Reminiscences on Influential Papers

This issue's contributors highlight their influences when it comes to their research agenda on parallel data processing and skyline queries, respectively. Enjoy reading!

While I will keep inviting members of the data management community, and neighboring communities, to contribute to this column, I also welcome unsolicited contributions. Please contact me if you are interested.

Pınar Tözün, *editor*
IT University of Copenhagen, Denmark
pito@itu.dk

Ashraf Aboulnaga
Qatar Computing Research Institute, Doha, Qatar
aaboulnaga@hbku.edu.qa

David J. DeWitt, Robert H. Gerber, Goetz Graefe, Michael L. Heytens, Krishna B. Kumar, and M. Muralikrishna.
GAMMA - A High Performance Dataflow Database Machine.
In Proceedings of the International Conference on Very Large Data Bases (VLDB), pages 228-237, 1986.

Jeffrey Dean and Sanjay Ghemawat.
MapReduce: Simplified Data Processing on Large Clusters.
In Proceedings of the Symposium on Operating Systems Design and Implementation (OSDI), pages 137-149, 2004.

The thread that links these two papers is *parallel shared-nothing dataflow query processing*.

Gamma was a highly influential parallel database system built at the University of Wisconsin in the 1980s. It introduced (or used) many ideas that were cutting-edge at the time, thus creating a template for building parallel database systems that we still use to this day. Gamma used shared-nothing parallelism and dataflow processing to improve scalability and reduce coordination between processing nodes. With that came horizontal partitioning of data, hash-based parallel algorithms for query execution, process-per-operator query scheduling, and masking node failures by data replication (which Gamma called chained declustering). The system was fully implemented on the hardware of the time, which gave the insights of the Gamma papers depth and practicality. For example, Gamma showed that disk and network bandwidth quickly become scalability bottlenecks and included optimizations to mitigate these bottlenecks.

A paper about Gamma in TKDE 1990 has the following sentence: "Gamma employs what appear today to be relatively straightforward solutions." It is remarkable that the ideas of Gamma were so influential that they were considered "straightforward" by 1990. It is even more remarkable that we still use these ideas today.

MapReduce brought the parallel shared-nothing paradigm into the era of big data. It introduced a specialized programming model that is more general than SQL yet restricted enough to allow for scalable execution. MapReduce also focused more on issues of massive scale, such as fault tolerance, execution skew, and monitoring. It is a testament to the influence of MapReduce that the ideas it introduced "appear today to be relatively straightforward" and have been used by most big data systems over the last 20 years.

At a personal level, during my PhD, I learned about parallel database systems from Gamma. And part of my research program after PhD was influenced by MapReduce. So these two papers were highly influential on me, and I still recommend them to anyone interested in high-performance data processing at scale.

Kian-Lee Tan
National University of Singapore, Singapore
tankl@comp.nus.edu.sg

Stephan Börzsönyi, Donald Kossmann, and Konrad Stocke.

The Skyline Operator.

In Proceedings of the 17th International Conference on Data Engineering (ICDE), pages 421-430, 2001.

Around late 2000/2001, an undergraduate student - Pin-Kwang Eng - came to me and wanted to pursue a PhD. I thought this would be a good time to explore a new topic. When I was looking through the list of accepted papers in ICDE 2001 for inspiration, I was attracted to a word in the title of this paper (a.k.a. Skyline). I wanted to read the paper to find out what this new operator was all about. I managed to get hold of a copy and I was not disappointed. Pin-Kwang and I went on to work on this problem and published the second "skyline" paper in the same year, and continued in this line of research on variations of the concept of skylining for a while.

Basically, a Skyline query on a multi-dimensional dataset returns a set of interesting points (i.e., the skyline) that are not dominated by any other points. Such queries are common in multi-objective optimization problems to facilitate decision making, and has wide applications in practice. A classic example is that of finding a cheap hotel that is near to the beach - in this case, the skyline points are those hotels that are no worse in both price and distance to the beach when compared to other hotels. A nice property of the operator is that there is no need to specify parameters, such as the relative importance of each dimension (as is often required by ranking methods). This paper provided a fairly comprehensive solution - besides designing the first set of skyline computation algorithms, it also showed how SQL can be extended to support the operator and discussed optimizations involving joins (e.g., to push the skyline operator through/into joins).

While multi-criteria decision making problems have been studied in other communities, this paper formulated the problem in a database setting, and opened a new research direction within the database community. It laid the groundwork for many subsequent research endeavors that studied different notions of skyline. Its influence continues till this day - just google and you will still find works citing this paper! And the database community now owned this name for this concept beautifully coined by the authors: "Skyline".