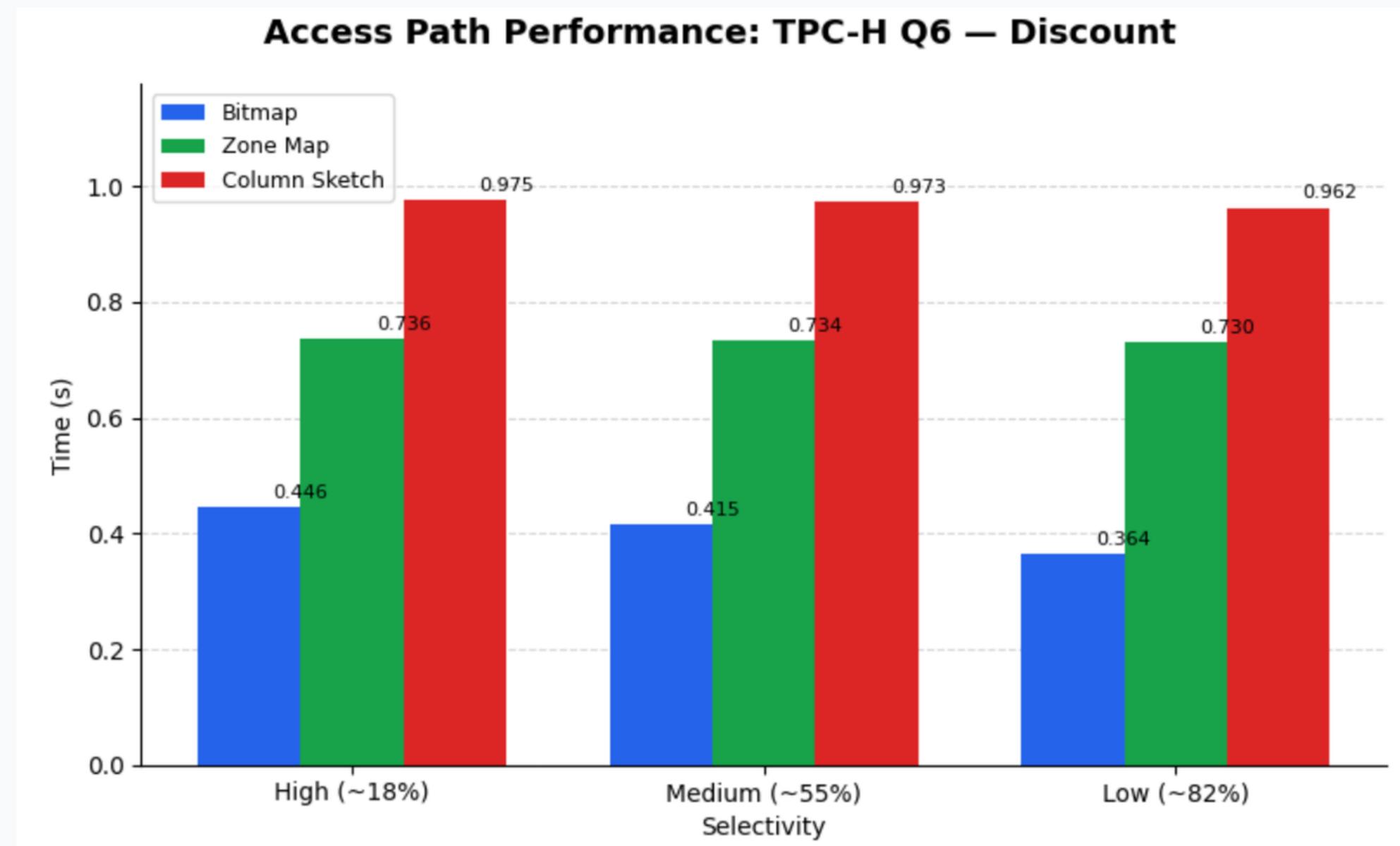
- Bitmaps suffer greatly from data with lots of points
- Column sketches thrive under high cardinality numeric columns
- Zonemaps benefit from clusters/semi sorted data
 - Shipdate is a perfect example



TPC-H 6 Results (Discount) (10 gb)

- Ranges from: 0.00 and 0.10

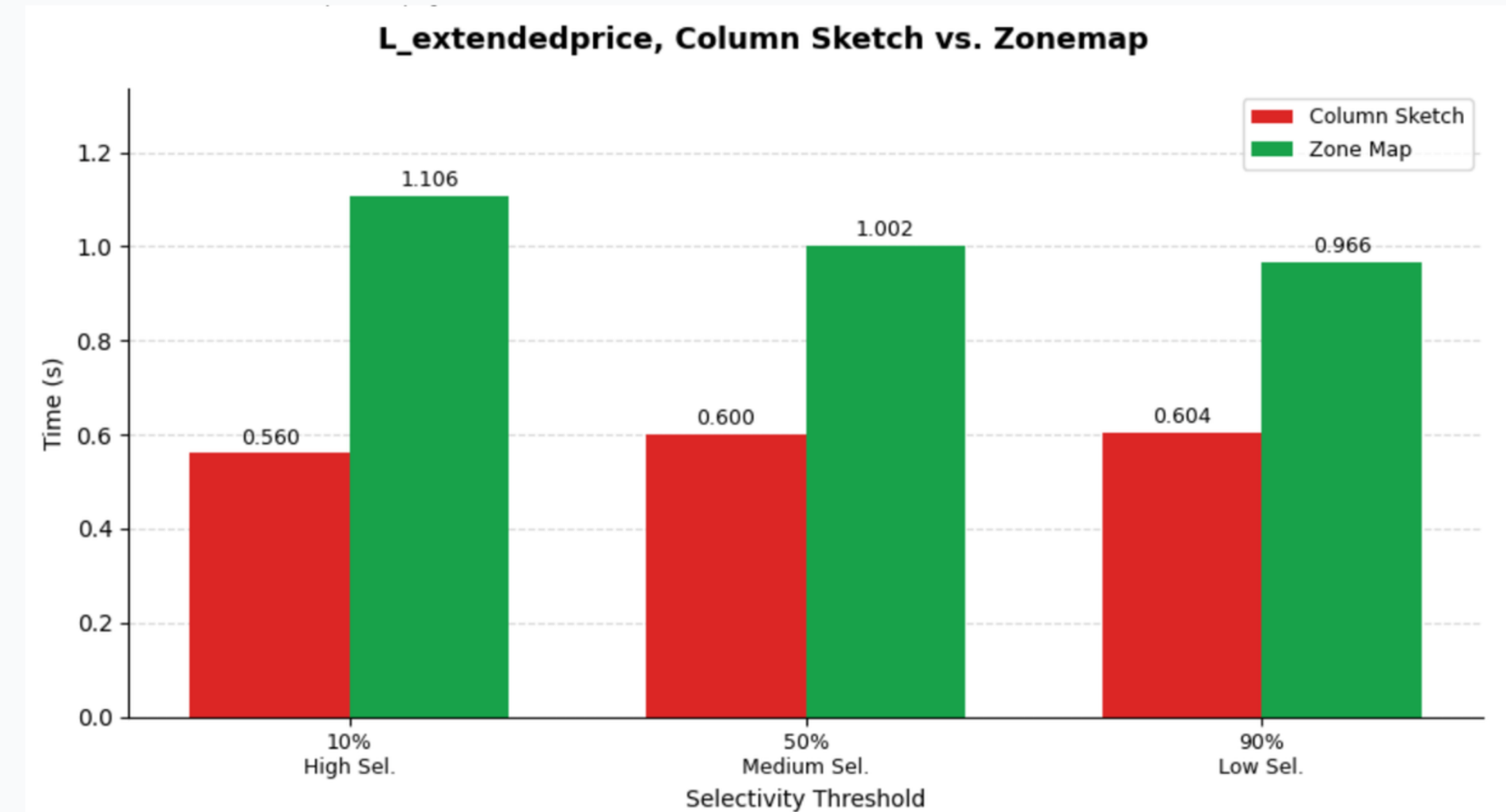
- Once again illustrates how bitmaps thrive under limited unique data points
- Zone maps beat out column sketches in similar workloads.



Extendedprice (Custom Query) Results (10 gb)

- Ranges from: ~ 900 to ~ 100,000

- Extendedprice is a high cardinality, unsorted column
- Zonemaps start to suffer due to lack of pruning
- Column Sketches thrive, compressing the column allowing for quick scans.



Conclusion

- Despite state of the art acceleration (AVX-512 vectorized sketch evaluation and CUBIT's concurrent bitmap indexing) no single access path dominated across workloads.
- Hardware optimizations shift the performance ceiling, but they do not change which method wins. Cardinality and sortedness remain the governing factors even in PAX-based DMBS.
 - Zonemaps when near sorted/clustered data
 - Bitmaps on low cardinality numeric columns
 - Column Sketches on high cardinality, unsorted data
- These rules held consistently across selectivity levels, query types, and scale
- Access path selection cannot be solved at the hardware layer alone, it must be data aware to be truly optimal.

Future work

- 01** Extend evaluation to more workloads, data distributions, and other indexes (B+, LSM)
- 02** Test and explore ideal ways for a DBMS to remain data aware.
- 03** Design a self tuning DBMS component that dynamically selects or transitions between access paths

References

- [1] B. Hentschel, M. S. Kester, and S. Idreos. 2018. Column Sketches: A Scan Accelerator for Rapid and Robust Predicate Evaluation. In Proceedings of the 2018 International Conference on Management of Data (SIGMOD '18).
- [2] J. Wang and M. Athanassoulis. 2024. CUBIT: Concurrent Updatable Bitmap Indexing. Proc. VLDB Endow. 18(2), pp. 399–412.
- [3] J. Wang, F. Xiao, and M. Athanassoulis. 2025. RABIT: Efficient Range Queries with Bitmap Indexing. Proc. ACM Manag. Data. 3(6), pp. 354.
- [4] Daniel J Abadi, Daniel S Myers, David J DeWitt, and Samuel R Madden. 2006. Materialization strategies in a column-oriented DBMS. In 2007 IEEE 23rd International Conference on Data Engineering. IEEE, 466–475.
- [5] A. Ailamaki, D. J. DeWitt, M. D. Hill, and M. Skounakis. 2001. Weaving Relations for Cache Performance. In Proceedings of the 27th International Conference on Very Large Data Bases (VLDB '01), pp. 169–180.
- [6] <https://github.com/junchangwang/CS561-Access-Path-Selection.git>

Thank you!