

Data Processing on the fast lane

Gustavo Alonso

Systems Group

Department of Computer Science

ETH Zurich, Switzerland

The team behind the work:

- Rene Müller (now at IBM Almaden)
- Louis Woods (now at Apcera)
- Jens Teubner (now Professor at TU Dortmund)

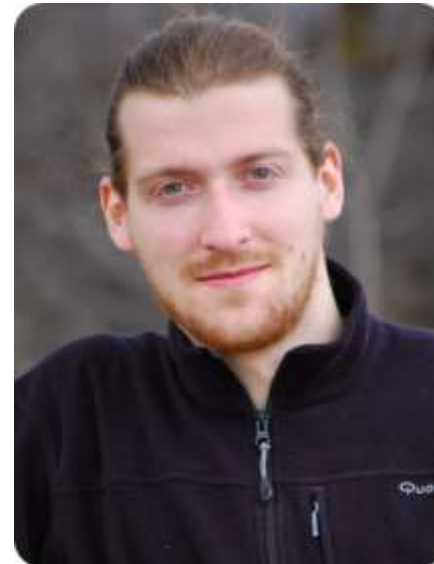
David Sidler



Muhsen Owaida



Zsolt Istvan



Kaan Kara










Data processing today:
Appliances
Data Centers (Cloud)

What is a database engine?

- As complex or more complex than an operating system
- Full software stack including
 - Parsers, Compilers, Optimizers
 - Own resource management (memory, storage, network)
 - Plugins for application logic
 - Infrastructure for distribution, replication, notifications, recovery
 - Extract, Transform, and Load infrastructure
- Large legacy, backward compatibility, standards
- Hugely optimized

Databases are
blindly fast at
what they do
well

Date Submitted	Company	System	Performance (tpmC)	Price/tpmC	V
11/25/14		Dell PowerEdge T620	112,890	.19 USD	
03/26/13		SPARC T5-8 Server	8,552,523	.55 USD	
02/22/13		IBM System x3650 M4	1,320,082	.51 USD	
09/27/12		Cisco UCS C240 M3 Rack Server	1,609,186	.47 USD	
04/11/12		IBM Flex System x240	1,503,544	.53 USD	
03/27/12		Sun Server X2-8	5,055,888	.89 USD	
01/17/12		Sun Fire X4800 M2 Server	4,803,718	.98 USD	

Databases = think big

ORACLE EXADATA

From Oracle documentation

Database Grid

- 8 Dual-processor x64 database servers

OR

- 2 Eight-processor x64 database servers

InfiniBand Network

- Redundant 40Gb/s switches
- Unified server & storage network



Intelligent Storage Grid

- 14 High-performance low-cost storage servers



- 100 TB **High Performance** disk, or
336 TB **High Capacity** disk

- 5.3 TB PCI Flash

- Data mirrored across storage servers

Database engine trends: Appliances

Oracle:

T7, SQL in Hardware,
RAPID

SAP:

OLTP+OLAP on main memory
Hana on SGI supercomputer

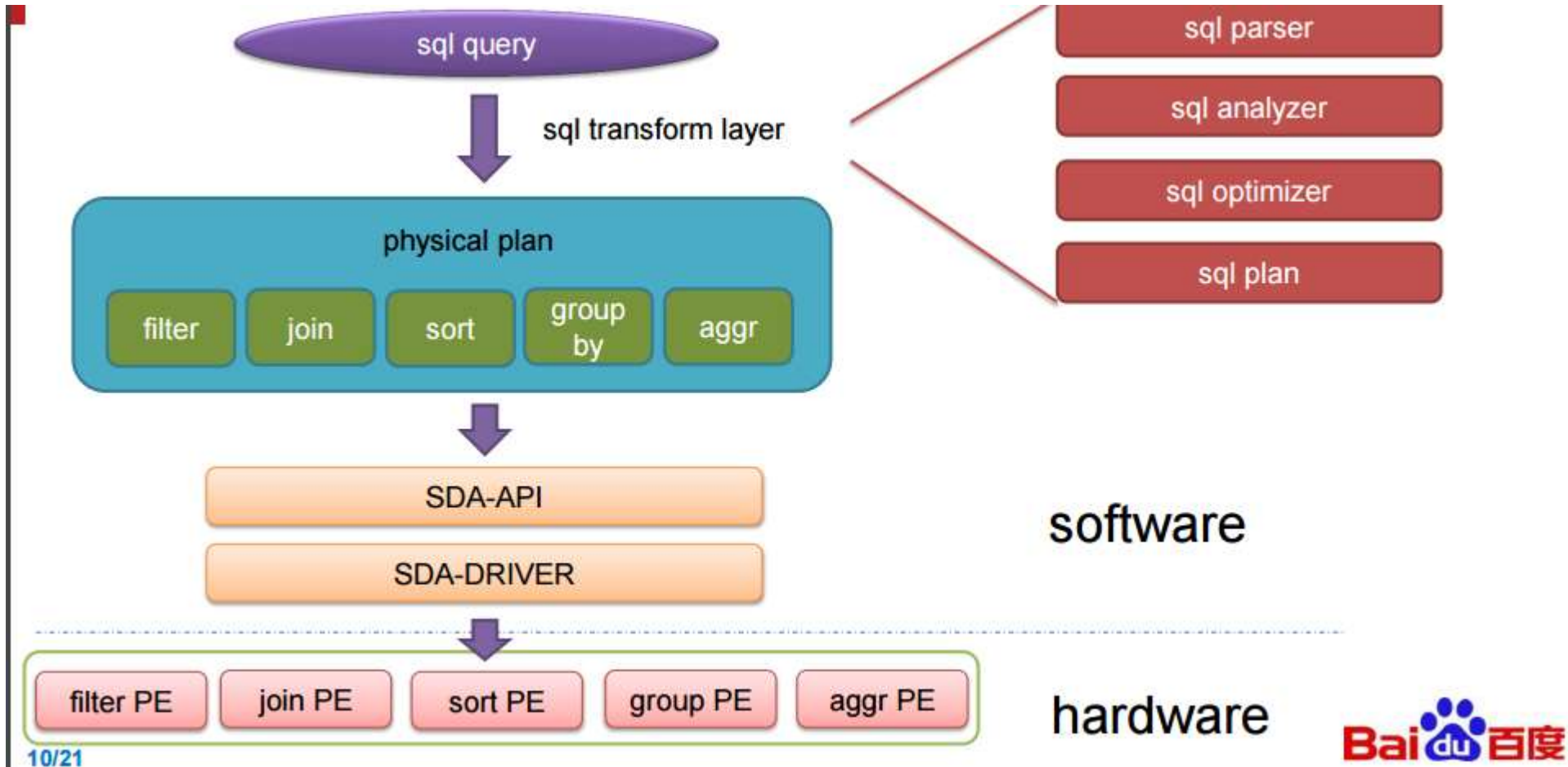


SAP Hana on SGI UV 300H
SGI documentation

Nobody ever got fired for using Hadoop on a Cluster

A. Rowstron, D. Narayanan, A. Donnelly, G. O'Shea, A. Douglas
HotCDP 2012, Bern, Switzerland

SQL on FPGAs



10/21

Presentation at HotChips'16 from Baidu

<http://www.nextplatform.com/2016/08/24/baidu-takes-fpga-approach-accelerating-big-sql/>

The challenge of hardware
acceleration

If it sounds too good to be true ..

Debunking the 100X GPU vs. CPU Myth: An Evaluation of Throughput Computing on CPU and GPU

Victor W Lee[†], Changkyu Kim[†], Jatin Chhugani[†], Michael Deisher[†],
Daehyun Kim[†], Anthony D. Nguyen[†], Nadathur Satish[†], Mikhail Smelyanskiy[†],
Srinivas Chennupaty^{*}, Per Hammarlund^{*}, Ronak Singhal^{*} and Pradeep Dubey[†]

victor.w.lee@intel.com

[†]Throughput Computing Lab,
Intel Corporation

^{*}Intel Architecture Group,
Intel Corporation

ABSTRACT

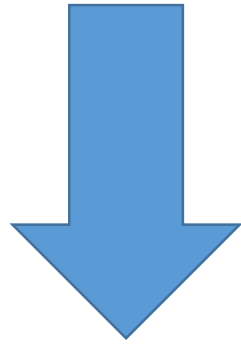
Recent advances in computing have led to an explosion in the amount of data being generated. Processing the ever-growing data in a timely manner has made throughput computing an important aspect for emerging applications. Our analysis of a set of important throughput computing kernels shows that there is an ample amount of parallelism in these kernels which makes them suitable for today's multi-core CPUs and GPUs. In the past few years there have been many studies claiming GPUs deliver substantial speedups (between 10X and 1000X) over multi-core CPUs on these kernels. To

The past decade has seen a huge increase in digital content as more documents are being created in digital form than ever before. Moreover, the web has become the medium of choice for storing and delivering information such as stock market data, personal records, and news. Soon, the amount of digital data will exceed exabytes (10^{18}) [31]. The massive amount of data makes storing, cataloging, processing, and retrieving information challenging. A new class of applications has emerged across different domains such as database, games, video, and finance that can process this huge amount of data to distill and deliver appropriate content to

Usual unspoken caveats in HW acceleration

- Where is the data to start with?
- Where does the data has to be at the end?
- What happens with irregular workloads?
- What happens with large intermediate states?
- What is the architecture?
- Is the design preventing the system from doing something else?
- Can the accelerator be multithreaded?
- Is the gain big enough to justify the additional complexity?
- Can the gains be characterized?

Do not replace, enhance



Help the CPU to do what
it does not do well

Text search in databases

3 Queries with increasing complexity:

```
Q1:    SELECT count(*) FROM address_table
WHERE  addr_string LIKE '%Strasse%';
```

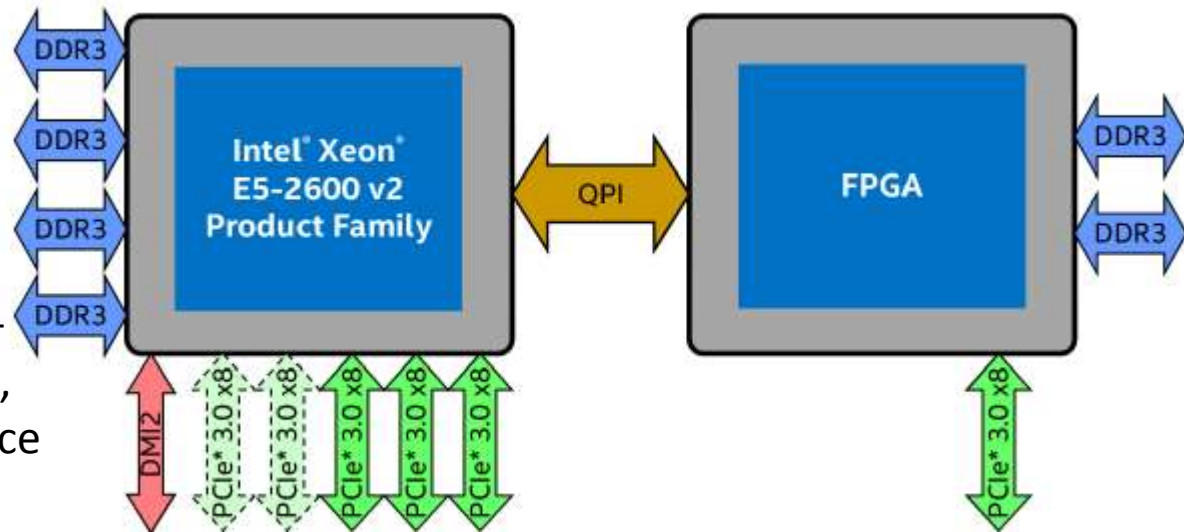
```
Q2:    SELECT count(*) FROM address_table
WHERE  addr_string LIKE '%Alan%Turing%Cheshire%';
```

```
Q3:    SELECT count(*) FROM address_table
WHERE  REGEXP_LIKE(addr_string, '(Strasse|Str\.)
.*(8[0-9]{4})');
```

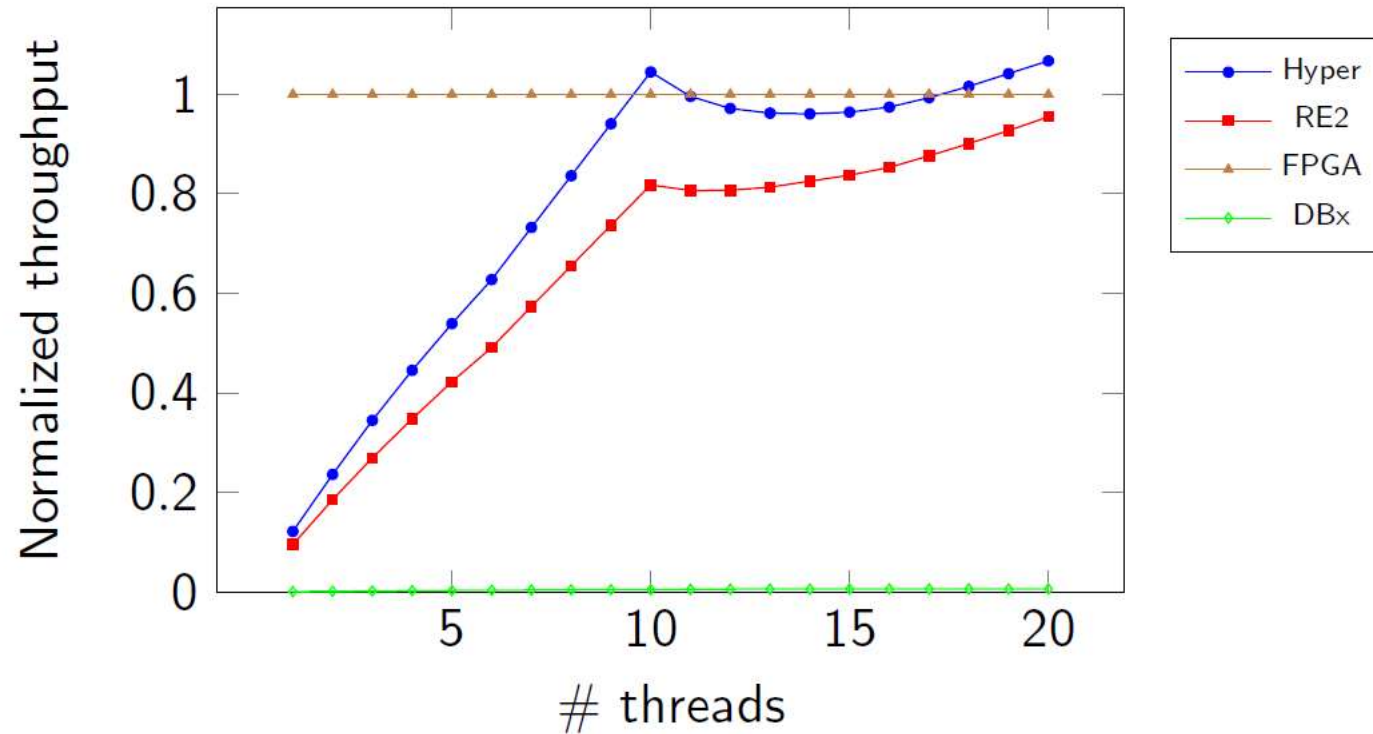
FCCM'16

INTEL HARP:

This is an experimental system provided by Intel any results presented are generated using pre-production hardware and software, and may not reflect the performance of production or future systems.

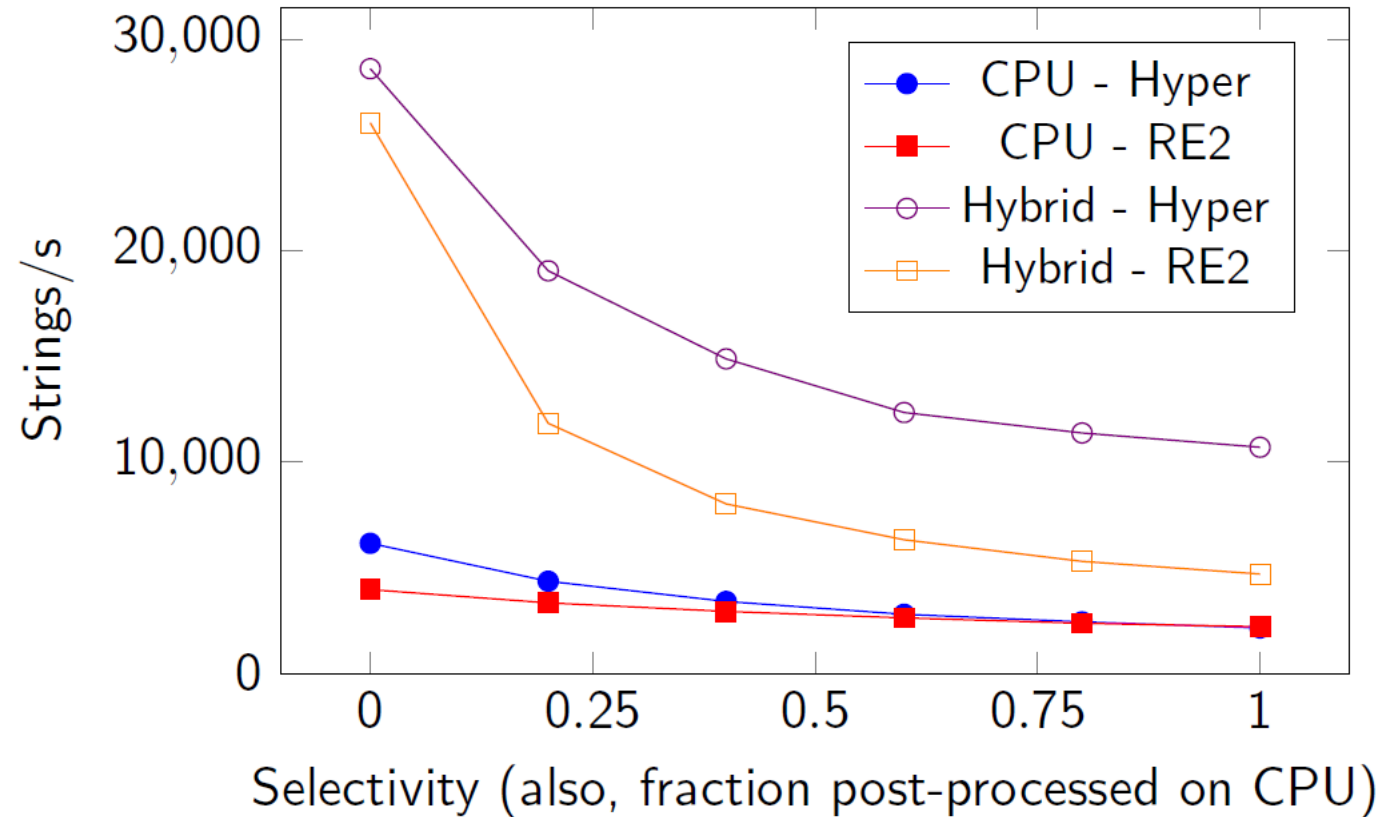


100% processing on FPGA



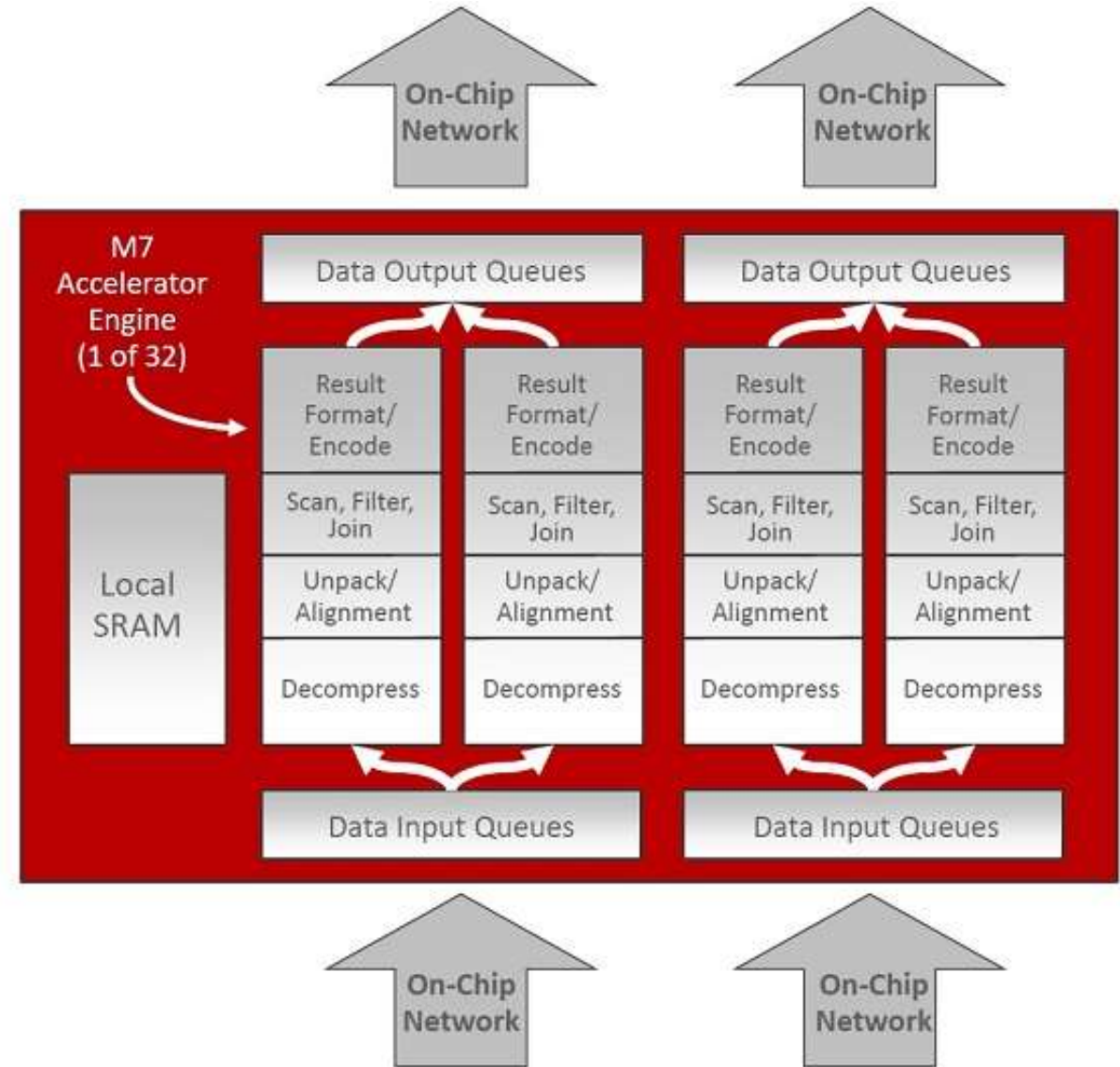
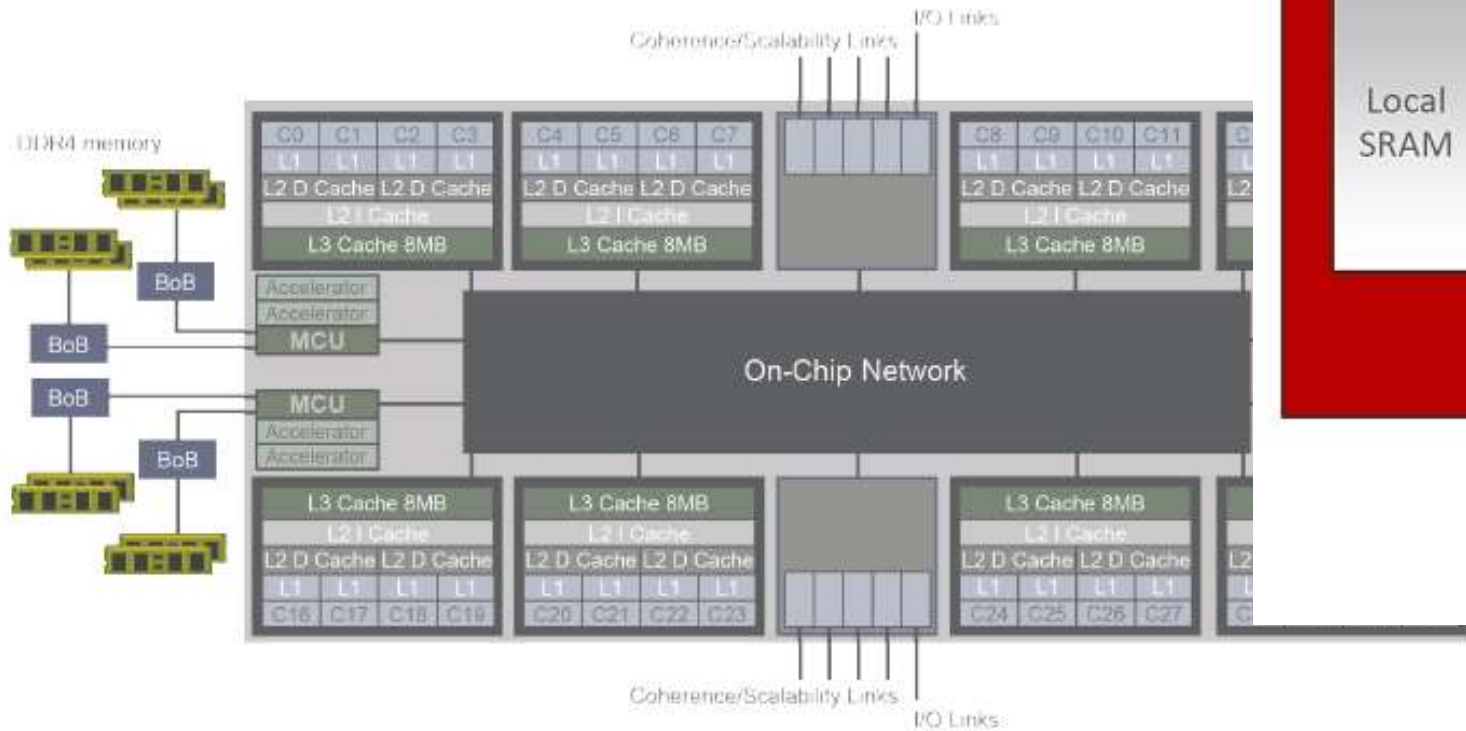
Q3: WHERE REGEXP_LIKE(address_string, '(Strasse|Str\..)*(8[0-9]4)')

Hybrid Processing CPU/FPGA



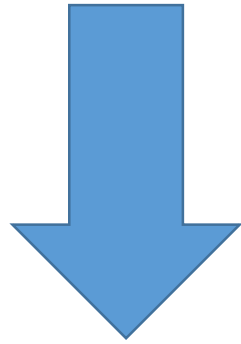
Regular expression: '(Strasse|Str\..)*(8[0-9]4).*delivery'

Accelerators to come

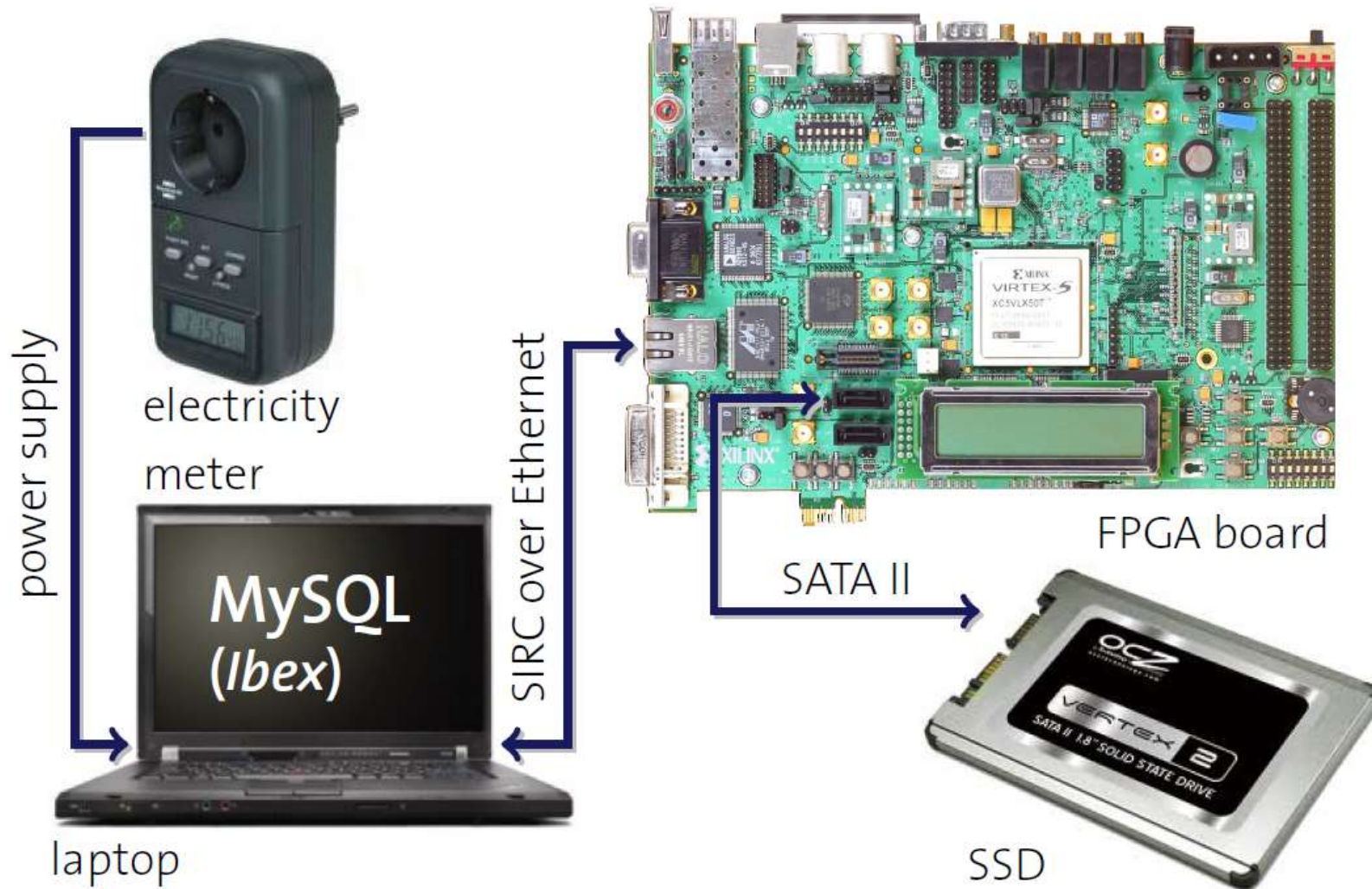


From Oracle M7 documentation

If the data moves, do it
efficiently



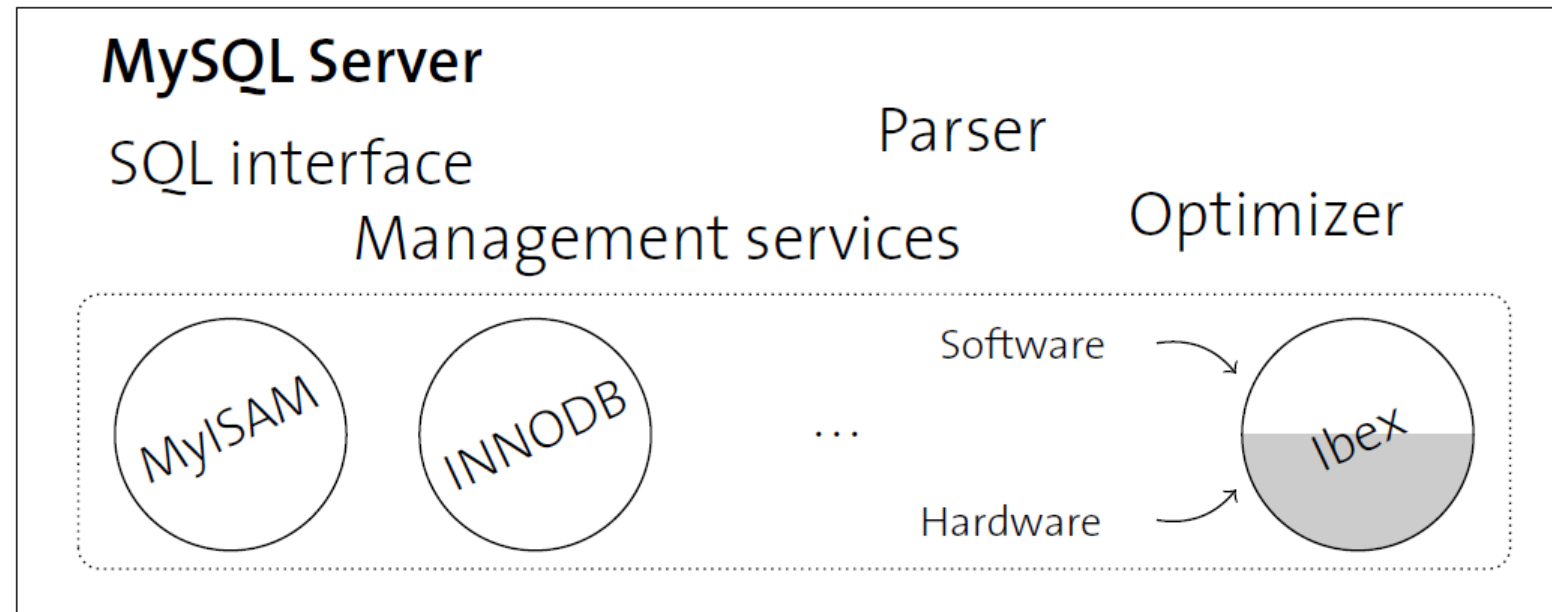
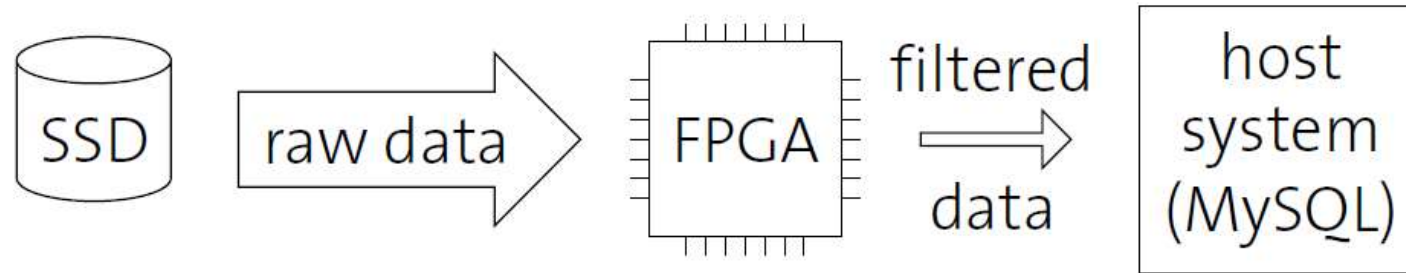
Bumps in the wire(s)



(Woods, VLDB'14)

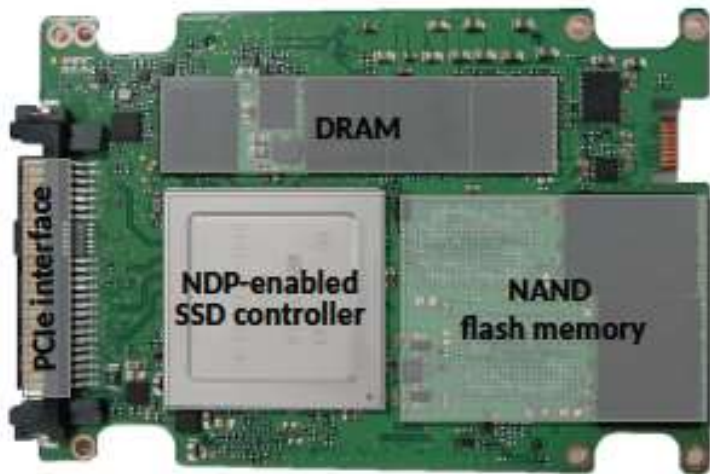
IBEX

A processor on the data path



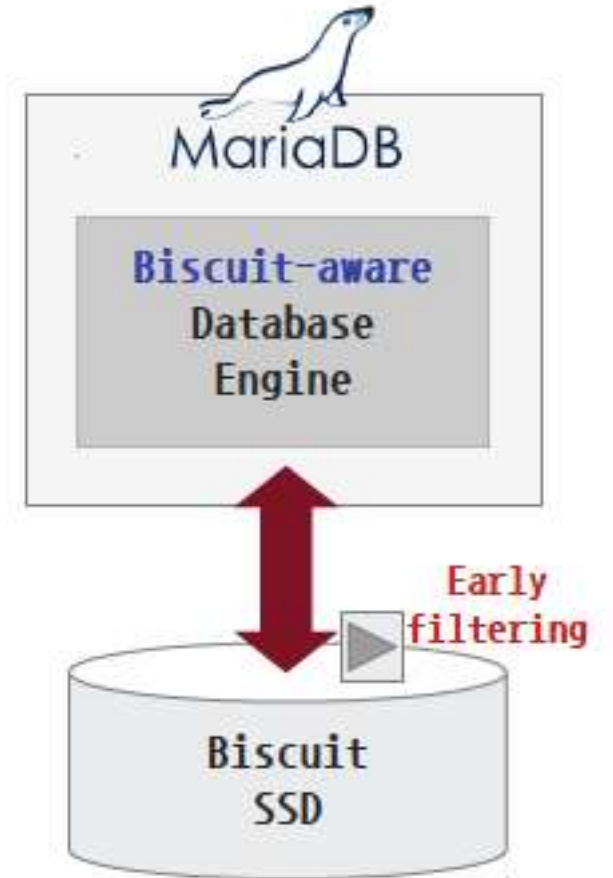
Storage to come

- Recent example BISCUIT from Samsung (ISCA'16)
 - User programmable Near-Data Processing for SSDs



[Inside of PM1725]

Item	Description
Host interface	PCIe Gen.3 x4 (3.2 GB/s)
Protocol	NVMe 1.1
Device density	1 TB
SSD architecture	Multiple channels/ways/cores
Storage medium	Multi-bit NAND flash memory
Compute resources for Biscuit	Two ARM Cortex R7 cores @750MHz with MPU
On-chip SRAM	< 1 MiB
DRAM	≥ 1 GiB



From Samsung presentation at ISCA'16

<http://isca2016.eecs.umich.edu/wp-content/uploads/2016/07/3A-1.pdf>

Sounds good?

The goal is to be able to do this at all levels:

Smart storage

On the network switch (SDN like)

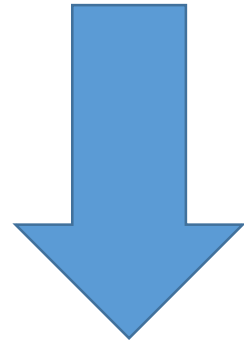
On the network card (smart NIC)

On the PCI express bus

On the memory bus (active memory)

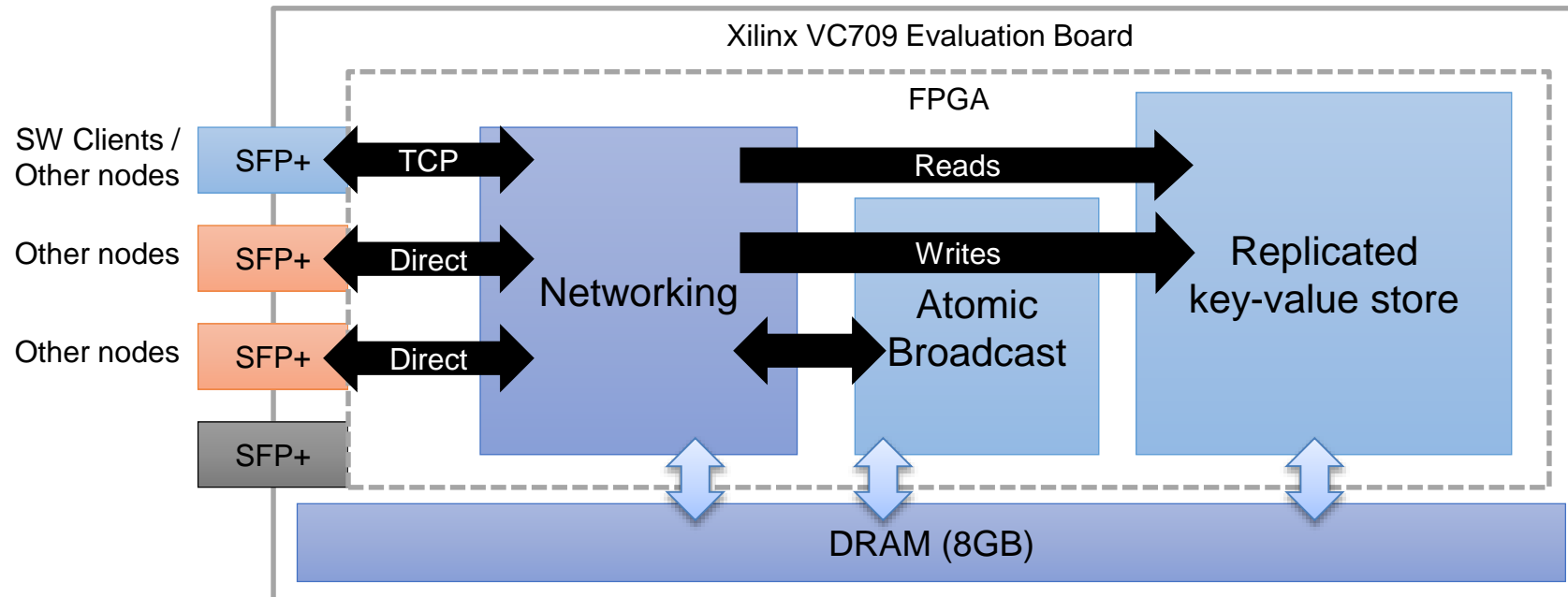
**Every element in the system
(a node, a computer rack, a cluster)
will be a processing component**

Disaggregated data center

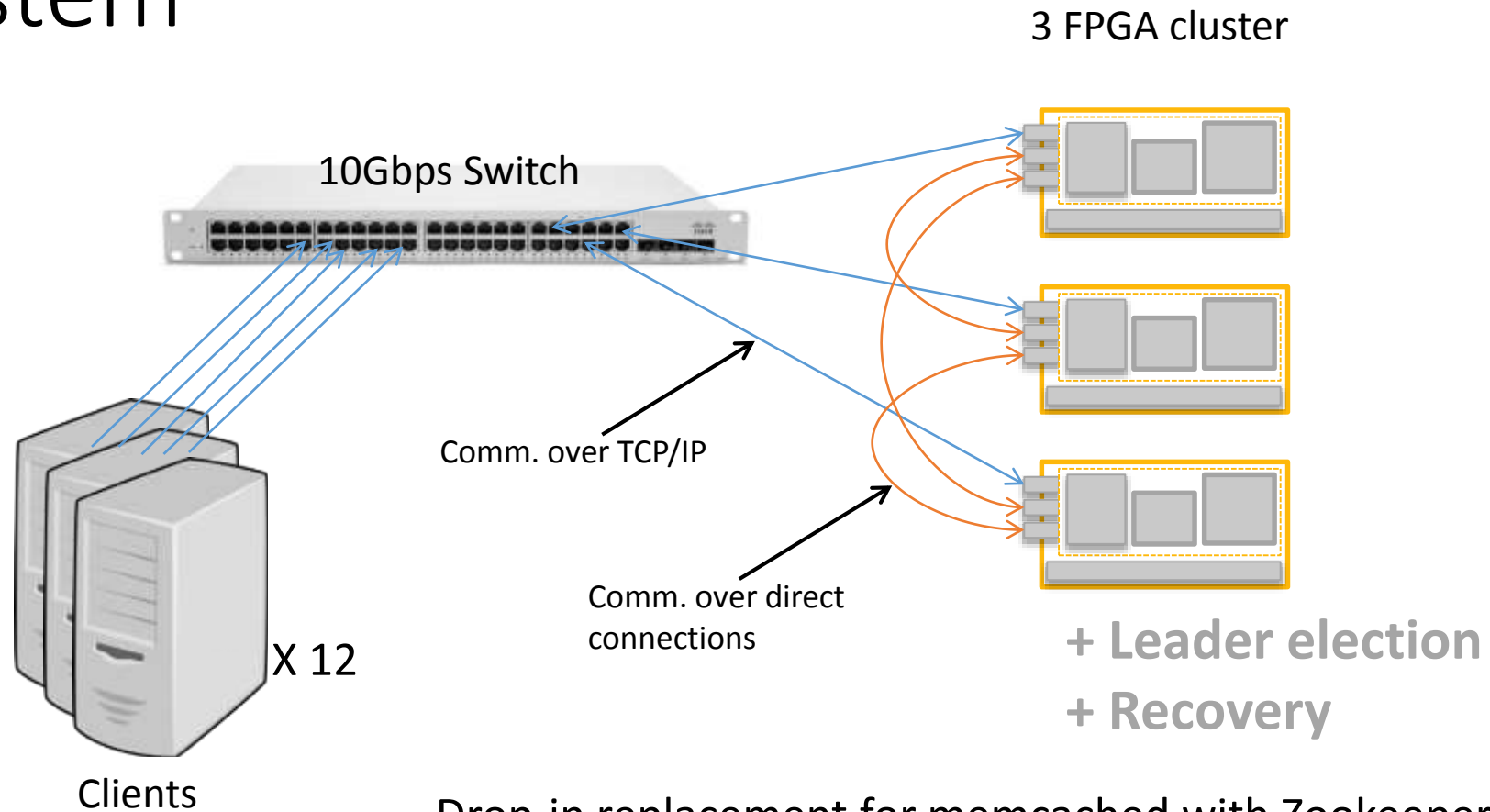


Near Data Computation

Consensus in a Box (Istvan et al, NSD'16)



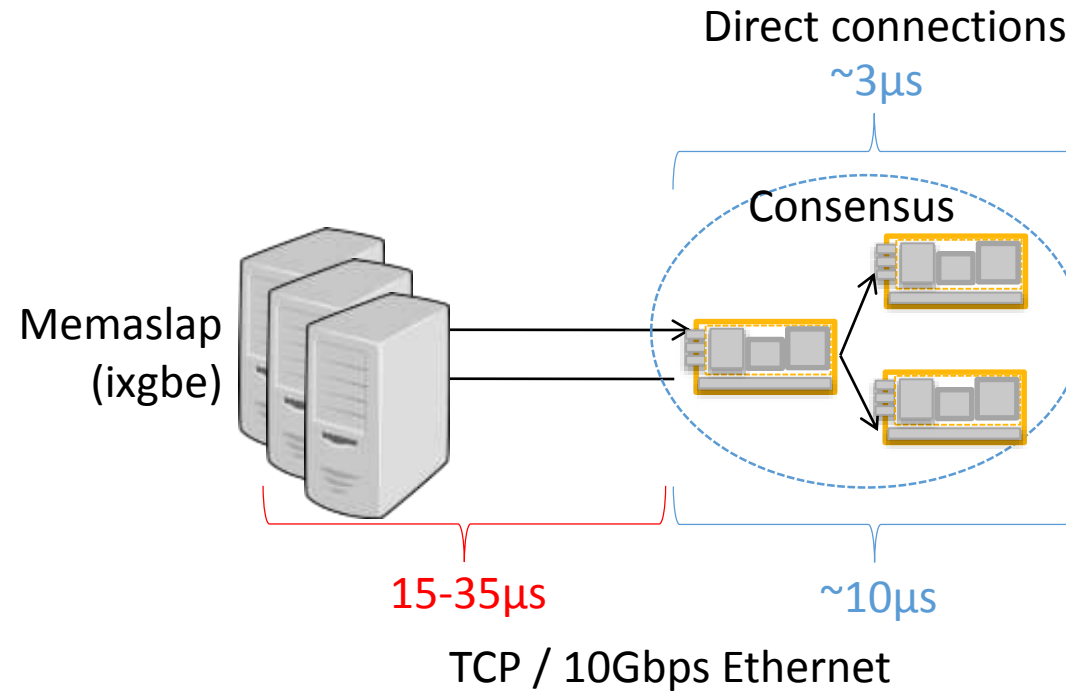
The system



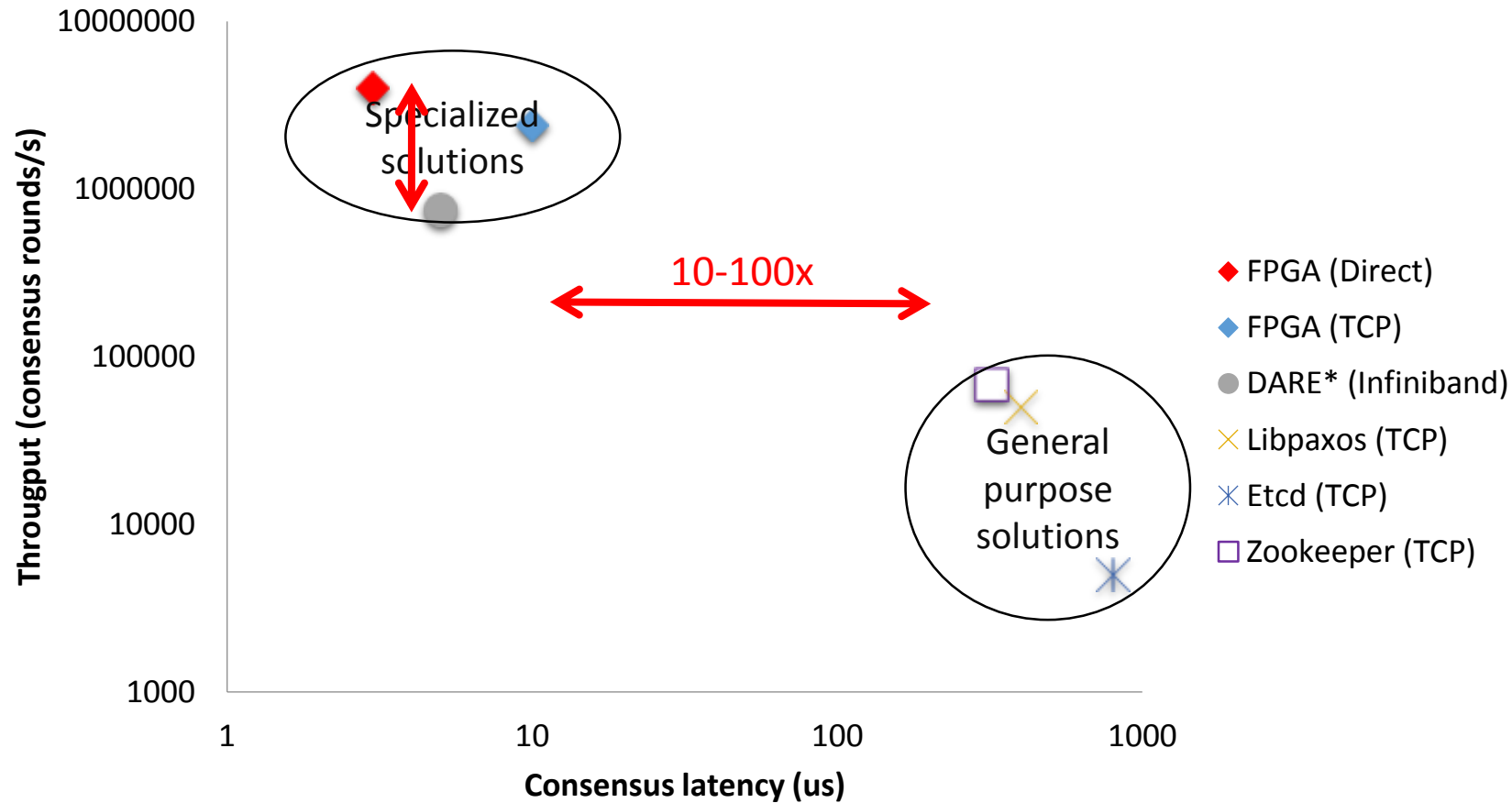
Drop-in replacement for memcached with Zookeeper's replication

- Standard tools for benchmarking (libmemcached)
 - Simulating 100s of clients

Latency of puts in a KVS



The benefit of specialization...



[1] Dragojevic et al. FaRM: Fast Remote Memory. In NSDI'14.

[2] Poke et al. DARE: High-Performance State Machine Replication on RDMA Networks. In HPDC'15.

*=We extrapolated from the 5 node setup for a 3 node setup.

This is the end ...

Most exciting time to be in research
Many opportunities at all levels and in all areas

FPGAs great tools to:
Explore parallelism
Explore new architectures
Explore Software Defined X/Y/Z
Prototype accelerators

FPGAs: the view from an outsider

Difficulty to program

- FPGAs are no more difficult to program than system software (OS, databases, infrastructure, etc.)
- Only a handful of programmers can do system software, my guess is system programmers are not many more than the people who can program FPGAs
- But FPGAs have no tools to enhance productivity, specially no freely available tools (GCC, instrumentation, libraries, open source tools ...)

CS vs EE

- EE = understand parallelism
- CS= understand abstraction

You need both (and these days a lot more: systems, algorithms, machine learning, data center architecture, ...)

Complete systems

- The proof of something that makes a difference is an end to end argument
- Showing that something is faster when running on an FPGA does not mean it will be faster when hooked into a real system (example: GPUs)