HYBRID FPGA-ACCELERATED SQL QUERY PROCESSING

Louis Woods, Zsolt István, and Gustavo Alonso

Systems Group, Dept. of Computer Science
ETH Zurich, Switzerland
email: {firstname.surname}@inf.ethz.ch

1. MOTIVATION

Ibex [?] is a novel database storage engine featuring hybrid, FPGA-accelerated query processing. The first prototype of *Ibex* has been implemented within the open-source MySQL database. In *Ibex*, an FPGA is inserted into the data path between disk and CPU to act as a query off-loading engine, operating on the stream of data towards the query processor. As a result, the volume of data hitting the CPU is substantially reduced, thereby decreasing energy consumption while increasing performance at the same time.

Main Objectives. *(i)* Show the performance impact of query off-loading, which can be measured directly in MySQL by comparing *Ibex* with other native storage engines such as MyISAM or INNODB. *(ii)* Demonstrate power savings due to query off-loading, which are observable, *e.g.*, by using an electricity meter. *(iii)* Present new approaches for evaluation of complex `WHERE` clause expressions, as well as `GROUP BY` aggregation queries in hardware.

Selection Component. To support Boolean `WHERE` clause expressions on FPGAs, the most efficient approach, so far, relied on partial reconfiguration [?]. However, parametrization is sufficient to support fairly complex expressions. The key idea is to store a *truth table* corresponding to the Boolean expression in a BRAM block. Each basic predicate is evaluated by a parameterizable circuit, and the combination of output signals is used as lookup address in the truth table.

GROUP BY Component. In contrast to previous work [?], our solution is based on a hash table, implemented using on-chip BRAM or off-chip DRAM. Thus, we do *not* require tables to be *sorted* before aggregation.

2. DEMO

The setup of the demo is depicted in Figure **??**. The MySQL database is installed on the laptop. Native MySQL tables are stored on the laptop's local SDD, whereas *Ibex* tables are stored on an equivalent (same model) SDD, which is directly connected to an XUPV5 board via SATA II. Communication between the FPGA and the laptop is based on Gigabit Ethernet for demonstration purposes.

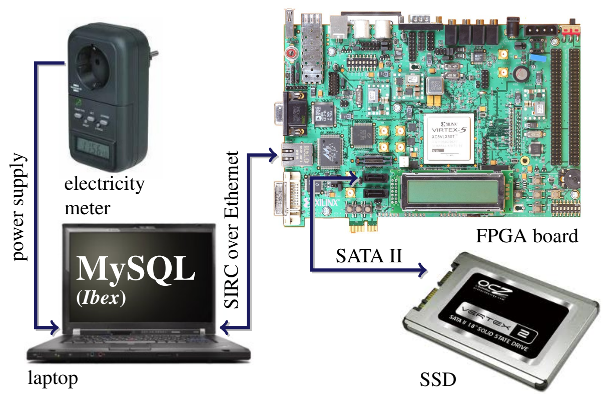


Fig. 1. Hybrid MySQL Server Setup.

Performance & Power Consumption. We compare performance and energy efficiency of *Ibex* to other storage engines that ship with MySQL. For example, running the following simple `GROUP BY` aggregation query on a 1 GB table,

```
SELECT id, sum(val) FROM table GROUP BY id
```

executes in **54.18 sec** with MyISAM, in **179.27 sec** with INNODB, and in only **3.71 sec** with *Ibex*. For 1 GB, 3.71 sec corresponds to a throughput of **270 MB/s**, close to the maximum read speed on a SATA II SSD. During query execution with MyISAM, the laptop consumes **45 watts** of power, whereas with *Ibex*, power consumption is only **31.5 watts**. The FPGA running the *Ibex* hardware is estimated to consume **2.8 watts** (Xilinx Power Analyzer tool).

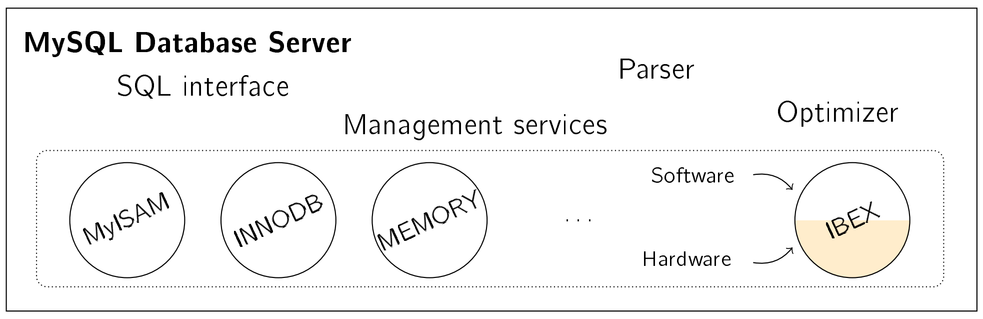
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Hybrid FPGA-Accelerated SQL Query Processing

Louis Woods, Zsolt István and Gustavo Alonso · ETH Zurich · Systems Group

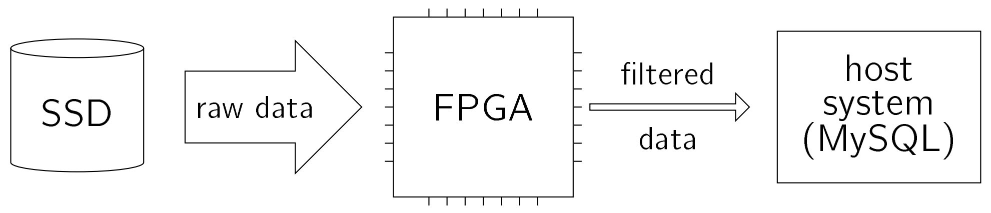
Ibex—An Intelligent Storage Engine

Ibex is a novel storage engine featuring *hybrid*, FPGA-accelerated query processing. The first prototype of *Ibex* has been implemented as a pluggable storage engine within the MySQL database. Unlike what is provided in existing data appliances with hardware acceleration, *Ibex* also supports `GROUP BY` aggregation next to the evaluation of complex `WHERE` clause expressions.



Data Path Integration of Ibex

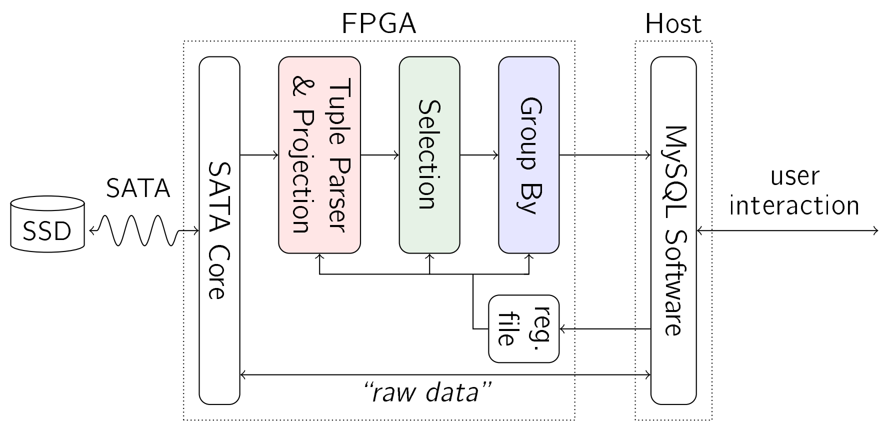
In *Ibex*, an FPGA is inserted into the data path between disk and CPU to act as a query off-loading engine operating on the stream of data towards the processing node(s).



As a result, the volume of data hitting the CPU is substantially reduced, thereby decreasing energy consumption while obtaining significant performance gains for a wide range of SQL queries.

Ibex's Hardware Engine Architecture

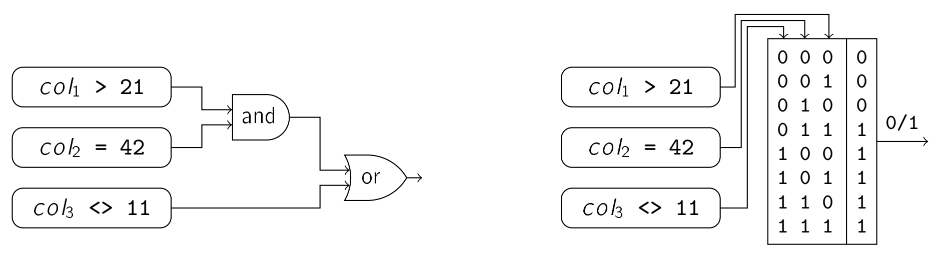
Since *Ibex* is designed for hybrid query processing, the *Ibex* hardware supports both *block-level* and *tuple-level* access to base tables.



The Selection Component

`... WHERE (col1>21 AND col2=42) or col3<>11 ...`

Hardcoded Logic Gates **Truth Table Approach**

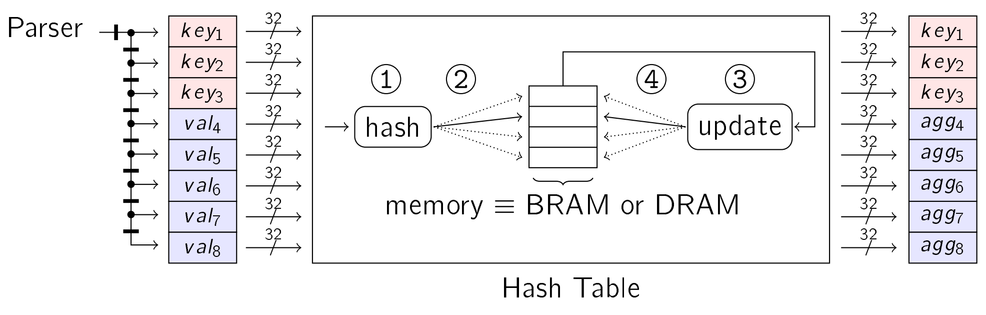


How can we support arbitrary Boolean expressions?

→ Lookup tables: Just like FPGAs simulate logic gates!
→ 1 × dual-ported BRAM block: 36 Kbit → truth table size $\leq 2^{15}$
→ Transfer time: 32 Kbit over Gigabit Ethernet ≡ only 32.8 μs

The GROUP BY Aggregation Component

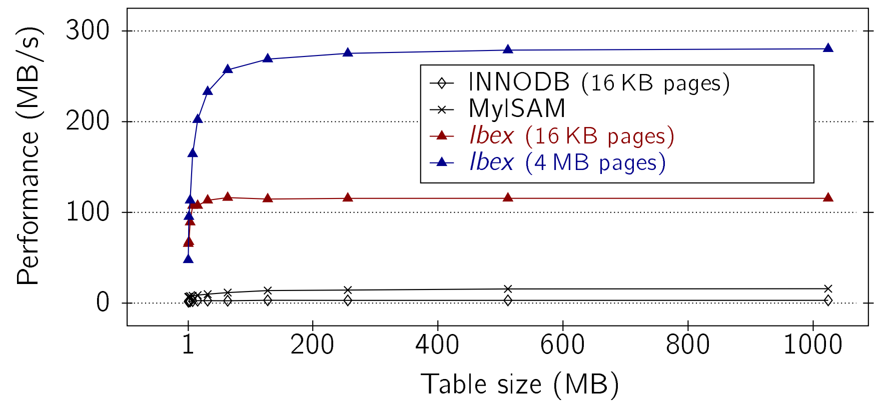
Tuple data is loaded into a wide input buffer (here, consisting of eight parameterizable, 32-bit slots) in a pipelined manner. What data is loaded is set at runtime for every query.



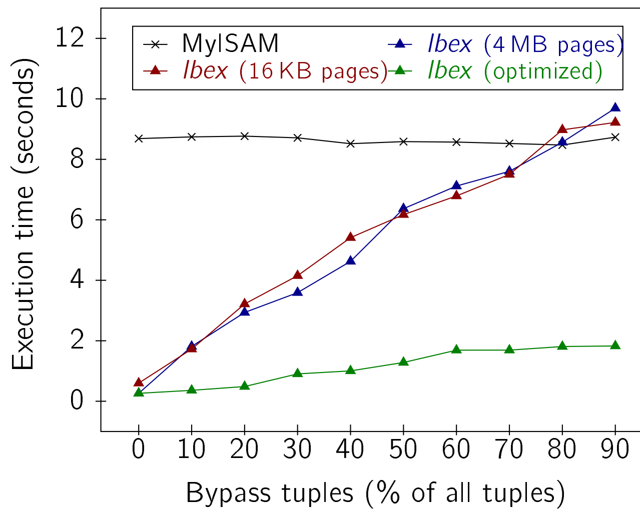
Grouping and aggregation is implemented using a hash table with the four fundamental, pipelined operations: (1) *hash*, (2) *read*, (3) *update*, and (4) *write*.

Full Off-Loading: Query Execution at Line-rate with Ibex

`GROUP BY` aggregation is *expensive* for MyISAM and INNODB. However, by fully off-loading this operator to an FPGA, we can achieve line-rate performance (SATA II ≡ 300 MB/s, but actual transfer rates of raw data are around 280 MB/s, due to protocol overhead).



Partial Off-Loading: The Impact of Bypass Tuples



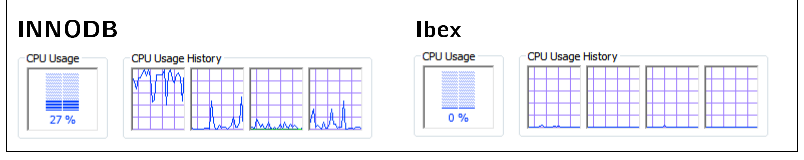
Instead of resolving hash collisions on the FPGA, we *bypass* colliding tuples to the host. In the case of *bypass tuples*, we need to do a separate `GROUP BY` and aggregation step in software.

Performance of MyISAM (⋆) is reached at approx. 80% bypasses. This can be improved by dealing with bypasses natively (▲) in the software part of *Ibex*.

Energy Efficiency (Workstation)

Query/Storage Engine	Δ-Power	Energy Consumption
Hybrid Join / MyISAM	22 watts	900 joules
Hybrid Join / INNODB	24 watts	7740 joules
Hybrid Join / *Ibex*	5 watts	288 joules
Group By / MyISAM	21 watts	3528 joules
Group By / INNODB	23 watts	6624 joules
Group By / *Ibex*	3 watts	216 joules

CPU usage when executing `GROUP BY`


INNODB
Ibex
CPU Usage
CPU Usage History
27 %
0 %