Computational Storage to Increase the Analysis Capability of Tier-2 HEP Data Sites

Chen Zou
Department of Computer Science
University of Chicago
Chicago, USA
chenzou@uchicago.edu

Andrew A. Chien
CS, University of Chicago
MCS, Argonne National Laboratory
Chicago, USA
achien@cs.uchicago.edu

Robert Gardner
Enrico Fermi Institute
University of Chicago
Chicago, USA
rwg@uchicago.edu

Ilija Vukotic
Enrico Fermi Institute
University of Chicago
Chicago, USA
ivukotic@uchicago.edu

***Abstract*—Large Hadron Collider(LHC) produces collision data at 100 PB/year which needs to be stored and analyzed for high energy physics(HEP) theories. We reconsider the design choices of HEP data centers and evaluate different upgrade options to improve their analysis capacity.**

Results show that computational storage is the most cost-effective and power-efficient upgrade option. By employing computational storage disks in the storage cluster, Higgs Boson analysis, HEP's stellar analysis workload, enjoys the speedup of 9.3x, exceeding the speedup delivered by more expansive upgrades options such as improving backbone network from 100 Gbps to 1000 Gbps and switching from hard disks to SSDs in compute cluster.

***Index Terms*—High energy physics, OLAP, Computational storage, RISC-V**

I. Introduction

Detecting collision events from 8 different experiments (e.g. ATLAS, CMS, LHCb, ALICE) around the ring, the Large Hadron Collider(LHC) produces a huge amount of data. Despite the filtering from purposely-built real-time triggering mechanisms, the collected events still mount up to 100PB per year for physicists around the world to analyze and test different theories of high-energy physics(HEP). Limited by cost considerations, the conventional wisdom on developing the HEP data centers is to buy disks to host the continuously growing data. As one would imagine, the growing analysis capacity requirement is overlooked.

Now is a good time to rethink these design choices. LHC is in the long shutdown period to upgrade for the High Luminosity LHC project that will increase the luminosity(and hence the collision rate and HEP data collection) by a factor of 10. Much greater analysis capacity will be needed once the LHC resumes its operation.

In this work, we aim to break the chain of conventional wisdom. By developing a parametric and detailed model of a tier-2 data center, the workhorse building block in the HEP grid, we carefully evaluate several different upgrade options to match the increasing analysis capacity demand.

Specific contributions include:

- Detailed modeling on HEP data center resources enables parametric performance analysis for HEP data centers.
- Identification and exemplification of employing computational storage that brings the highest analytical capacity improvement.
- A consistent analytical model from conventional bottleneck analysis that produces clear evidence for why computational storage is a better solution.

We will first provide backgrounds on HEP data centers and analysis workload in Section II. Our modeling for Tier-2 HEP data sites and experiment methodology is introduced in Section III. The experiments evaluating different upgrading options are detailed in Section IV. The analytical performance model along with cost considerations is discussed in Section V. We discuss related work in Section VI.

II. Background

A. High energy physics data centers

To store and process the enormous amount of data ($\sim$ 100 PB each year) from the different experiment apparatus from the large hadron collider (LHC) and enable collaboration of scientists around the globe, CERN builds a grid [1] connecting more than one hundred computing centers in more than 40 countries. These data centers are organized in tiers:

- Tier-0: CERN LHC producing data of interesting collision events and storing them (EB-level) in tape archives.
- Tier-1: Main data centers at major participating countries, storing backup copies of collision data (10 - 100 PB) and handle collision event reconstruction and recalibration.
- Tier-2: Community data centers inside research institutes and universities, sharing the grid responsibility of replicating collision events and handling analysis requests. Storing PB-level data.

Collision event data are distributed and replicated across the grid through distribution policies for data integrity purposes. To reduce the data movement and improve system efficiency, computations are usually shipped to the data center site holding the corresponding data copy.

B. Higgs boson analysis

In this study, we use an HEP stellar workload: Higgs boson analysis [2] on public CMS collision data [3] to understand the performance of the level 2 data centers. Higgs boson analysis map two steps of computation to each collision event:

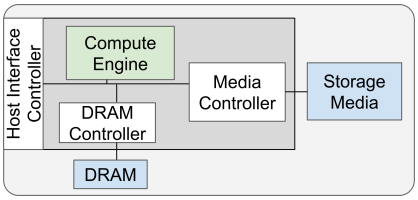

Fig. 1: Computational storage device

- Apply filters to muons, global muons, electrons tracks.
- Calculate statistics on filtered muons, global muons, electrons tracks in the event for potential Higgs bosons.

The most important computation characteristics are that the computation mapped to each event is independent from the other event. This signifies the huge parallelism available. The public CMS dataset contains 250 million collision events totaling 78TB (47TB real events are used in the study, there are also 31TB monte-carlo-simulated events) stored in roughly 25000 files, which corresponds to at least 25000-way of parallelism, let along we can further subdivide the event files. This level of parallelism exposes the opportunity of low-latency analysis which would greatly reduce the turn-around time and improve the analysis capacity in tier-2 data centers.

C. Computational storage

The storage industry has seen rapid increases in storage device capacity and bandwidth. It shifts bottlenecks in the modern data center architecture to the CPU and interconnects and draws attention to the old question on how to best divide compute and storage dating back to the 1990s.

Computational storage addresses this changing balance. By adding computation capabilities inside the storage devices as in Figure 1, computing can be shifted closer to storage, reducing data movement. Further, computational storage also allows computation to scale with storage bandwidth.

III. Performance modeling

A. Model for a tier-2 data center

We build the performance model for a tier-2 data center after the UChicago tier-2 data center(named as UChicagoT2), which is shown in Figure 2.

The data center has one compute cluster named 'UCT2' and one storage cluster named 'DCache' which as the name suggested, manages the collision event data with DCache [4].

The model for UCT2 compute cluster is composed of 256 compute nodes with 32 cores and a 1TB hard disk in each node. Each UCT2 node is connected to the top-level router of the cluster with a 10-Gbps Ethernet link.

The model for DCache storage cluster is composed of 30 compute nodes, each node equipping with 40 1TB hard disks. Each DCache node is connected to the cluster top-level router with a full-duplex 10-Gbps Ethernet link.

UCT2 and DCache cluster are connected to each other as well as the WAN network with a full-duplex 100-Gbps Ethernet link. Hard disks are modeled to have 200 MB/s read bandwidth and 100 MB/s write bandwidth. As a result, with the current baseline configuration, the UCT2 compute cluster and DCache storage cluster respectively possesses 410 Gbps and 1920 Gbps disk read bandwidth in total. These aggregate numbers are meaningful later in the analytic system performance model.

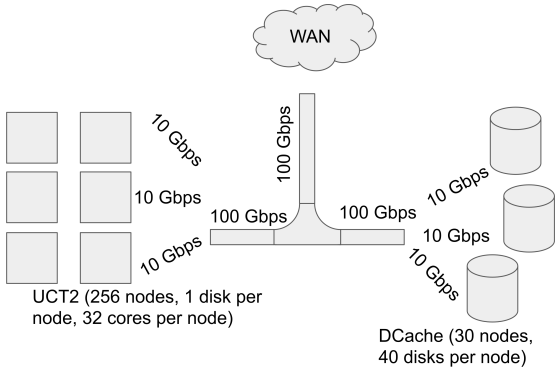

Fig. 2: UChicagoT2 data center modeling

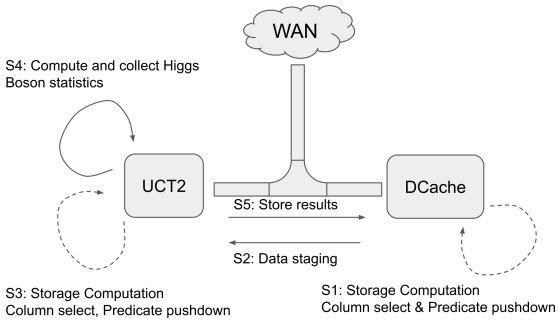

Fig. 3: Higgs boson analysis mapping to the modeled UChicagoT2 data center

B. Workload

Higgs boson analysis with 47TB CMS data is used as the workload. We analyzed the computational structure and high parallelism of the Higgs boson analysis in Section II-B. Here we discuss how the Higgs boson analysis is implemented in the modeled UChicagoT2 cluster in Figure 3.

As discussed in Section II-A, the computation are shipped to the data center holding the corresponding data to avoid excessive network transfer. Thus, it is assumed that 47 TB CMS data is hosted in the DCache cluster in our modeled tier-2 data center.

The Higgs boson analysis workload is carried out in a multi-stage fashion in our modeled tier-2 data center, The processing of each file (3GB, 10000 events) in the public CMS dataset is treated as independent jobs. Each job is composed of the following stages:

- Stage 1: In-storage computation at DCache cluster
- Stage 2: Data staging: DCache $\rightarrow$ UCT2
- Stage 3: In-storage computation at UCT2 cluster
- Stage 4: Higgs boson Statistics compute at UCT2 cluster
- Stage 5: Store back results: UCT2 $\rightarrow$ DCache

Stage 1 and Stage 3 are optional and only applicable when computational storage is equipped at the DCache or UCT2 cluster. We measure the application-dependent throughput of Stage 4 on one of the UCT2 machines. We warm up the OS page cache by reading the file five times to get rid of the disk performance influence and measure the real computation throughput.

C. Approach

The configurations and performance numbers discussed above model the current UChicago T2 data center, which is the baseline. We further consider different upgrades:

- Improving backbone networks speed from 100 Gbps to 1000 Gbps.
- Replacing hard disks with solid-state drives in either UCT2 and DCache cluster or both.
- Adding computational storage.

We evaluate analysis performance under baseline and with different upgrades to determine the most effective way to improve the analysis capacity of a tier-2 data center.

Considering the analysis workload is parallel as discussed in Section II-B, we employ a throughput-oriented discrete event performance model. A C++ simulator implements the above performance model with task-level granularity, where each stage for a job is instantiated as a task. It simulates task progress under the customizable resource properties specified in the performance model and produce the application performance measured in overall latency.

D. Metric

We measure performance with the metric 'latency' of Higgs boson analysis that analyzes 47TB collision events. We track the system performance and bottleneck through 'utilization' of different kinds of resources. The utilization is generally defined as the ratio between aggregate(across nodes) throughput and aggregate bandwidth capacity for the network links and storage devices. For the computing devices, the utilization is directly the average CPU utilization across cores and nodes.

For the cost, we measure silicon area as well as power consumption. Intuitively, the best upgrades are most cost-efficient and energy-efficient.

IV. Experiments

We first evaluate performance of the current UChicago tier-2 data center. And then we assess the upgrade options of increasing the available disk bandwidth by switching from hard disks to SSDs, upgrading network with higher bandwidth capacity for the backbone network connecting the UCT2 compute cluster and the DCache storage cluster, and employing computational storage at either UCT2 or DCache cluster. From there, we analyze the Higgs boson analysis performance of each upgrade and determine which one works the best to improve the analysis capacity.

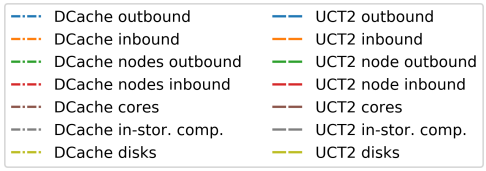

Fig. 4: Shared legend for different resources. DCache and UCT2 differ in linestyle for the same type of resource.

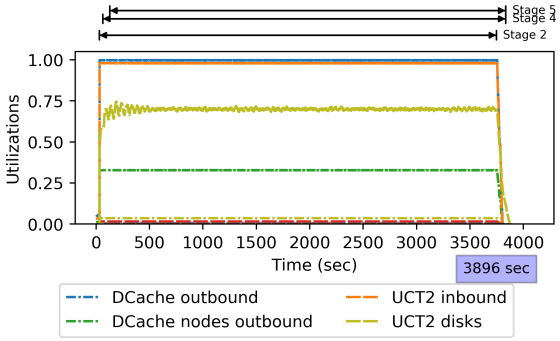

Fig. 5: Performance and resource utilization in baseline

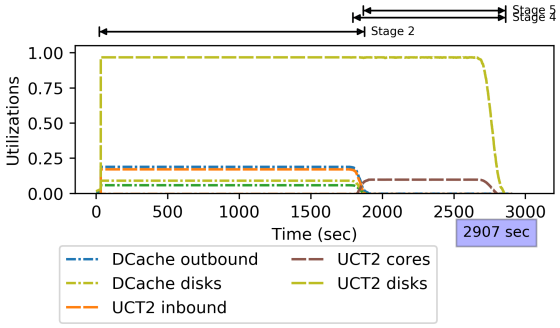

Fig. 6: Upgrade backbone network: 100 Gbps to 1000 Gbps

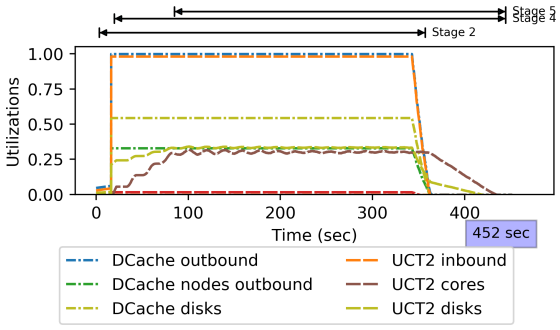

Fig. 7: 1000 Gbps backbone network + UCT2 SSDs

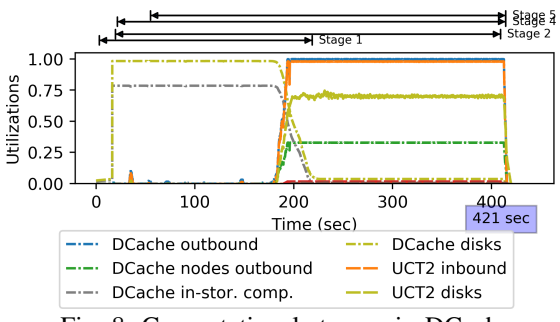

Fig. 8: Computational storage in DCache

A. Baseline

Almost all the figures plot the utilization of different resources described in Section III-A in UChicagoT2 and follow the same organization. The figures all use the same legend shown in Figure 4. We will still include a legend for the resources with notable utilization under each plot for easier identification of the highly utilized resources. On top of each plot, there are marks shown the period when tasks of each stage are being processed.

The configuration of different components in the baseline model is defined in Section III-A. Figure 5 details its Higgs boson analysis performance. Each line plots the utilization of a specific resource over time. The number in the shaded box just below the x-axis (i.e. 3896 sec) is the overall latency of the analysis. As shown in the legend and the plot, DCache outbound 100 Gbps link and UCT2 inbound 100 Gbps link were almost always at nearly 100% utilization, performing Stage-2 tasks of transferring data from DCache to UCT2. These two resources are the performance bottleneck in the baseline.

B. Improving backbone network from 100 Gbps to 1000 Gbps

We consider upgrading the backbone network, i.e., upgrading DCache cluster inbound/outbound links and UCT2 cluster inbound/outbound links from 100 Gbps to 1000 Gbps, to address the aforementioned performance bottlenecks. The performance and utilization of Higgs boson analysis on the upgraded UChicagoT2 data center are shown in Figure 6. The upgrade increases analysis performance by 1.3x. Backbone network (DCache outbound and UCT2 inbound) utilization drop significantly, while UCT2 disks become the system bottleneck. This led to the stall of Stage 4 tasks that read events from UCT2 disks while Stage 2 tasks are intensively writing to UCT2 disks during the DCache to UCT2 event file transfer (see stage duration at the top of the figure).

C. Further replacing HDDs with SSDs in UCT2

On top of the network upgrade, we further consider replaces HDDs with SSDs for UCT2 cluster as its disks were the performance bottleneck. SSDs are modeled as 4 GB/s for read and 2 GB/s for write, as opposed to 200 MB/s and 100 MB/s for HDDs as discussed in Section III-A. This led to a further 6.4x latency reduction, and 8.6x from baseline, as shown in Figure 7. The system performance bottleneck circles back to backbone networks as both DCache outbound and UCT2 inbound utilization is at nearly 100% for most of the analysis.

D. Employing computational storage

We consider a non-conventional upgrade to the UChicagoT2 data center directly from baseline: adding storage computation capability to hard disks in DCache cluster. Column selecting (slimming) are offloaded to computational storage such that only selected fields in each data tuple needed for later statistics calculation are transferred to UCT2 cluster.

Each computational storage disk embeds an in-order single-scalar core for in-storage-device computation on top of the data streams from storage media. We model this core after an in-order RISC-V core Ibex [5]. RISC-V is chosen for its compilation tool-chain availability to streamline the evaluation, but we believe the results are independent of specific ISA and transferrable. We use Spike [6] to simulate column select tasks and emit execution traces for performance analysis.

Figure 8 details the performance after only upgrading DCache cluster to use computational storage disks from baseline. Because data is selectively transferred out from DCache storage after the offloaded column select operations, the bandwidth requirement on the later resources (backbone networks, UCT2 disks, UCT2 compute) is largely reduced. Overall, this upgrade generates 9.3x speedup, 7.4% better than upgrading both the backbone network and storage in UCT2 from HDDs to SSDs. The system bottleneck is DCache disks serving data to DCache in-storage computation. If further improving DCache disks' bandwidth to match the in-storage computation, Higgs boson analysis can enjoy another 10% speedup.

V. Discussion

A. Analytical performance model

In fact, due to the highly parallel and efficient pipeline nature of the Higgs boson analysis, the analysis performance measured in latency can be calculated approximately through the system performance.

Without computational storage(as it would change the amount of data for later stages), we could view the analysis as the process of shovelling same amount of data into pipes with different bandwidth: DCache disk read, DCache $\rightarrow$ network, UCT2 disk write, UCT2 disk read, UCT2 compute. The results write back stage is omitted as the amount of data (statistics) to store back is more than 10000 times less. Under this framing, the application performance is clearly determined by the pipe with lowest bandwidth:

$$
\begin{aligned}
bn &\doteq BackboneNetworkBandwidth \\
dd &\doteq DCacheAggregateDiskReadBandwidth \\
ud &\doteq UCT2AggregateDiskReadWriteBandwidth \\
uc &\doteq UCT2AggregateComputeThroughput \\
ApplicationLatency &= \frac{EventDataSize}{min(bn, dd, ud, uc)}
\end{aligned}
$$

Table I compares the latency based on the analytical model and the latency from the simulation. The analytical latency only significantly deviates from simulated latency in the upgrades of both 1000 Gbps backbone and UCT2 SSD. The 8.6x speedup in this case makes the latency (around 70 sec) of a single stage-4 task (Higgs boson statistics computing on a single 3.8 GB CMS data file) visible.

Under this analytical model, the upgrade consideration essentially follows the 'Liebig's barrel'. This is the reason we see network upgrade is of top priority and then upgrading compute cluster UCT2 storage with SSDs.

	Data Size	Bottleneck	Bandwidth	Analytical Latency	Simulated Latency
Baseline	47.2 TB	Backbone network	100 Gbps	3776 s	3896 s
1000 Gbps Backbone		UCT2 disks R+W	410 / 3 Gbps	2762 s	2907 s
1000 Gbps Backbone + UCT2 SSD		Backbone network	1000 Gbps	378 s	452 s

TABLE I: Analytical model VS simulated results

	Area(mm^2)	Power(mw)
Computational storage disks	1.58	9.64
Extra CPU in DCache	63.41	23216.00
Ratio	40.2	2408.5

TABLE II: Silicon area and power

B. The case for computational storage

Computational storage at DCache is the out-of-box solution that breaks 'Liebig's barrel' trade-off as a local optimum. By preprocessing and emits only the needed data, computational storage reduces the data size for all resources used in the later stage. It essentially improves the effective bandwidths of these later resources by the reciprocal of the selection ratio (averaging at 6.53% in Higgs boson analysis on CMS data).

As a result, employing computational storage alone would virtually upgrades both the backbone network and UCT2 SSDs, and much more, greatly improving analysis capacity in tier-2 data sites.

C. Cost considerations

Many may wonder, does the benefits of computational storage apply by adding computing resources(CPUs) in the DCache cluster and perform the column selecting tasks on the added computing resources. The short answer is 'Yes', but that would not the most cost-efficient or energy-efficient solution.

We compare the costs in silicon area and power consumptions of employing CPUs or computational storage disks to perform the column selecting tasks before sending out from the DCache storage cluster. Through benchmarking, the computing performance on column selection of 40 in-scalar RISC-V cores amounts to 9.1x of a single Skylake-SP core. We compare the silicon area and power of eight SkylakeSP [7] cores with fourty ibex [5] cores, representing the two scenario. The numbers for ibex cores comes from synthesis results with Synopsys Design Compiler. The numbers for SkylakeSP comes from our best estimation based on its die shots and thermal design power.

The results are shown in Table II. Employing extra CPUs in DCache for the column selection tasks costs 40x more silicon area and 2400x more power than employing computational storage disks.

VI. Related work

There has been several work [8], [9] trying to modernize the computing infrastructure for global CERN/HEP collaboration. Girone summarized the unprecedented computing and data challenges posed by hugely increased collision rates after the HLLHC upgrade [9]. Rocha et. al. reproduced the famous Higgs boson analysis with Kubernetes and containers to showcase the power of container and computing orchestration with modern computing management systems in exploiting data-level parallelism. However, what is missing is a thorough reevaluation of the design choices made in building and developing the HEP data sites. which is the essence of this paper.

On the other hand, the academia has witnessed several point studies of applying computational storage for file system [10]–[15] or database [10], [13], [16]–[21] applications. These work feature performance speedup from employing computational storage but did not consider the implications in terms of the data center as a whole, which is one of the goals in this work.

VII. Summary

We built a throughput-oriented performance model for tier-2 data centers, the workhorses in HEP grid, to evaluate different upgrade options for improving analysis capacity. We find that computational storage is the out-of-box solution that breaks the traditional 'Liebig's Barrel' dilemma and provide the most cost-efficient and power-efficient upgrades, because its ability to decrease amount of data to process for later stages.

Out future work includes prototyping a computational storage disks [22] as well as architecting a domain-specific processor for high-energy physics in-storage computing.

References

[1] "The grid: A system of tiers," https://home.cern/science/computing/grid-system-tiers.

[2] S. Chatrchyan, V. Khachatryan, A. M. Sirunyan, A. Tumasyan, W. Adam, E. Aguilo, T. Bergauer, M. Dragicevic, J. Erö, C. Fabjan *et al.*, "Observation of a new boson at a mass of 125 gev with the cms experiment at the lhc," *Physics Letters B*, vol. 716, no. 1, pp. 30–61, 2012.

[3] "Higgsexample20112012," https://github.com/cms-opendata-analyses/HiggsExample20112012.

[4] "dcache: distributed storage for scientific data," https://www.dcache.org/.

[5] P. D. Schiavone, F. Conti, D. Rossi, M. Gautschi, A. Pullini, E. Flamand, and L. Benini, "Slow and steady wins the race? a comparison of ultra-low-power risc-v cores for internet-of-things applications," in *2017 27th International Symposium on Power and Timing Modeling, Optimization and Simulation (PATMOS)*. IEEE, 2017, pp. 1–8. [Online]. Available: https://github.com/lowRISC/ibex

[6] "Spike RISC-V ISA Simulator," https://github.com/riscv/riscv-isa-sim.

[7] "Skylake SP Die Shot," https://i.imgur.com/Na64wWe.jpg.

[8] R. Rocha and L. Heinrich, "Reperforming a nobel prize discovery on kubernetes."

[9] M. Girone, "Reperforming a nobel prize discovery on kubernetes."

[10] Z. István, D. Sidler, and G. Alonso, "Caribou: intelligent distributed storage," *Proceedings of the VLDB Endowment*, vol. 10, no. 11, pp. 1202–1213, 2017.

[11] M. Ajdari, P. Park, J. Kim, D. Kwon, and J. Kim, "Cidr: A cost-effective in-line data reduction system for terabit-per-second scale ssd arrays," in *2019 IEEE International Symposium on High Performance Computer Architecture (HPCA)*. IEEE, 2019, pp. 28–41.

[12] M. Ajdari, P. Park, D. Kwon, J. Kim, and J. Kim, "A scalable hw-based inline deduplication for ssd arrays," *IEEE Computer Architecture Letters*, vol. 17, no. 1, pp. 47–50, 2017.

[13] Z. Ruan, T. He, and J. Cong, "Insider: Designing in-storage computing system for emerging high-performance drive," in *2019 USENIX Annual Technical Conference (USENIX ATC 19)*, 2019, pp. 379–394.

[14] J. LeFevre and N. Watkins, "Skyhook: Programmable storage for databases." Boston, MA: USENIX Association, Feb. 2019.

[15] C. Huang, H. Simitci, Y. Xu, A. Ogus, B. Calder, P. Gopalan, J. Li, and S. Yekhanin, "Erasure coding in windows azure storage," in *Presented as part of the 2012 USENIX Annual Technical Conference (USENIX ATC 12)*, 2012, pp. 15–26.

[16] B. Gu, A. S. Yoon, D.-H. Bae, I. Jo, J. Lee, J. Yoon, J.-U. Kang, M. Kwon, C. Yoon, S. Cho *et al.*, "Biscuit: A framework for near-data processing of big data workloads," in *ACM SIGARCH Computer Architecture News*, vol. 44, no. 3. IEEE Press, 2016, pp. 153–165.

[17] L. Woods, Z. István, and G. Alonso, "Ibex: an intelligent storage engine with support for advanced sql offloading," *Proceedings of the VLDB Endowment*, vol. 7, no. 11, pp. 963–974, 2014.

[18] G. Koo, K. K. Matam, H. Narra, J. Li, H.-W. Tseng, S. Swanson, M. Annavaram *et al.*, "Summarizer: trading communication with computing near storage," in *Proceedings of the 50th Annual IEEE/ACM International Symposium on Microarchitecture*. ACM, 2017, pp. 219–231.

[19] I. Jo, D.-H. Bae, A. S. Yoon, J.-U. Kang, S. Cho, D. D. Lee, and J. Jeong, "Yoursql: a high-performance database system leveraging in-storage computing," *Proceedings of the VLDB Endowment*, vol. 9, no. 12, pp. 924–935, 2016.

[20] Y. Fang, C. Zou, A. J. Elmore, and A. A. Chien, "Udp: a programmable accelerator for extract-transform-load workloads and more," in *2017 50th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO)*. IEEE, 2017, pp. 55–68.

[21] Y. Fang, C. Zou, and A. A. Chien, "Accelerating raw data analysis with the accorda software and hardware architecture," *Proceedings of the VLDB Endowment*, vol. 12, no. 11, pp. 1568–1582, 2019.

[22] C. Zou and A. A. Chien, "Assasin: Architecture support for stream computing to accelerate computational storage," in *Under review*.