Combining Two Worlds: MonetDB with Multi-Dimensional Index Structure Support to Efficiently Query Scientific Data

Paul Blockhaus, **David Broneske**, Martin Schäler*, Veit Köppen, Gunter Saake

University of Magdeburg, *Karlsruhe Institute of Technology

SSDBM 2020, Demo-Track

Motivation

- Big data analytics and scientific workloads force DBMSs to do massive filtering
- Multi-dimensional index structures to the rescue
- None of them executed in a full-fledged DBMS

Figure 8: Throughput when executing range queries with an average selectivity of 0.38% (1 cluster) to 27.40% (20 clusters) on 1M 5-dimensional data objects using 24 software threads depending on the number of clusters.

[Sprenger et al., SSDBM 2018]

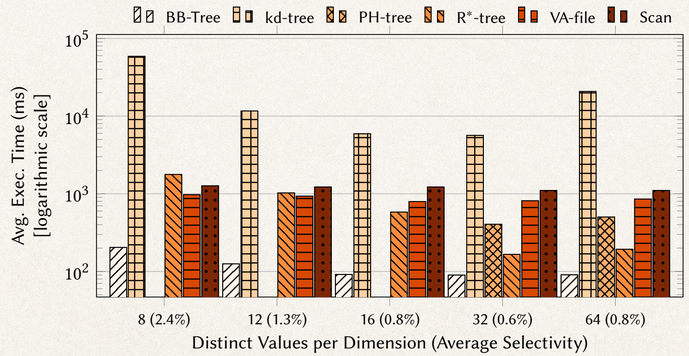
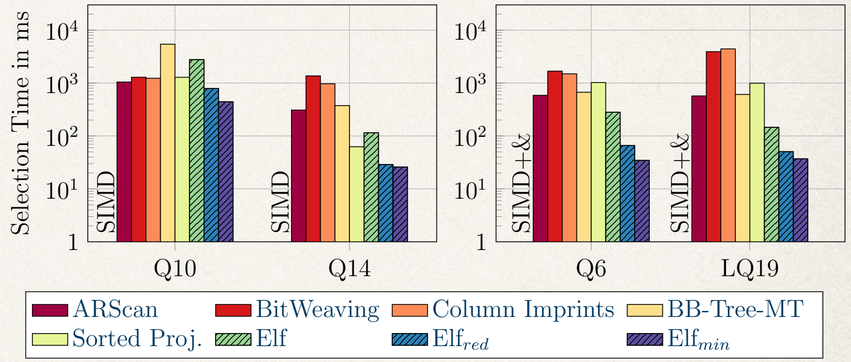


Figure 14: Performance of MDRQ with a varying selectivity depending on number of distinct values per dimension (n=10M, m=5, UNIFORM).

[Sprenger et al., EDBT 2018]



[Broneske, Phd Thesis 2019]

Elf — A Multi-Dimensional Tree Structure

Table

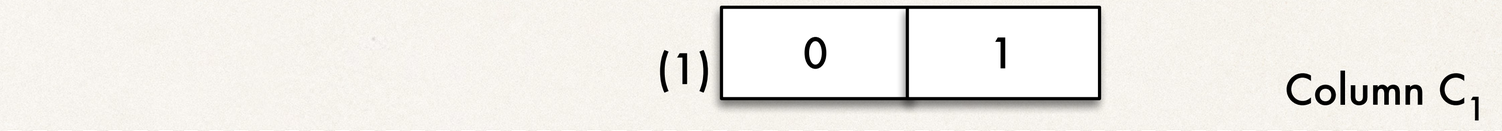
	C_1	C_2	C_3	C_4
T_1	0	1	0	1
T_2	0	2	0	0
T_3	1	0	1	0

[Broneske et al., ICDE 2017]

Elf — A Multi-Dimensional Tree Structure

Table

	C_1	C_2	C_3	C_4
T_1	0	1	0	1
T_2	0	2	0	0
T_3	1	0	1	0

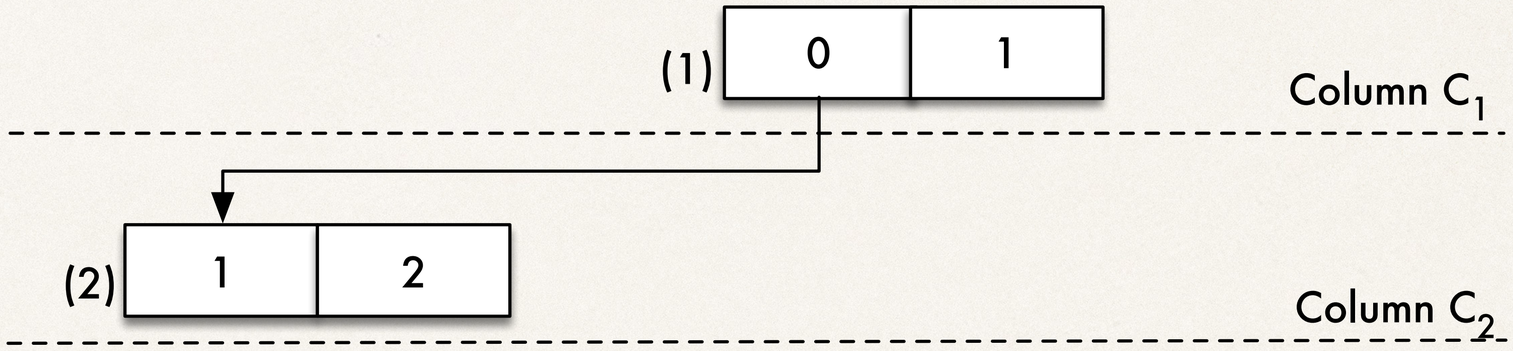


Corresponding Elf for the Table's data [Broneske et al., ICDE 2017]

Elf — A Multi-Dimensional Tree Structure

Table

	C_1	C_2	C_3	C_4
T_1	0	1	0	1
T_2	0	2	0	0
T_3	1	0	1	0

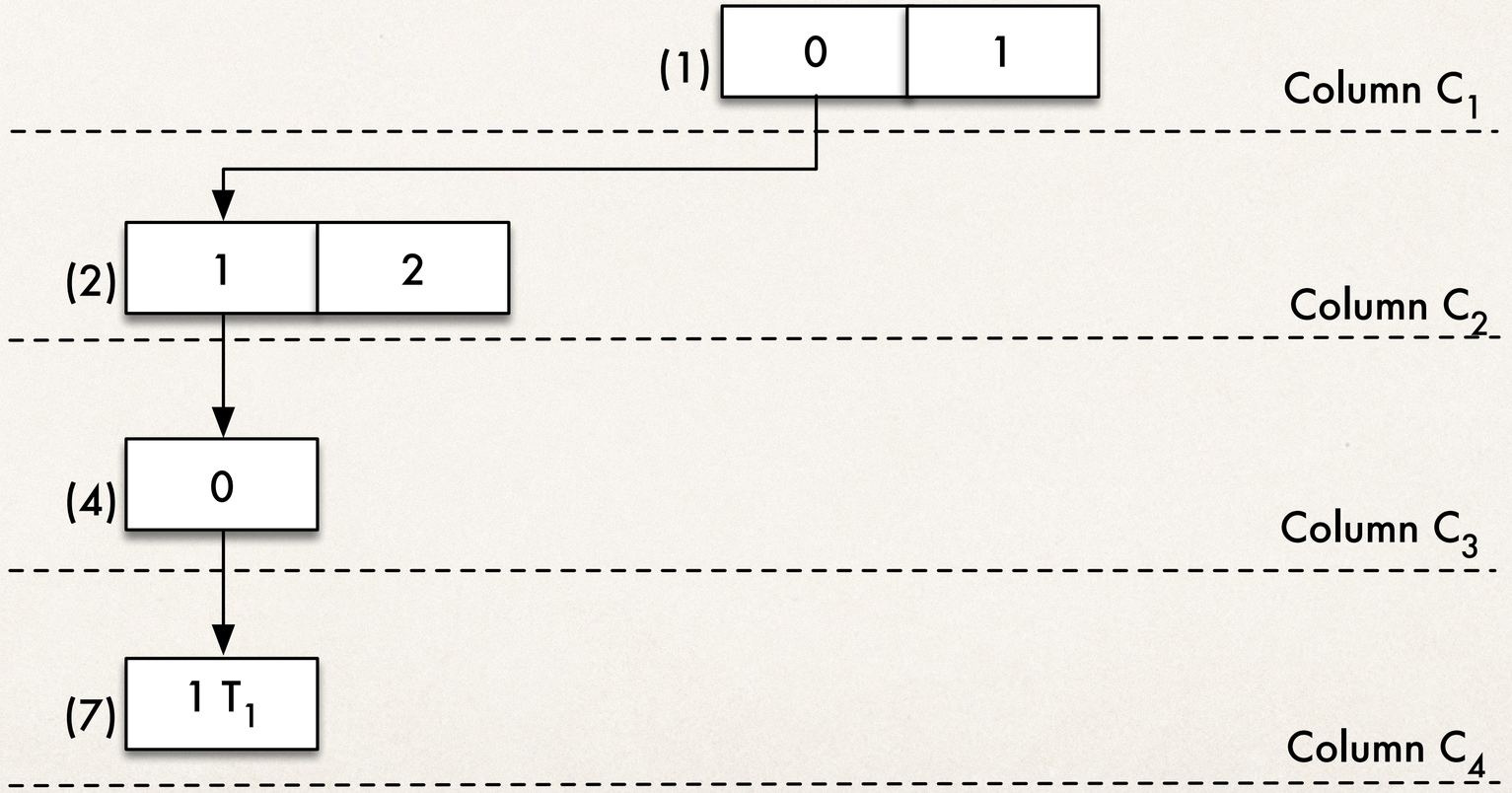


Corresponding Elf for the Table's data [Broneske et al., ICDE 2017]

Elf — A Multi-Dimensional Tree Structure

Table

	C_1	C_2	C_3	C_4
T_1	0	1	0	1
T_2	0	2	0	0
T_3	1	0	1	0

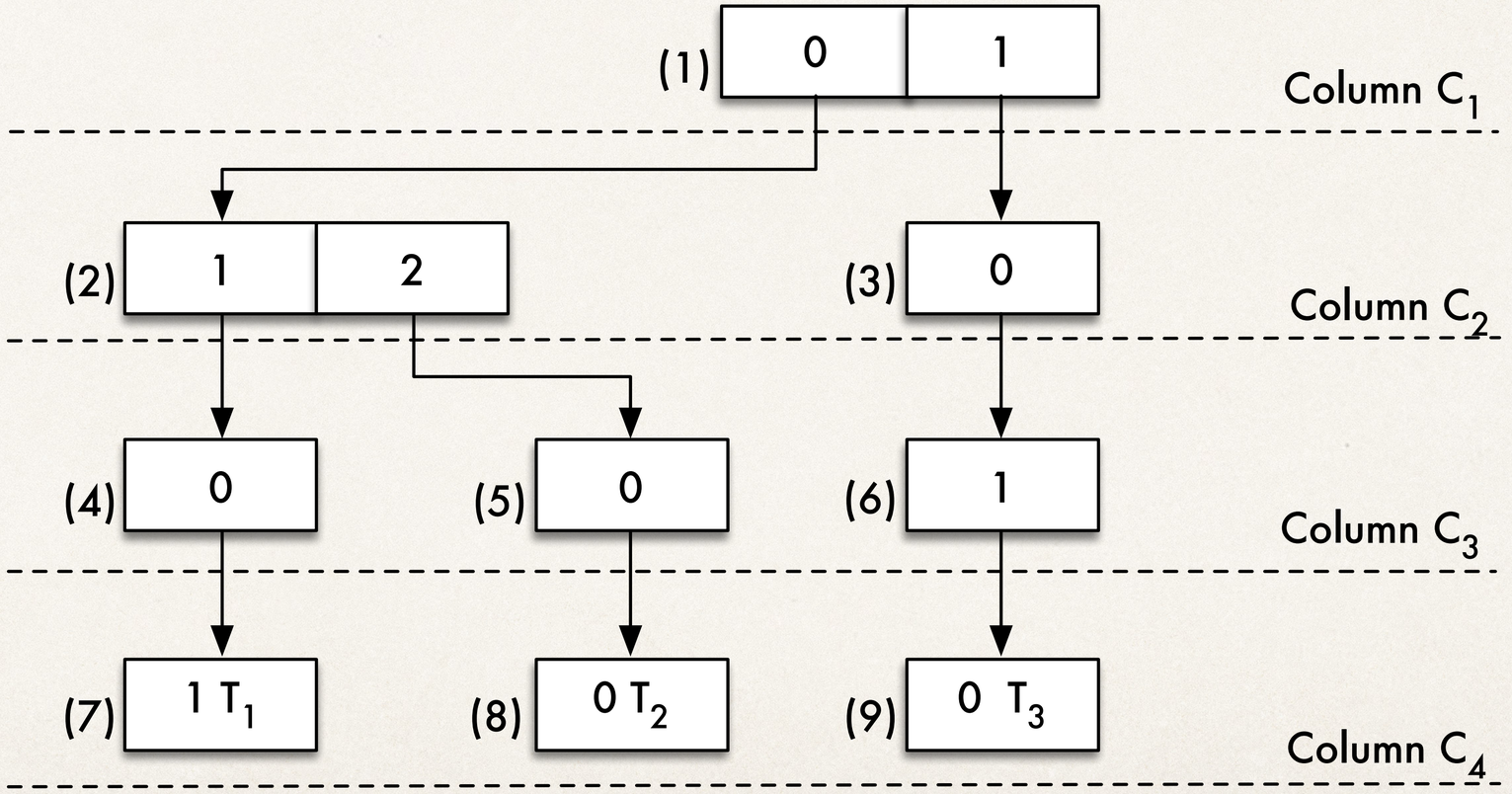


Corresponding Elf for the Table's data [Broneske et al., ICDE 2017]

Elf — A Multi-Dimensional Tree Structure

Table

	C_1	C_2	C_3	C_4
T_1	0	1	0	1
T_2	0	2	0	0
T_3	1	0	1	0

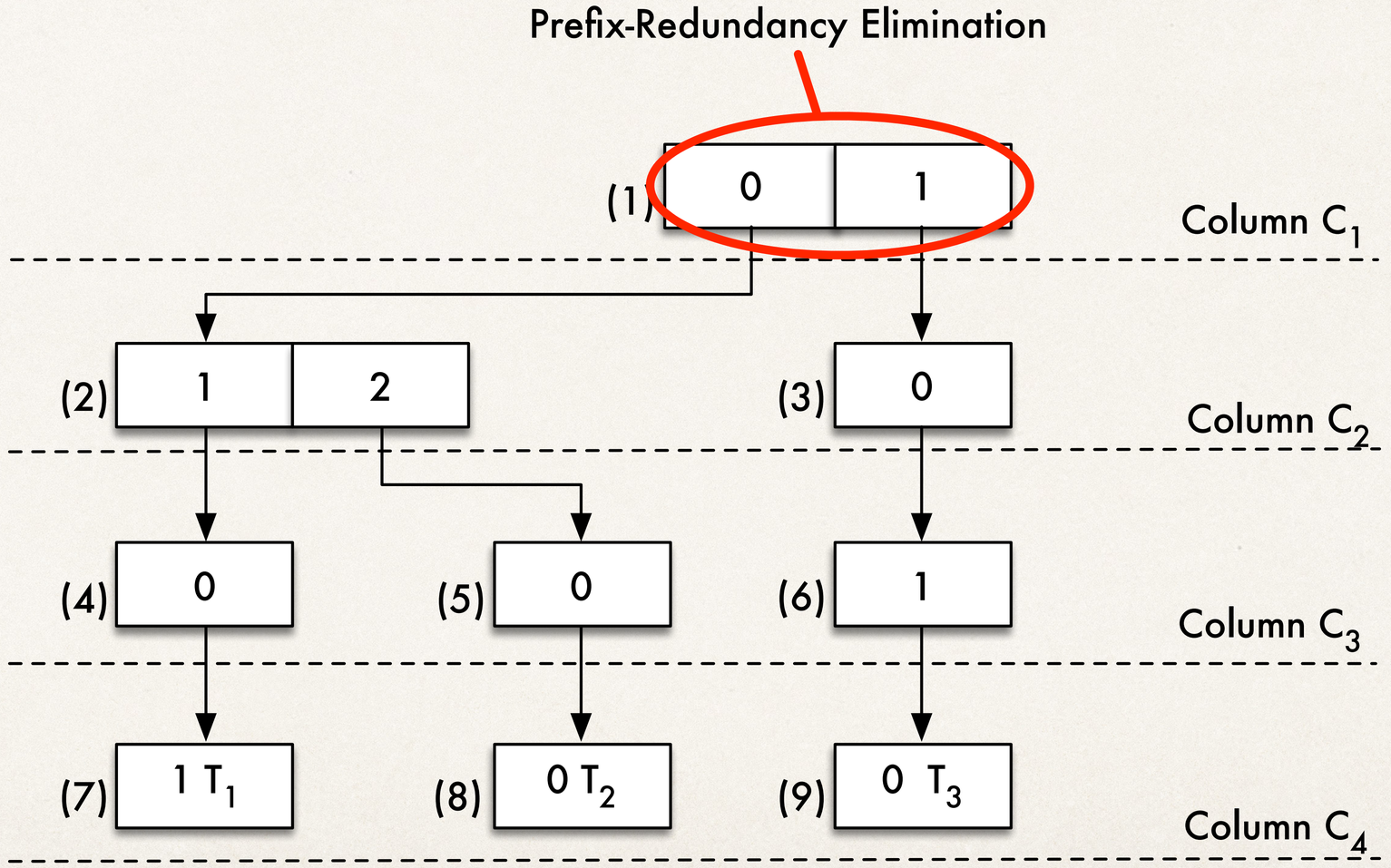


Corresponding Elf for the Table's data [Broneske et al., ICDE 2017]

Elf — A Multi-Dimensional Tree Structure

Table

	C_1	C_2	C_3	C_4
T_1	0	1	0	1
T_2	0	2	0	0
T_3	1	0	1	0



Corresponding Elf for the Table's data [Broneske et al., ICDE 2017]

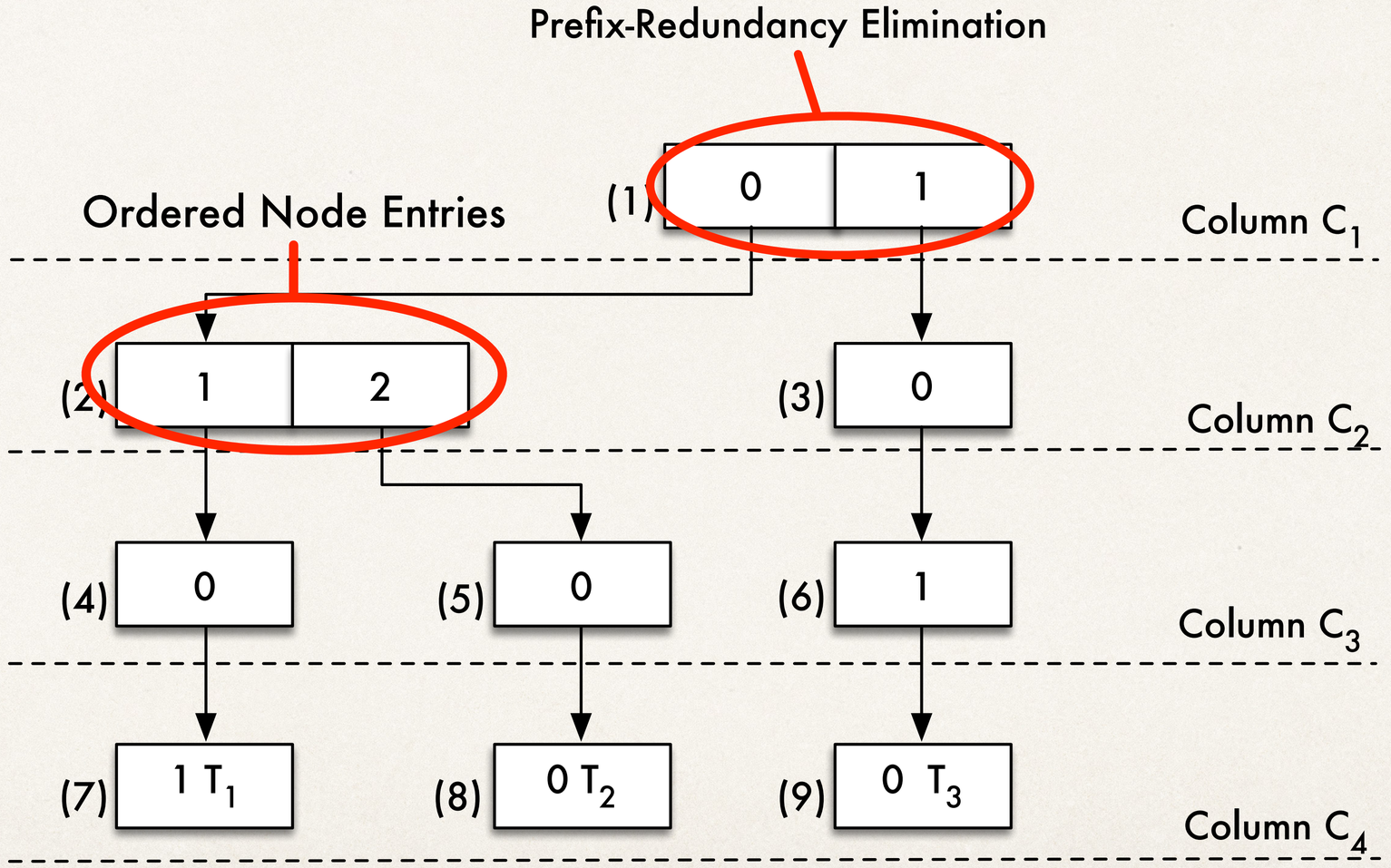
Elf — A Multi-Dimensional Tree Structure

Table

	C_1	C_2	C_3	C_4
T_1	0	1	0	1
T_2	0	2	0	0
T_3	1	0	1	0

Example Query:

```
SELECT count(*)
FROM Table
WHERE C1 < 2
AND C2 < 1
```



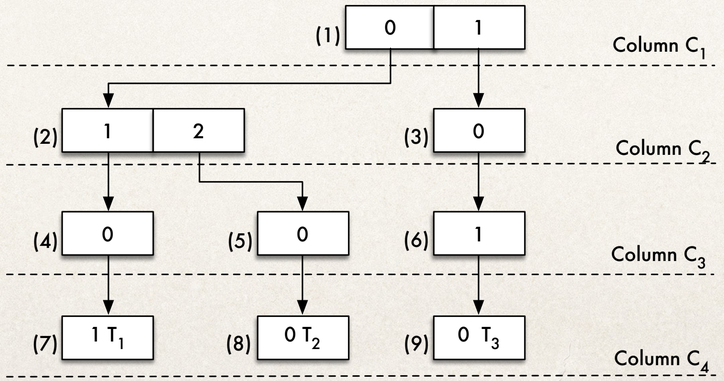
Corresponding Elf for the Table's data [Broneske et al., ICDE 2017]

DEMO

Benchmark Usage

- Try it out yourself:
  - http://www.elf.ovgu.de/SSDBM_Demo.html
  - https://git.iti.cs.ovgu.de/elf/monetelf-frontend
  - https://git.iti.cs.ovgu.de/elf/monet-elf

- Contact us:
  - david.broneske@ovgu.de
  - paul.blockhaus@st.ovgu.de



References

D. Broneske, V. Köppen, G. Saake und M. Schäler (Apr. 2017). “Accelerating multi-column selection predicates in main-memory - The Elf approach”. In: *Proceedings of the International Conference on Data Engineering (ICDE)*. IEEE, S. 647–658.

S. Sprenger, P. Schäfer, U. Leser (July 2018). „Multidimensional range queries on modern hardware“ In: *Proceedings of the International Conference on Scientific and Statistical Database Management*. ACM, S. 1–12

D. Broneske (May 2019). „Accelerating mono and multi-column selection predicates in modern main-memory database systems“. In *PhD Thesis,* University of Magdeburg.

S. Sprenger, P. Schäfer, U. Leser (Aug. 2019). „BB-Tree: A practical and efficient main-memory index structure for multidimensional workloads“. In: *Proceedings of the International Conference on Extending Database Technology (ICDE)*.