# 有意思的论文：KV-Direct内存键值存储（2017）

Accela Zhao Accela推箱子 2017-12-02

**---- KV-Direct论文----**

[KV-Direct: High-Performance In-Memory Key-Value Store with Programmable NIC](https://lrita.github.io/images/blog/kv-direct.pdf)

[KV-Direct Introduced by The Morning Paper](https://blog.acolyer.org/2017/11/23/kv-direct-high-performance-in-memory-key-value-store-with-programmable-nic/)

**---- 一些背景----**

键值存储Key-Value Store经常被用作构造分布式存储或数据库的积木。流行的KV-Store如RocksDB基于LevelDB改进，强化了LSM-tree的compact（经典的Universal Compaction），增加诸多易用功能。例如，RocksDB被用于Ceph（分布式块、对象、文件存储）作单结点（OSD）内的存储引擎（BlueStore）。CockroachDB（仿Google Spanner式的分布式关系数据库）以RocksDB作基本存储，并将SQL表映射成KV操作。

[Universal Compaction](https://github.com/facebook/rocksdb/wiki/rocksdb-basics)

[Ceph BlueStore](http://www.sysnote.org/2016/08/19/ceph-bluestore/)

[CockroachDB SQL Mapping to KV](https://www.cockroachlabs.com/blog/sql-in-cockroachdb-mapping-table-data-to-key-value-storage/)

另一方面，内存数据库如今发展迅速。商用的如Pivotal GemFire、SAP HANA支持复杂强大的功能。开源如内存Cache的Memcached、Redis也被互联网界广泛使用。总体上，计算机硬件内存容量迅速增大。可以用磁盘、SSD存储大规模的普通数据，而对要求高吞吐量高一致性的Transaction处理（如账单结算、金融）则使用内存数据库。

**---- KV-Direct ----**

由此想来，内存KV-Store就是一个值得探索的方向。The Morning Paper已讲KV-Direct，基本覆盖所有重点。

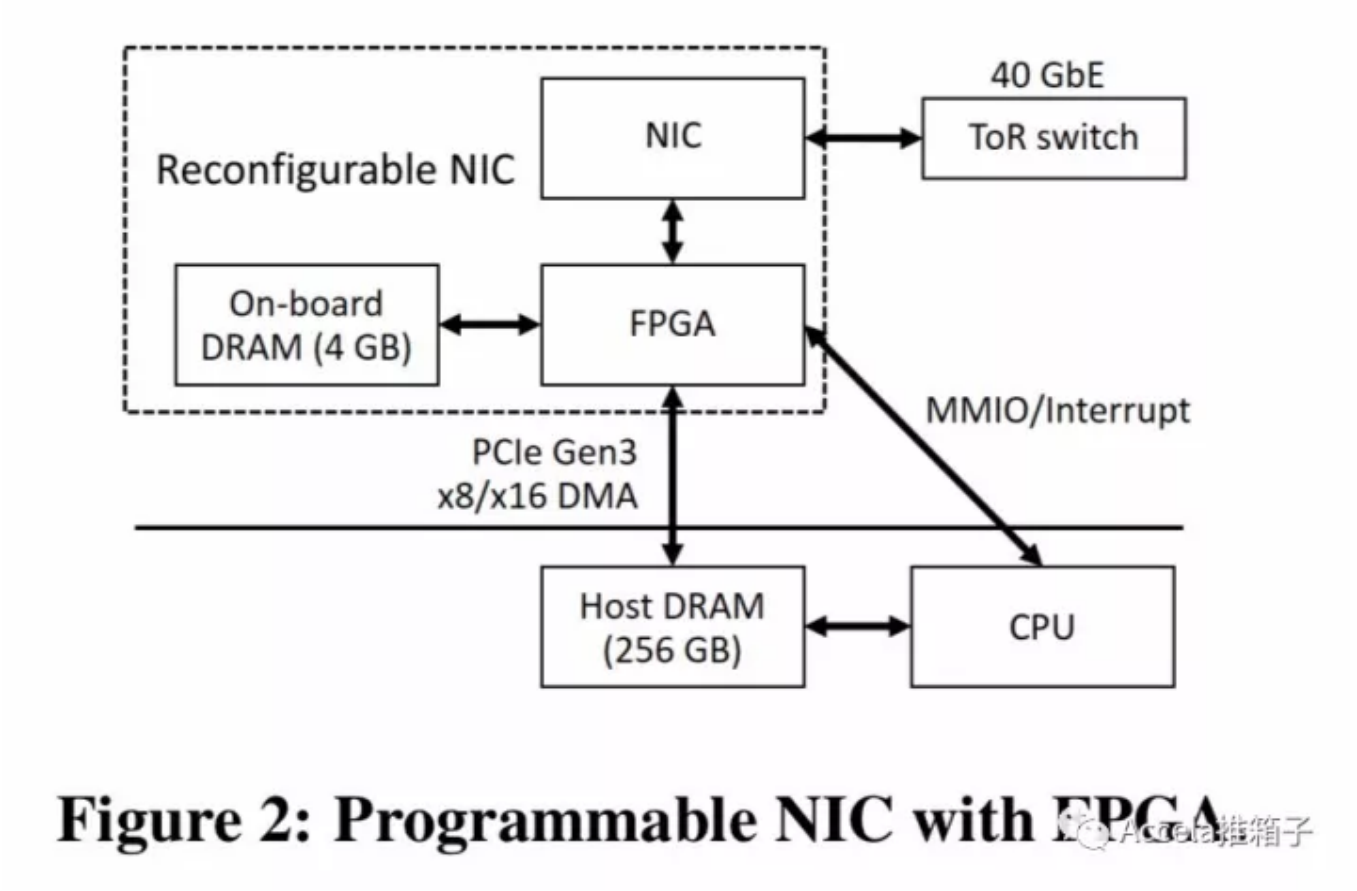


**Figure 2: Programmable NIC with FPGA**

KV-Direct真正的亮点是把KV-Store的增删改查操作烧入FPGA执行，无需远端CPU参与（One-sidedRDMA）；单NIC达到180Mops远超其它内存存储如RAMCloud、FaRM。更强大的是，FPGA位于Programmable NIC（可编程网卡；P.S.新代加FPGA的硬件常被称作Reconfigurable XX）内，配有卡内DRAM，可存部分KV数据；这意味着此数据操作不需经过PCIe总线，在NIC卡内完成，访问延迟极低；KV-Direct达到tail latency小于10us。再强大的是，论文测试一块KV-Direct NIC的吞吐量可抵十多个CPU核，但一台服务器是可以插许多NIC的。最后，FPGA比CPU省电得多（数据中心TCO大头是耗电和服务器散热，热量也因耗电起），KV-Direct能效Kops/W是其它内存存储3x（算能效时是加了FPGA自身耗电的）。

| KVS | Comment | Bottleneck | Tput (Mops) | | Power Efficiency (Kops/W) | | Avg Delay (μs) | |
|---|---|---|---|---|---|---|---|---|
| | | | GET | PUT | GET | PUT | GET | PUT |
| Memcached [25] | Traditional | Core synchronization | 1.5 | 1.5 | ~5 | ~5 | ~50 | ~50 |
| MemC3 [23] | Traditional | OS network stack | 4.3 | 4.3 | ~14 | ~14 | ~50 | ~50 |
| RAMCloud [59] | Kernel bypass | Dispatch thread | 6 | 1 | ~20 | ~3.3 | 5 | 14 |
| MICA [51] | Kernel bypass, 24 cores, 12 NIC ports | CPU KV processing | 137 | 135 | 342 | 337 | 81 | 81 |
| FaRM [18] | One-sided RDMA for GET | RDMA NIC | 6 | 3 | ~30 (261) | ~15 | 4.5 | ~10 |
| DrTM-KV [72] | One-sided RDMA and HTM | RDMA NIC | 115.2 | 14.3 | ~500 (3972) | ~60 | 3.4 | 6.3 |
| HERD'16 [37] | Two-sided RDMA, 12 cores | PCIe | 98.3 | ~60 | ~490 | ~300 | 5 | 5 |
| Xilinx'13 [5] | FPGA (with host) | Network | 13 | 13 | 106 | 106 | 3.5 | 4.5 |
| Mega-KV [75] | GPU (4 GiB on-board RAM) | GPU KV processing | 166 | 80 | ~330 | ~160 | 280 | 280 |
| **KV-Direct (1 NIC)** | Programmable NIC, two Gen3 x8 | PCIe & DRAM | 180 | 114 | 1487 (5454) | 942 (3454) | 4.3 | 5.4 |
| **KV-Direct (10 NICs)** | Programmable NIC, one Gen3 x8 each | PCIe & DRAM | 1220 | 610 | 3417 (4518) | 1708 (2259) | 4.3 | 5.4 |

**Table 3: Comparison of KV-Direct with other KVS systems under long-tail (skewed Zipf) workload of 10B tiny KVs. For metrics not reported in the papers, we emulate the systems using similar hardware and report our approximate measurements. For CPU-bypass systems, numbers in parentheses report power difference under peak load and idle.**

KV-Direct内部有许多有意思的优化，如减少每个KV操作所需的DMA数量，定制的哈希表和slab内存分配器，乱序执行引擎中使用的reservation station和cache（有意思的一点是，磁盘存储的commit语义要求用户数据一定落盘，而内存存储的commit语义允许用户数据位于cache或内存都行，可以更快返回调用），等等。此外KV-Direct还在FPGA中实现了对KV的vector操作，对机器学习、图处理等有用。

另外，汇总RDMA优化的有FaRM论文，而RAMCloud算是分布式内存存储的鼻祖（也就这几年），它还实现了SILK作Secondary索引（数据库常需对非主键查询）。

[FaRM: No compromises: distributed transactions withconsistency, availability, and performance](http://sigops.org/sosp/sosp15/current/2015-Monterey/printable/227-dragojevic.pdf)

[Design Guidelines for High Performance RDMA Systems](https://www.usenix.org/system/files/conference/atc16/atc16_paper-kalia.pdf)

[RAMCloud: Log-structured Memory for DRAM-based Storage](https://www.usenix.org/system/files/conference/fast14/fast14-paper_rumble.pdf)

[SLIK: Scalable Low-Latency Indexes for a Key-Value Store](https://www.usenix.org/node/196191)

KV-Direct的性能数据呈数量级超过同代系统，但论文测试也有水分。与RAMCloud、FaRM相比，KV-Direct没有实现on-disk logging（一般内存存储是要管数据持久化的）、transaction、分布式管理等功能，功能轻量自然速度快。测试中较好的性能数据使用的是

非常小的键值对（~10B），实际用户一般会用大得多的。而论文设计可以以饱和PCIe带宽和网络带宽为目标，实际产线负载往往不会如此理想。这些水分是大部分论文通有的，但KV-Direct的亮点非常出色。

**---- FPGA ----**

在深度学习带火了GPU的同时，FPGA的使用也日渐普及。一般将其用于独立的重复性计算，如压缩、加密、网络处理（网络虚拟化白牌交换机等光靠软件难以跟上网速）、定制化的计算（如专用算Page Rank）等等。数据中心级或云级的大量FPGA产线部署已经实现，如Catapult池化架构，可以为Bing提供搜索和网页排序服务

[A Cloud-Scale Acceleration Architecture](ftp://ftp.cs.utexas.edu/pub/dburger/papers/MICRO16.pdf)

FPGA、ASIC、GPU、SmartNIC、RDMA、NVM、SR-IOV等等随着价格降低，越来越普及，逐渐渗透到各种领域。原本它们常用于高性能计算，被称作Accelerator。ASIC用于Google深度学习专用芯片TPU；GPU广泛用于深度学习；SmartNIC基本同前文的可编程网卡；RDMA原本昂贵高大上，现在普及到SSD集群标配，内存计算必用（否则带宽跟不上内存速度）；NVM包括了将Flash接到PCIe、接到NVMe协议、接到内存线、（接SCSI/SAS则常指SSD），它用于存储系统的cache和journal非常有效；SR-IOV多指网卡硬件虚拟PF、VF给虚拟机用，解决了软件做IO虚拟化开销大的老大难问题（虚拟化起初由软件实现兴起，最后因性能多离不开硬件支持，最终靠着IntelVT-d、VT-i、VT-c、VT-x）。这些Accelerator，如今随着高速云计算和虚拟化成为日常。

KV-Direct将FPGA更推进了一步，从做重复计算，到将存储的基本功能增删改查编写进FPGA。可以看到这些Accelerator已经越来越深的渗透到经典的存储系统，不仅仅是边缘性地帮助压缩和加密，而是改变基础功能。在未来也许可以看到存储架构的大幅转变，而数据中心级或云级的FPGA池化则可以想见。以FPGA为基础的存储系统会是什么样子？

过去，存储系统以scale-out为名，脱离定制硬件，主流转到commodity hardware宣传上来。如今，随着越来越多带有定制性的Accelerator加入，也许历史又将转回定制硬件的色彩，或者commodity hardware的含义逐渐发生改变，在螺旋中演进。

喜欢此内容的人还喜欢

全国悲痛！巨星陨落，他是14亿国人都该致谢的救星！

宪法小卫士

抱团股到底咋了！

研报社