ETH zürich

Consensus in a Box
Inexpensive Coordination in Hardware

Zsolt István, David Sidler, Gustavo Alonso, Marko Vukolic[*]

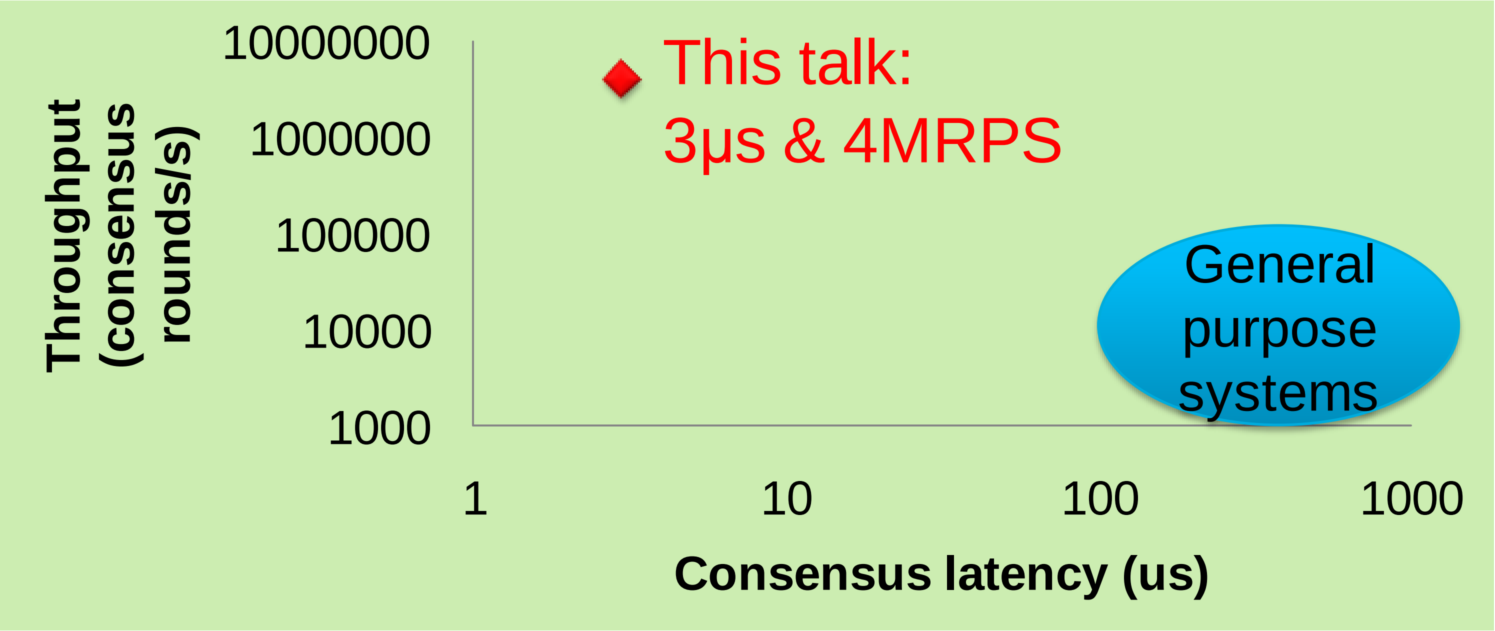
Systems Group, Department of Computer Science, ETH Zurich

[*]IBM Research, Zurich

Systems@ETH zürich

Motivation: Cost of consensus

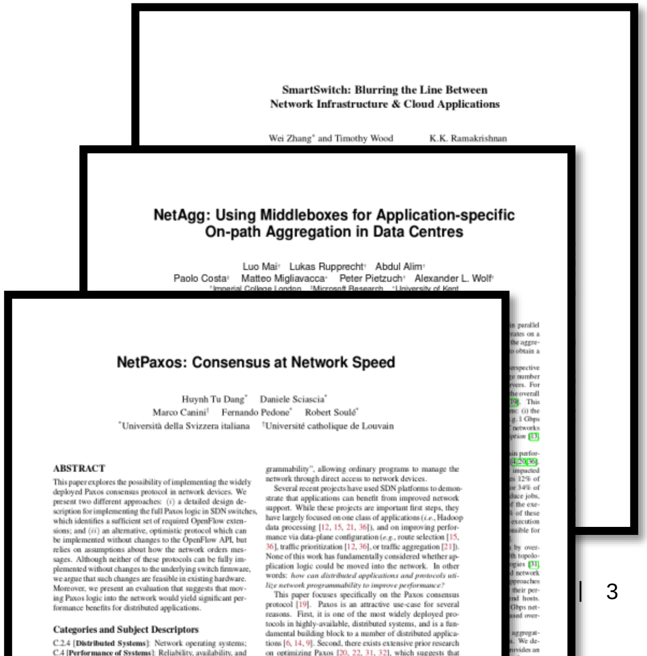
- Consensus is an essential function in datacenters
- How can consensus be made inexpensive?



Related work

- Speeding up consensus is an important problem
  - Related work in networking, systems, HPC, etc.

- Specialized hardware can remove traditional limitations



SmartSwitch: Blurring the Line Between Network Infrastructure & Cloud Applications

Wei Zhang* and Timothy Wood K.K. Ramakrishnan

NetAgg: Using Middleboxes for Application-specific On-path Aggregation in Data Centres

Luo Mai¹ Lukas Rupprecht¹ Abdul Alim¹
Paolo Costa² Matteo Migliavacca³ Peter Pietzuch¹ Alexander L. Wolf¹
¹Imperial College London ²Microsoft Research ³University of Kent

NetPaxos: Consensus at Network Speed

Huynh Tu Dang* Daniele Sciascia*
Marco Canini† Fernando Pedone* Robert Soulé*
*Università della Svizzera italiana †Université catholique de Louvain

ABSTRACT

This paper explores the possibility of implementing the widely deployed Paxos consensus protocol in network devices. We present two different approaches: (*i*) a detailed design description for implementing the full Paxos logic in SDN switches, which identifies a sufficient set of required OpenFlow extensions; and (*ii*) an alternative, optimistic protocol which can be implemented without changes to the OpenFlow API, but relies on assumptions about how the network orders messages. Although neither of these protocols can be fully implemented without changes to the underlying switch firmware, we argue that such changes are feasible in existing hardware. Moreover, we present an evaluation that suggests that moving Paxos logic into the network would yield significant performance benefits for distributed applications.

Categories and Subject Descriptors

C.2.4 [**Distributed Systems**]: Network operating systems; C.4 [**Performance of Systems**]: Reliability, availability, and

grammability", allowing ordinary programs to manage the network through direct access to network devices.

Several recent projects have used SDN platforms to demonstrate that applications can benefit from improved network support. While these projects are important first steps, they have largely focused on one class of applications (*i.e.*, Hadoop data processing [12, 15, 21, 36]), and on improving performance via data-plane configuration (*e.g.*, route selection [15, 36], traffic prioritization [12, 36], or traffic aggregation [21]). None of this work has fundamentally considered whether application logic could be moved into the network. In other words: *how can distributed applications and protocols utilize network programmability to improve performance?*

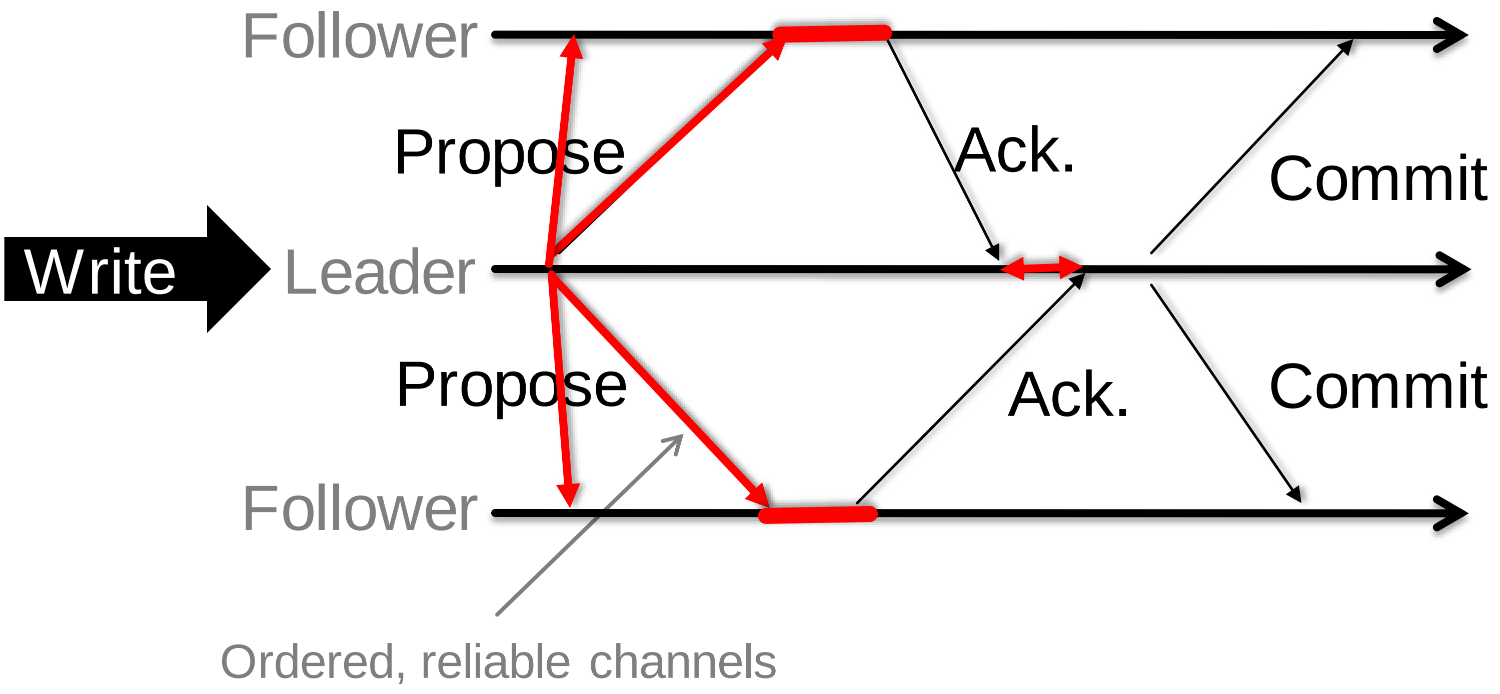
This paper focuses specifically on the Paxos consensus protocol [19]. Paxos is an attractive use-case for several reasons. First, it is one of the most widely deployed protocols in highly-available, distributed systems, and is a fundamental building block to a number of distributed applications [6, 14, 9]. Second, there exists extensive prior research on optimizing Paxos [20, 22, 31, 32], which suggests that

[1] Zhang et al. Smartswitch: Blurring the line between network infrastructure & cloud applications. In HotCloud'14
[2] Mai et al. NetAgg: Using Middleboxes for Application-Specific On-path Aggregation in Data Centres. In CoNEXT'14
[3] Dang et al. NetPaxos: Consensus at Network Speed. In SOSR'15
[4] Poke et al. DARE: High-Performance State Machine Replication on RDMA Networks. In HPDC'15.

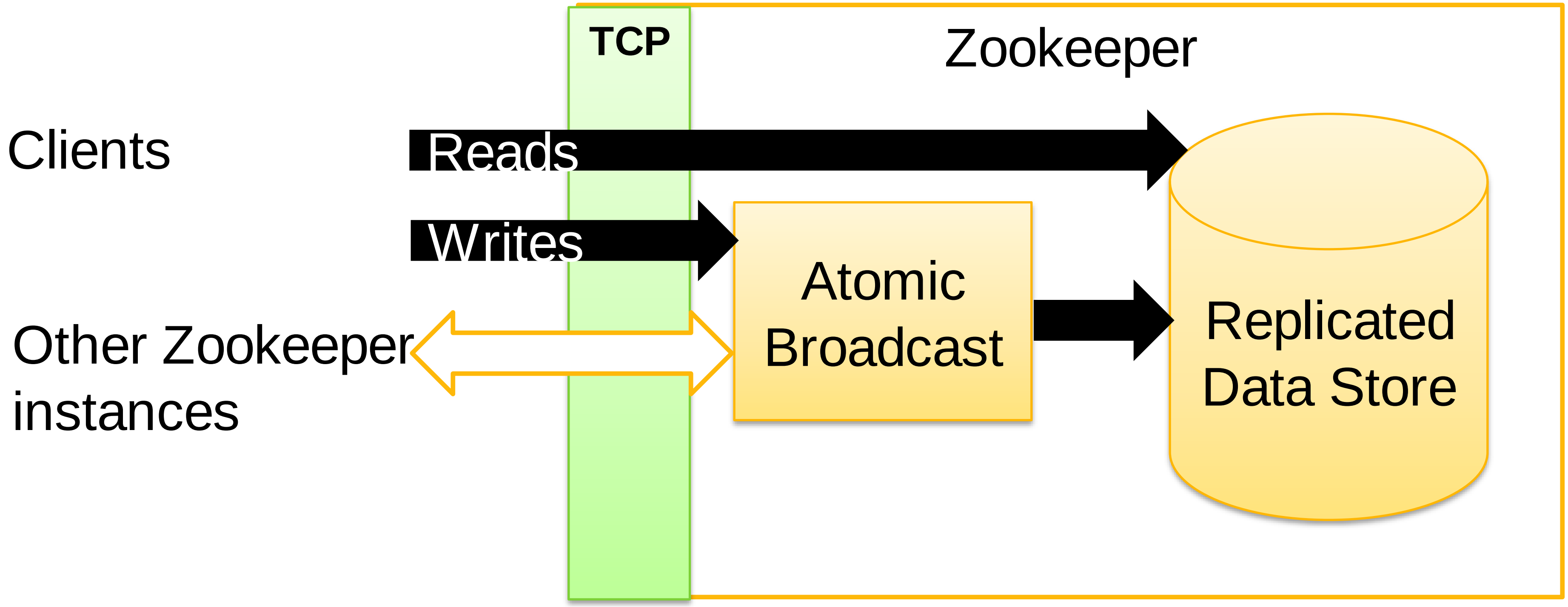
Consensus in a Box

- **Why?** Consensus is expensive, but desired
- **What?** Atomic broadcast – Zookeeper’s ZAB protocol
- **How?** Specialized processor on FPGA
  - Tight integration with 10Gbps networking + deep pipelining
- **Evaluation?** Drop-in replacement for Memcached with Zookeeper’s replication

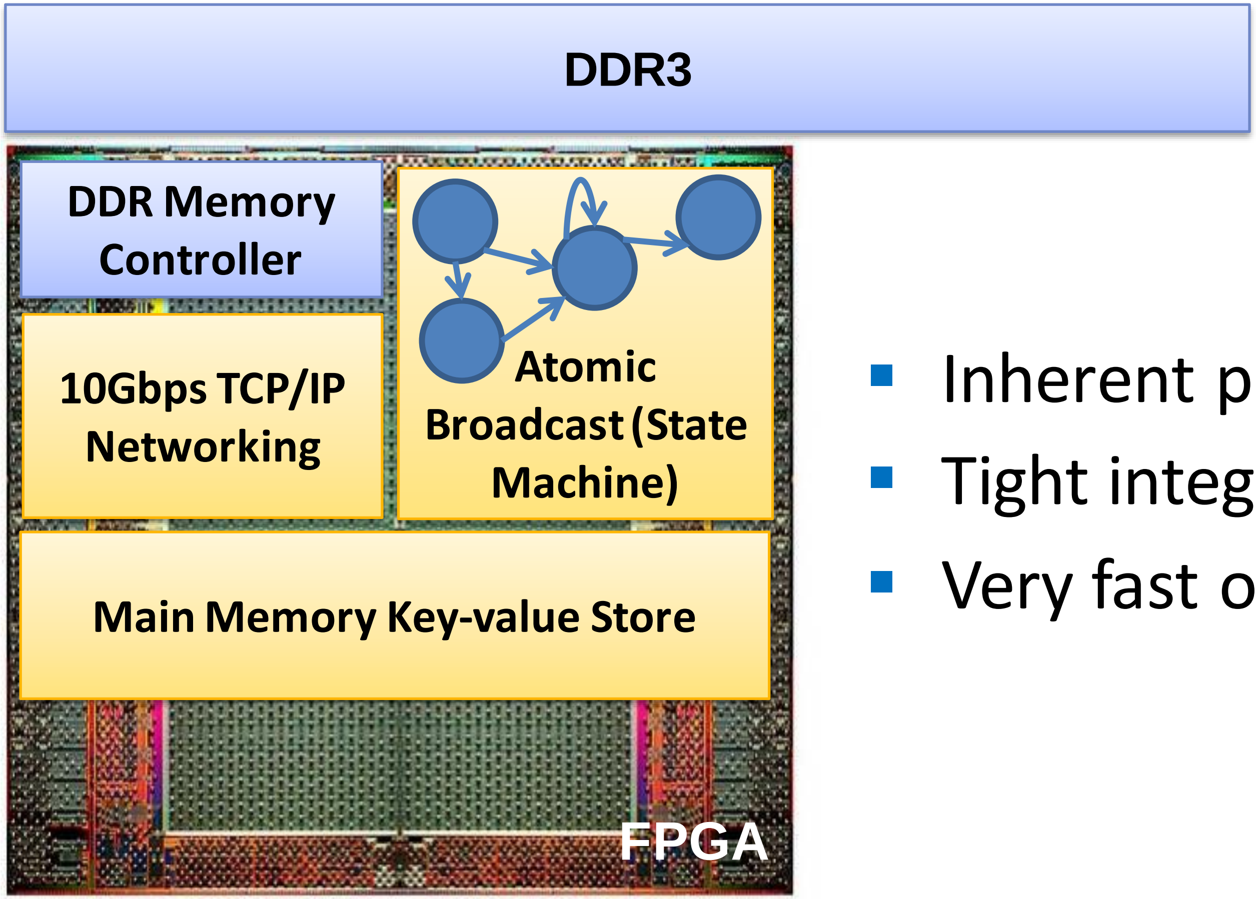
Zookeeper's Atomic Broadcast



Zookeeper from 10000ft



Specialized processor architecture



DDR3

DDR Memory Controller

10Gbps TCP/IP Networking

Atomic Broadcast (State Machine)

Main Memory Key-value Store

FPGA

- Inherent parallelism
- Tight integration
- Very fast on-chip memory

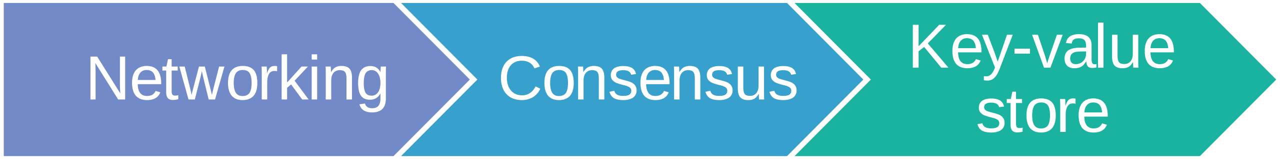
What makes it go fast…

Latency

- **Networking optimizations**
  - Low-latency on-chip buffers for RX path
  - Datacenter and application-specific knowledge
- **Predictable behavior**
  - Fast local caches for common case behavior

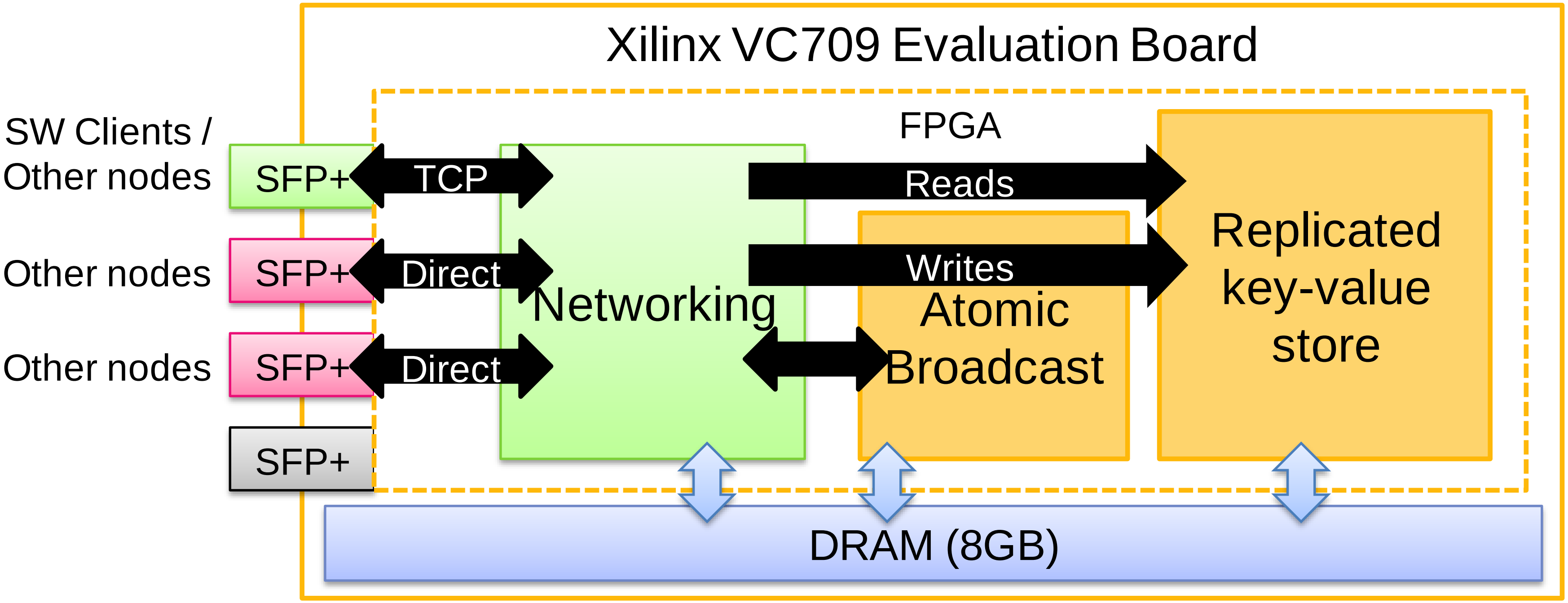
Throughput

- **Pipelined execution**

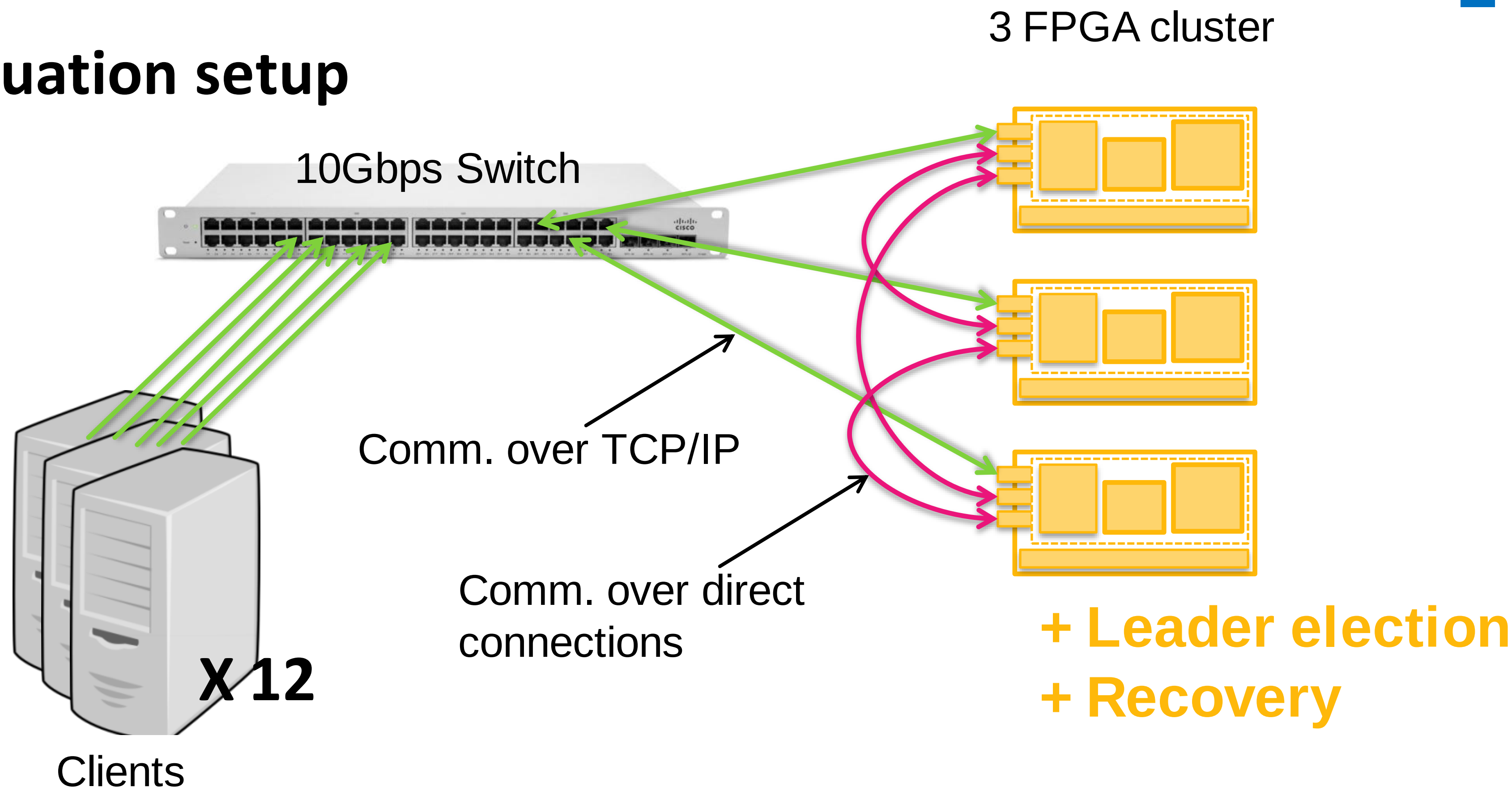


Deployment and Evaluation

Hardware platform



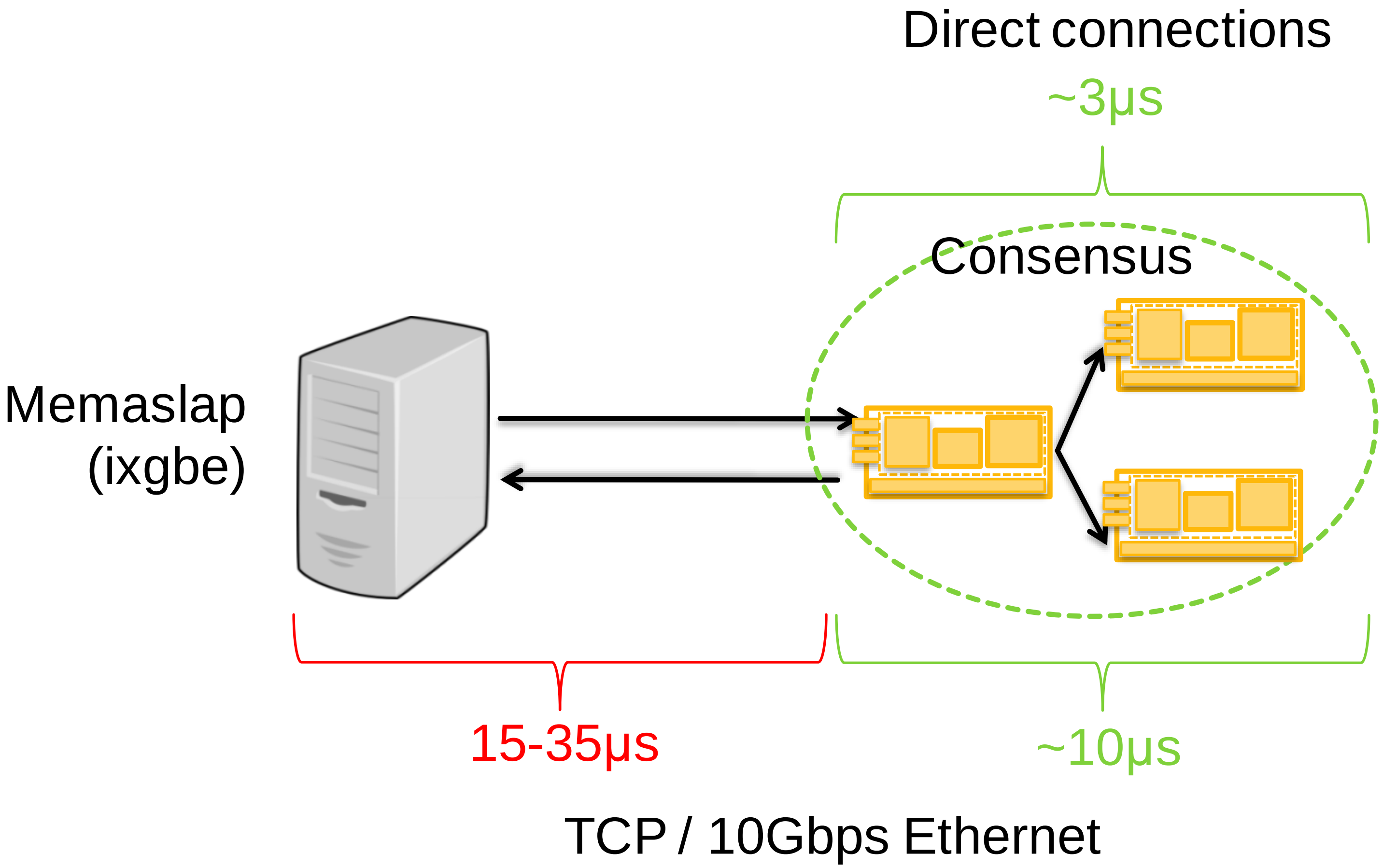
Evaluation setup



+ Leader election
+ Recovery

- Drop-in replacement for Memcached with Zookeeper's replication
- Standard tools for benchmarking (libmemcached)
  - Simulating 100s of clients

Latency of KVS writes (consensus)



Direct connections

~3µs

Consensus

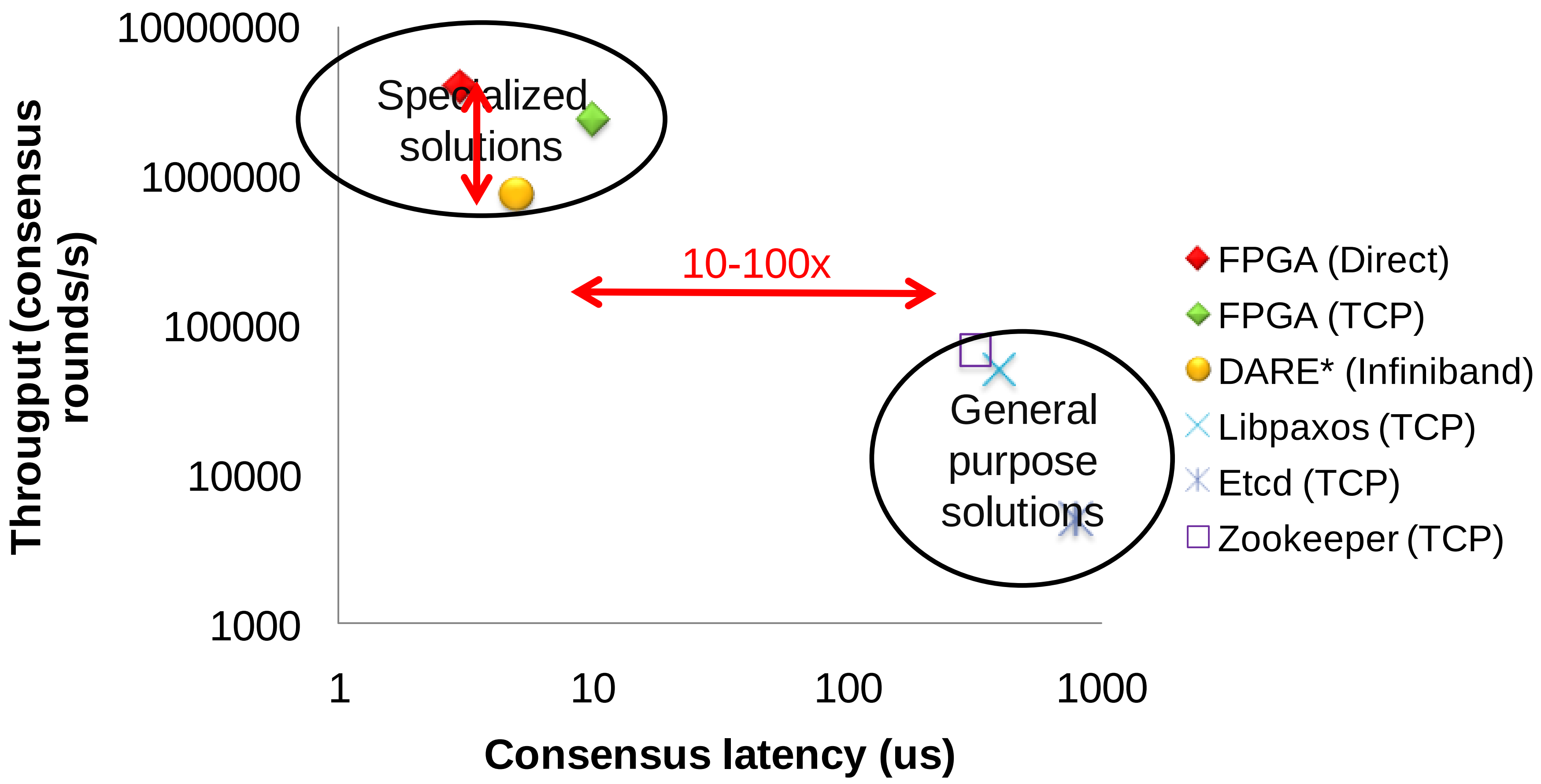
Memaslap (ixgbe)

15-35µs

~10µs

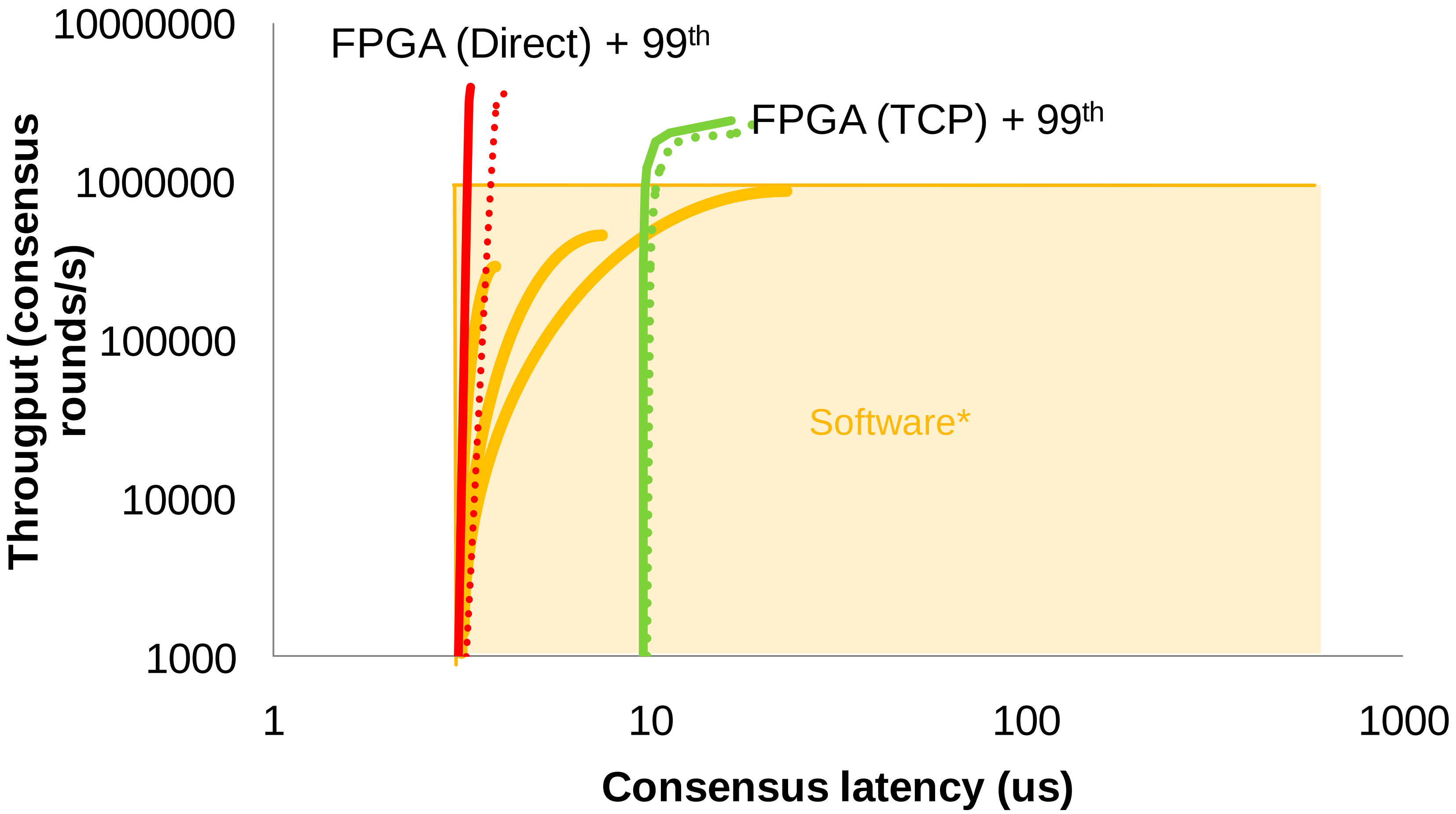
TCP / 10Gbps Ethernet

The benefit of specialization…

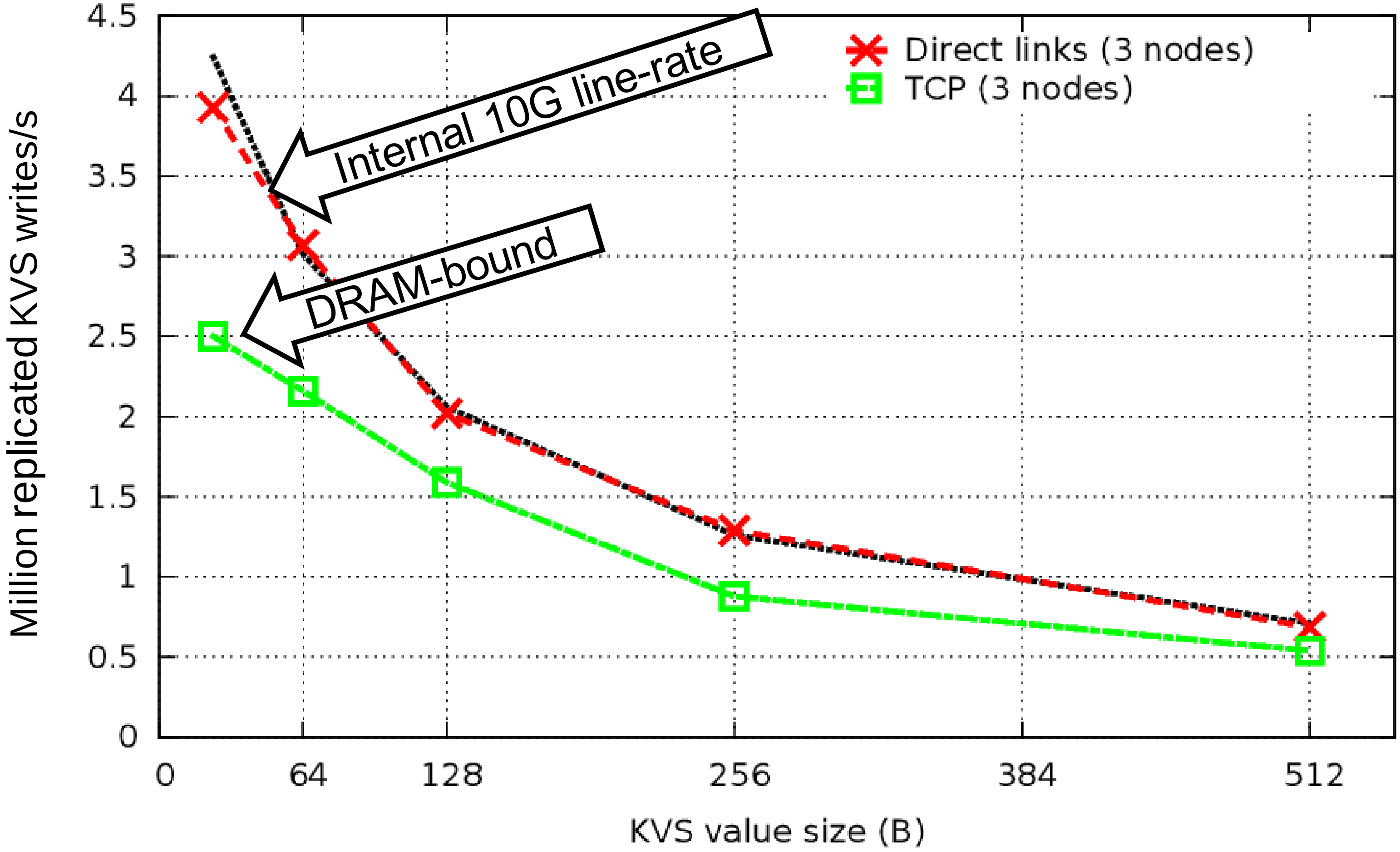


[1] Dragojevic et al. FaRM: Fast Remote Memory. In NSDI'14.
[2] Poke et al. DARE: High-Performance State Machine Replication on RDMA Networks. In HPDC'15.
*=We extrapolated from the 5 node setup for a 3 node setup and removed estimated client overhead.

Predictable hardware performance

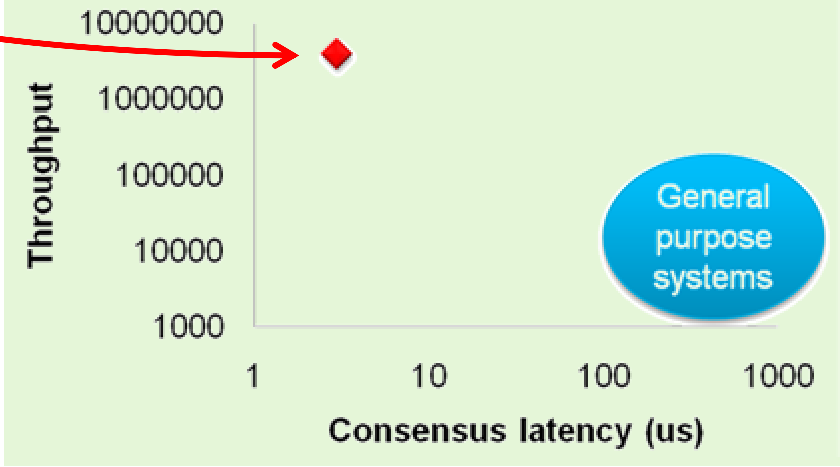


Throughput overview



Conclusion

- Consensus is expensive but often necessary
  - Solution: specialization and tight integration with networking
- We built high-throughput low-latency consensus in hardware



- Specialized hardware opens up new opportunities for smarter networks

{zsolt.istvan},{david.sidler}@inf.ethz.ch