

URL	TITLE	ABSTRACT	PUBLICATION	DATE	DOI
http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=6709792	Learning and Representing Object Shape Through an Array of Orientation Columns	Recognizing an object from its background is always a very challenging task for pattern recognition, especially when the size, pose, or illumination of the object or the background are changing. The most essential method of handling this classical problem is to learn and define the structure of an object using its topological or geometrical features and components. To create a data structure that can describe the spatial relationships of object components formally and join knowledge learning and applying in a seamless loop, a representation platform must be developed. This platform can serve as a shared workspace not only for learning but also for recognition. In this paper, the platform is established by simulating the primary functional modules in the biological primary visual cortex (V1). V1 is located at the middle level of the visual information processing system. As the conjunction of low-level data and high-level knowledge, it performs visual processing for general purposes. Orientation columns in V1 are simulated in our platform, and an array of such columns is designed to represent the orientation features of edges in an image. With this platform, formalized prototypes are designed to represent each typical view of an object and thus the object concept. Data- and concept-driven processing can shift iteratively on this platform. The processes of acquiring knowledge of an object and applying that knowledge coincide with each other perfectly. The experimental results show that our algorithm can learn from a small training set and can recognize the same types of objects in natural background without any preliminary information. This bioinspired representation platform offers a promising prospect for the handling of semantic-concerned problems that need prior knowledge.	IEEE Transactions on Neural Networks and Learning Systems		10.1109/TNNLS.2013.2293178

http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=8187108	The Design and Implementation of Modern Column-Oriented Database Systems	Database system performance is directly related to the efficiency of the system at storing data on primary storage (for example, disk) and moving it into CPU registers for processing. For this reason, there is a long history in the database community of research exploring physical storage alternatives, including sophisticated indexing, materialized views, and vertical and horizontal partitioning. In recent years, there has been renewed interest in so-called column-oriented systems, sometimes also called column-stores. Column-store systems completely vertically partition a database into a collection of individual columns that are stored separately. By storing each column separately on disk, these column-based systems enable queries to readjust the attributes they need, rather than having to read entire rows from disk and discard unneeded attributes once they are in memory. The Design and Implementation of Modern Column-Oriented Database Systems discusses modern column-stores, their architecture and evolution as well the benefits they can bring in data analytics. There is a specific focus on three influential research prototypes, MonetDB, MonetDB/X100, and C-Store. These systems have formed the basis for several well-known commercial column-store implementations. Their similarities and differences are described and they are discussed in terms of their specific architectural features for compression, late materialization, join processing, vectorization and adaptive indexing (database cracking). The Design and Implementation of Modern Column-Oriented Database Systems is an excellent reference on the topic for database researchers and practitioners.	The Design and Implementation of Modern Column-Oriented Database Systems		10.1561/1900000024
---	--	---	--	--	--------------------

http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=7560435	Optimizing parallel join of column-stores on heterogeneous computing platforms	GPU and integrated multi-core CPU-GPU architecture has powerful parallel processing capability and programmable pipeline, which gradually becomes a hot area of database researches. In order to fully explore the potential abilities of heterogeneous platform, enhancing the query performance of the column-storage database. In this paper, we are taking full account of differences among system architecture based on heterogeneous platforms, firstly proposed data partition strategy for join operation on multi-thread platform-ICMD algorithm, using stream processor to process sub-space join operation in parallel. Secondly, through the implementation of query dynamic load by using task allocation model evaluation, which makes the query execution in parallel between multi-core CPU, GPU and other accelerator components. At the same time, the use of on-chip global synchronization and efficient implementation, local memory reuse optimized ICMD connection algorithm. Using SSB benchmark test, the experimental results show that based on the platform of Intel HD Graphics 4600, ICMD connection query received 1.35 speedup compared to the CPU version and received 18% performance improvement compared with GPU query engine-Ocelot.	2016 IEEE Information Technology, Networking, Electronic and Automation Control Conference		10.1109/ITNEC.2016.7560435
---	--	---	--	--	----------------------------

http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=7501946	AMC pin waveguide flange for screw redundant millimeter and submillimeter measurements	Measurements with waveguide flanges at frequencies above 100GHz have a considerable issue with leakage due to problems with achieving good electrical contact between the opposite flanges. The higher the frequency, the higher is the requirement for full contact. However, by using an artificial magnetic conducting (AMC) flange on one side of the interface, full electric contact is not needed between the two joining flanges. The AMC is realized as a pin-surface, and the leakage is stopped by a parallel-plate stopband like in gap waveguides. This paper describes how these AMC pin waveguide flanges can be used for screw redundant measurements.	2016 87th ARFTG Microwave Measurement Conference (ARFTG)		10.1109/ARFTG.2016.7501946
---	--	--	--	--	----------------------------

http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=7423087	WACO: Workload Aware Column Order for Scan Operator in Wide Table	Entering the big data era, wide table, which contains thousands of columns, is being widely adopted in cloud systems for its facilitation to critical fields such as warehouse/log analysis system, scientific applications and RDF storage. Although taking the advantage of avoiding expensive join in distributed environment, wide table puts an urgent demand for fast scan during data processing. However, represented by horizontal row-store, vertical column store and hybrid columnar store, existing promising data placement structures are witnessing a massive waste of computing resources on disk seeks, due to their ignorance of column order in physical data placement. To make the matter worse, some superior placement methods require additional adjustment to query processing engine or violate the data replication strategy and thus changed error-prone and incur additional disk seeks. In this paper, we explore a Workload Aware Column Order solution, WACO, to boost scan operator in wide table. The new WACO solution maximizes the sequential disk access and turns out to be transparent to underlying cloud systems, which therefore does not exhibit any above-mentioned shortcomings. In particular, acquire query workload, we exploit the recent access patterns on wide table and their frequencies via query logs. Given access patterns and their workload, we investigate the column placement strategy and proof that it belongs to NP-hard to figure out the optimal column layout. Furthermore, we propose a linear programming based solution to efficiently obtain an effective column order in internal physical placement. To make our solution robust and practical, we implement such scan-optimized data placement strategy as a library and thus it is a seamless integration with the underlying system and does not	2015 IEEE Conference on Collaboration and Internet Computing (CIC)		10.1109/CIC.2015.24
---	---	---	--	--	---------------------

http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=7473371	Optimization Design for Beam and Column of Steel Structure Residence	The sequential quadratic programming method is a kind of effective approach of optimization. In this paper it is used to reduce the steel consumption of the steel structure residence. Steel consumption of H-section steel beams and square steel tubular columns is taken as objective function, which has relatively larger proportion than others. Design variables are the beam height, web thickness, flange width, flange thickness and the square steel tubular column length. The strength, stiffness and stability of the steel beam and the square steel tubular column are taken as constraint condition under the relevant Building Code. By using the sequential quadratic programming method, we set up optimization mathematical model for optimizing the steel structure residence, and then obtain the optimal section size of the steel beams and the square steel tubular columns. After the structural optimization design, the consumption of steel GL1 was reduced by 10.2%, the consumption of square steel tubular column was reduced by 15.0%, compared with those of the original design. Relative to the original design, the total steel consumption was reduced by 12.6%. The approach proposed in this paper is an effective method for the optimization design of steel structure residence.	2015 8th International Conference on Intelligent Computation Technology and Automation (ICICTA)		10.1109/ICICTA.2015.155
---	--	--	---	--	-------------------------

http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=7490733	Column-stored system join optimization on coupled CPU-GPU architecture	Heterogeneous architecture is the new trend of the development of Computer system central processor unit (CPU). Taking advantage of its powerful computer power has been a new research hotspot in database system field. First, in order to enhance the query performance of column-oriented database, we propose a data partition model which is environment sensitive. The data partition model provides optimal data division for every processing unit dynamically, by monitoring the CPU occupancy rate. Then, for GPU memory access optimization, we propose a DFAT estimate model for prefetching. At the same time, we optimize GPU memory access based on coalesced access; We implement a sort-merge join algorithm on a PC with an integrated CPU-GPU chip, which adopts our data partition model and our cost model in prefetching. Our strategy is able to distribute data to different processing units automatically, and can make sort-merge join achieve a performance improvement of 33% on coupled CPU-GPU architecture.	2015 4th International Conference on Computer Science and Network Technology (ICCSNT)		10.1109/ICCSNT.2015.7490733
---	--	--	---	--	-----------------------------

http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=7382334	GBFSJ: Bloom Filter Star Join algorithms on GPUs	With the GPUs computing and storage capability continue increasing, GPUs-based parallel accelerated algorithms got many attention and success. The star join is a core and base action in relational database systems and has great impact on system performance. In this paper, we propose a massive parallelism and high scalability star join algorithm: GPUs Bloom Filter Star Join (GBFSJ). In this algorithm, fact table and dimension tables are column storage, and every column is processed separately. Star join is composed of several parallel hash join procedures. To facilitate and improve the executions of hash join on GPUs, bloom filter is used here instead of hash lookup. Bloom filter is a high space efficiency probabilistic data structure and suitable for parallel processing; each filtered dimension table build one bit vector which can be easily stored and processed. Also, GBFSJ algorithm integrate late materialization and the fact table is accessed only once, avoid large number of intermediate results and fact table multi-access. GBFSJ improves concurrency and reduce I/O operations and obtain better performance.	2015 12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)		10.1109/FSKD.2015.7382334
---	--	--	--	--	---------------------------

http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=7160266	Decomposition of natural join based on domain-interval fragmented column indices	The paper describes decomposition of natural join relational operator based on the column indices and domain-interval fragmentation. This decomposition admits parallel executing the resource-intensive relational operators without data transfers. All column index fragments are stored in main memory in compressed form to conserve space. During the parallel execution of relational operators, compressed index fragments are loaded on different processor cores. These cores unpack fragments, perform relational operator and compress fragments of partial result, which is a set of keys. Partial results are merged in the resulting set of keys. DBMS use the resulting set of keys for building the resulting table. Described approach allows efficient parallel query processing for very large databases on modern computing cluster systems with many-core accelerators. A prototype of the DBMS coprocessor system was implemented using this technique. The results of computational experiments are presented. These results confirm the efficiency of proposed approach.	2015 38th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)		10.1109/MIPRO.2015.7160266
---	--	--	--	--	----------------------------

http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=7152468	Cowic: A Column-Wise Independent Compression for Log Stream Analysis	Nowadays massive log streams are generated from many Internet and cloud services. Storing log streams consumes a large amount of disk space and incurs high cost. Traditional compression methods can be applied to reduce storage cost, but are inefficient for log analysis, because fetching relevant log entries from compressed data often requires retrieval and decompression of large blocks of data. We propose a column-wise compression approach for well-formatted log streams, where each log entry can be independently compressed or decompressed for analysis. Specifically, we separate a log entry into several columns and compress each column with different models. We have implemented our approach as a library and integrated it into two applications, a log search system and a log joining system. Experimental results show that our compression scheme outperforms traditional compression methods for decompression times and has a competitive compression ratio. For log search, our approach achieves better query times than using traditional compression algorithms for both in-core and out-of-core cases. For joining log streams, our approach achieves the same join quality with only 30% memory of uncompressed streams.	2015 15th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing		10.1109/CCGrid.2015.45
---	--	--	---	--	------------------------