

# FastBit: An Efficient Indexing Technology For Accelerating Data Intensive Science



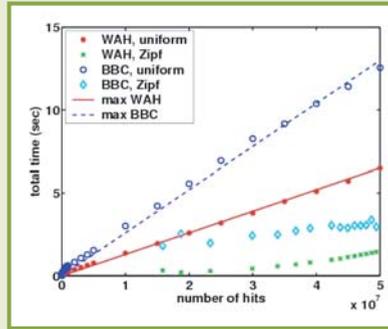
FastBit uses compressed bitmap indices to solve very large search problems that are common in data-intensive sciences

## Key features of FastBit:

- It uses an efficient compression technique called Word-Aligned Hybrid (WAH) code.
- WAH compressed bitmap indices are modest in size compared with other commonly used indices, such as B-trees.
- When answering user queries, WAH compressed bitmap indices are usually ten times as fast (at least twice) as the best known Byte-aligned Bitmap Code (BBC) compressed bitmap indices.
- On single-attribute range queries, the query response time is a linear function of the number of hits. This is optimal.
- The WAH compressed bitmap index is superior to other optimal indices (such as B+-tree index) because it can be efficiently combined to answer multi-attribute range queries.

FastBit is not only efficient in practice but also optimal in theory

Kesheng Wu, Ekow Otoo and Arie Shoshani



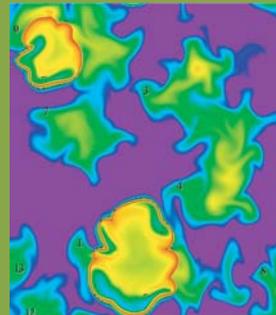
Tests confirm that the query response time is at most a linear function of the number of hits.

| 2.2 Million Records<br>12 commonly queried attributes |             | B+-tree DBMS | Projection index | Bitmap index |      |      |
|---|-------------|--------------|------------------|--------------|------|------|
|   |             |              |                  | DBMS         | BBC  | WAH  |
| Size (MB)   |             | 408          | 113              | 111          | 118  | 186  |
| Query processing time (seconds)                       | 1-attribute | 0.95         | 0.51             | 0.05         | 0.03 | 0.02 |
|   | 2-attribute | 2.15         | 0.56             | 0.39         | 0.17 | 0.04 |
|   | 5-attribute | 2.23         | 0.67             | 2.42         | 0.76 | 0.17 |

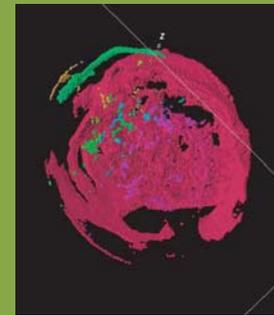
Tests on a set of high-energy physics data show that WAH compressed bitmap indices are ten times faster than the best known indices in commercial systems.

DEX: FastBit applied to region finding

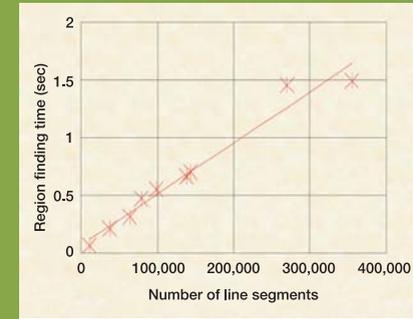
Kurt Stockinger, Kesheng Wu, John Shalf, Wes Bethel, Arie Shoshani, Wendy Koegler, and Jacqueline Chen



Flame fronts in a 2D combustion simulation.



A layer of an exploding supernova (3D simulation).

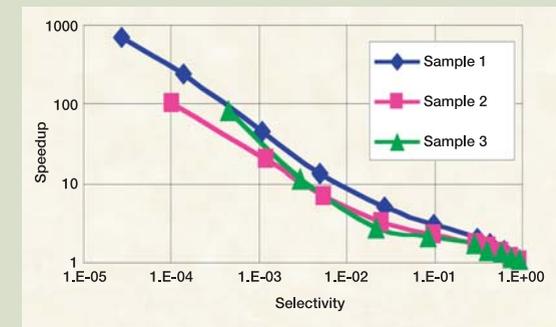
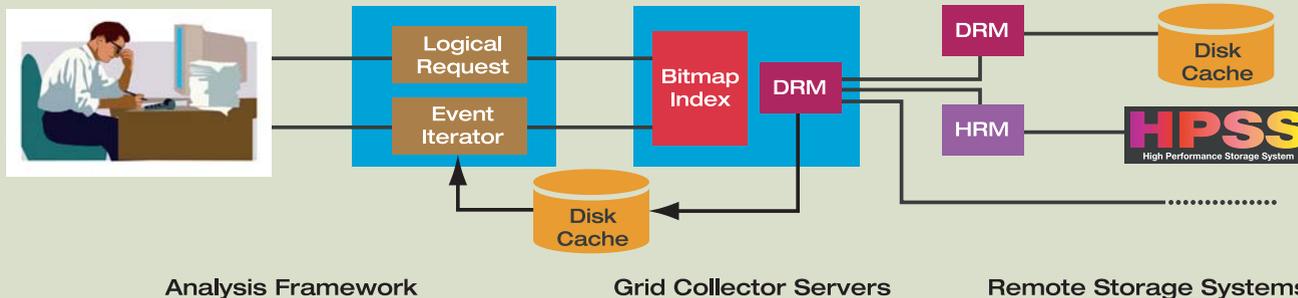


Time to identify regions in 3D Supernova simulation.

To find regions of interest, we first use FastBit to select cells and then group cells into connected regions (represented as line segments). The total time required is a linear function of the number of line segments, which is also theoretically optimal. This performance enables on-line exploration: finding regions on a 3D mesh with more than 110 million points takes less than 2 seconds.

## Grid Collector for High-Energy Physics (International Supercomputer Conference 2005 award winner)

Kesheng Wu, Junmin Gu, Jerome Lauret, Arthur M. Poskanzer, Arie Shoshani, Alexander Sim, and Wei-Ming Zhang



The lower the selectivity, the greater the speedup.

A STAR analysis job usually analyzes a fraction of the events it reads. Grid Collector allows it to read only the selected events. It not only speeds up search for rare events, but also speeds up analysis jobs that require a large number of events.