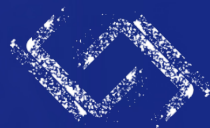




国家高能物理科学数据中心

National HEP Science Data Center

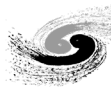


高能所计算中心

IHEP Computing Center



国家高能物理科学数据中心
National HEP Science Data Center



中国科学院高能物理研究所
Institute of High Energy Physics, Chinese Academy of Sciences



高能所计算中心
IHEP Computing Center

2025年超级陶粲装置研讨会

高能物理数据处理可计算存储系统设计与实现

中国科学院高能物理研究所

程耀东

2025年7月4日

CONTENTS



技术背景



系统实现



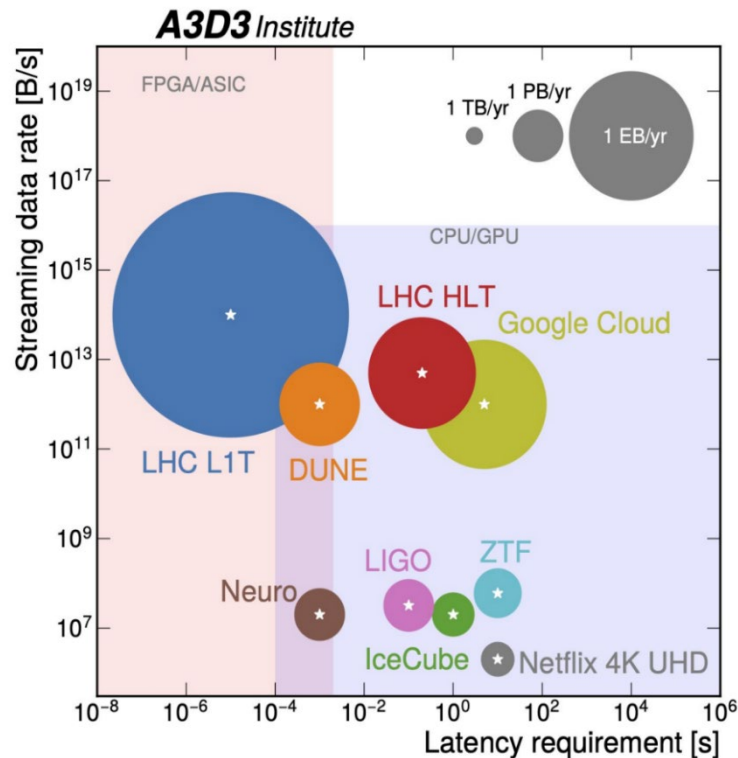
示范应用



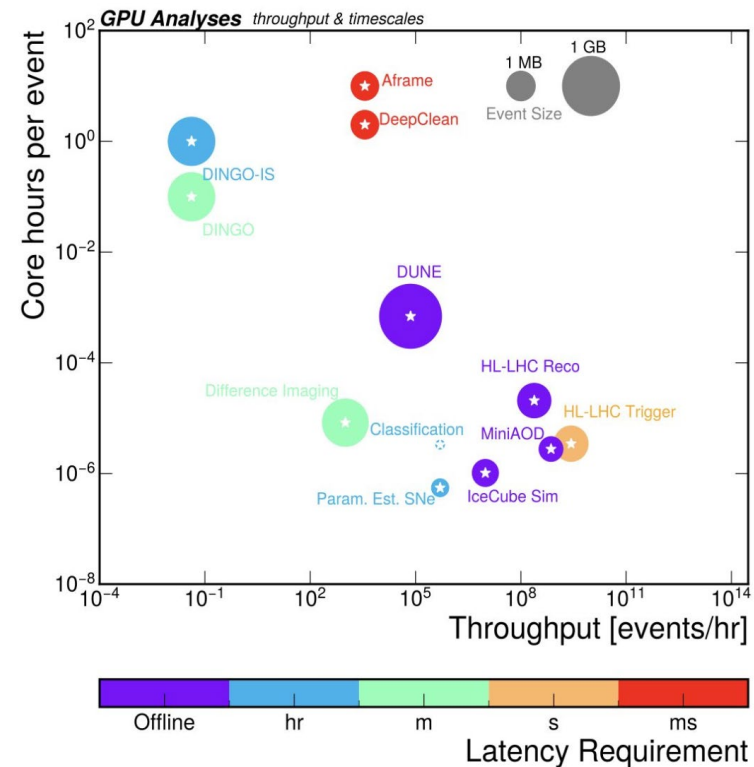
未来展望

高能物理的计算需求与特点

- ✓ 高数据率
- ✓ 低延迟
- ✓ 复杂的数据结构
- ✓ 复杂的处理算法
- ✓ 庞大的数据存储量



来源: [A3D3](#)



[ArXiv:2306.0816](#)

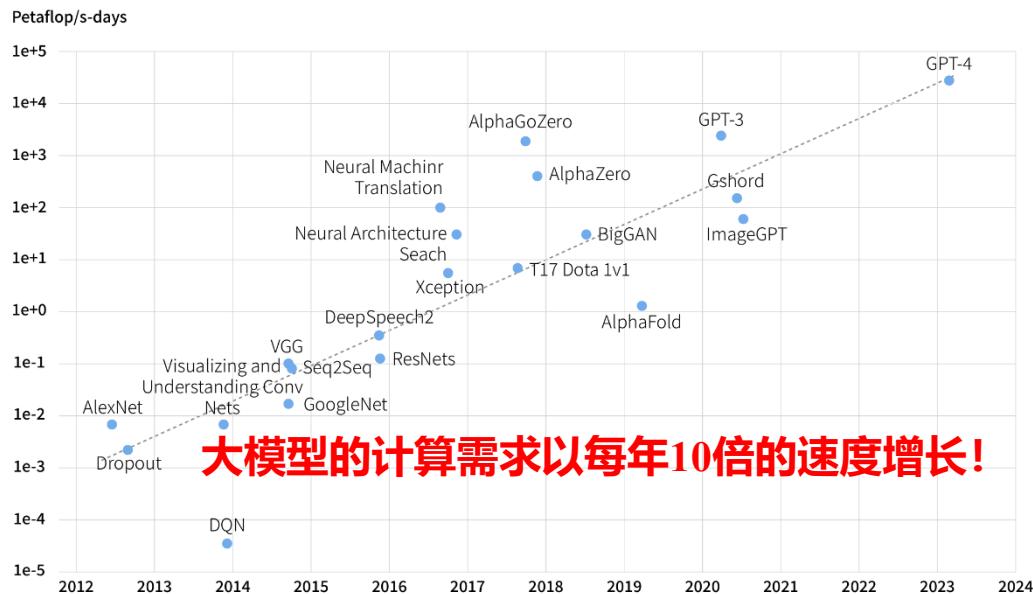
CPU性能的增长速度无法跟上应用需求的增加



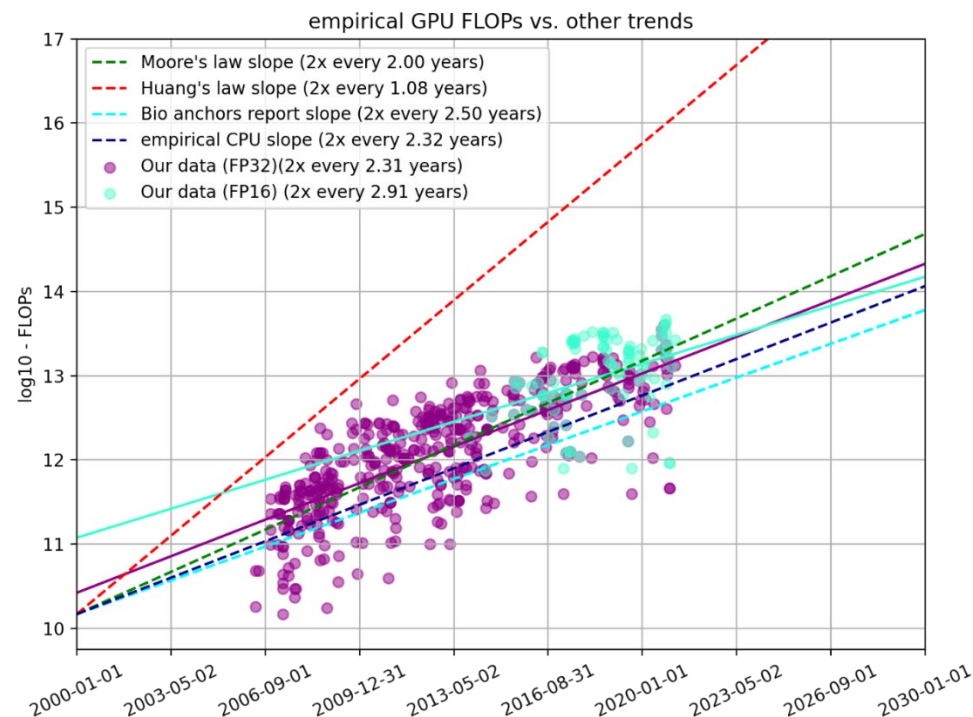
大量使用协处理器 (GPU、FPGA等)



硬件的发展跟不上计算需求



<https://arxiv.org/pdf/2408.14158>



<https://epoch.ai/blog/trends-in-gpu-price-performance>

CPU 性能 – 摩尔定律 (Moore' s Law); GPU 性能 – 黄氏定律 (Huang' s Law)

“翻一番周期”：GPU最快 (1.08年)，网络和 SSD (≈1.5 年)，CPU (≈2 年)，DRAM 带宽 (≈4.3 年)



传统通用架构面临的挑战

高功耗与低能效

GPU服务器功耗达数千瓦，能效比低，如NVIDIA B200单卡功耗超1200W，数据搬运能耗是计算的650倍。传统架构在处理大规模数据时，**能耗问题**成为制约其发展的关键因素。

延迟抖动与性能不稳定

操作系统调度与**多任务争抢资源**，导致确定性性能缺失，如实时处理中微秒级抖动影响精度。在高能物理等对精度要求极高的领域，性能的不稳定可能导致实验结果的偏差。



低效率与带宽瓶颈

冯诺依曼架构下数据频繁搬运，带宽瓶颈显著，如GPU显存与存储系统间延迟影响计算效率。数据搬运的低效率导致计算资源的浪费，难以满足现代科研对实时性和高吞吐量的需求。

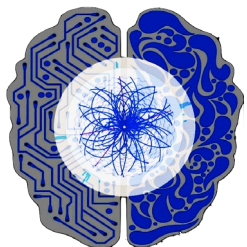
扩展性差与成本高昂

存储与计算分离，EB级数据场景下横向扩展成本高，难以支持动态负载均衡。传统架构在面对大规模数据存储和计算需求时，扩展性差成为其难以克服的瓶颈。

解决方案：专用计算+应用优化！



国内外研究现状



FastML Lab

Real-time and accelerated ML for fundamental sciences



CERN FastML实验室



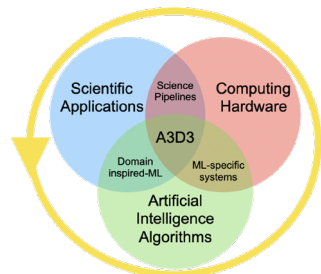
FPGA: 现场可编程门阵列

- ✓ 硬件: 低功耗、低延迟、可变浮点精度、性能稳定
- ✓ 软件: 基于FPGA的AI编程框架, 易用性越来越好

BNL: 研制基于FPGA的FLEX硬件, 用于数据读出和处理

IHEP: 研制基于FPGA的可计算存储服务器

- ✓ 基于PCIe标准, 方便灵活, 内置内存和存储, 适合数据处理

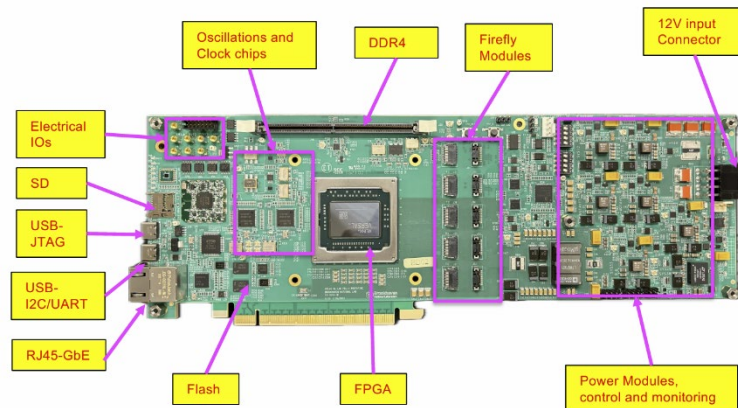


美国A3D3研究所

Accelerated AI
Algorithms for Data-
Driven Discovery



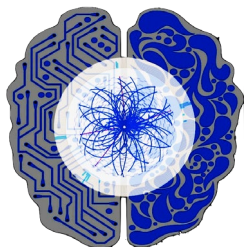
高能物理领域发起的的
开源项目



BNL
FLEX



国内外研究现状

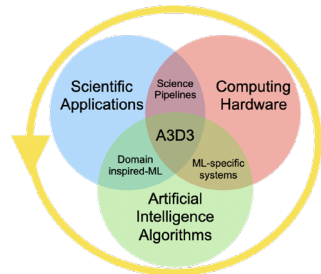


FastML Lab

Real-time and accelerated ML for fundamental sciences



CERN FastML实验室



美国A3D3研究所

**Accelerated AI
Algorithms for Data-
Driven Discovery**



FPGA: 现场可编程门阵列

- ✓ 硬件: 低功耗、低延迟、可变浮点精度、性能稳定
- ✓ 软件: 基于FPGA的AI编程框架, 易用性越来越好

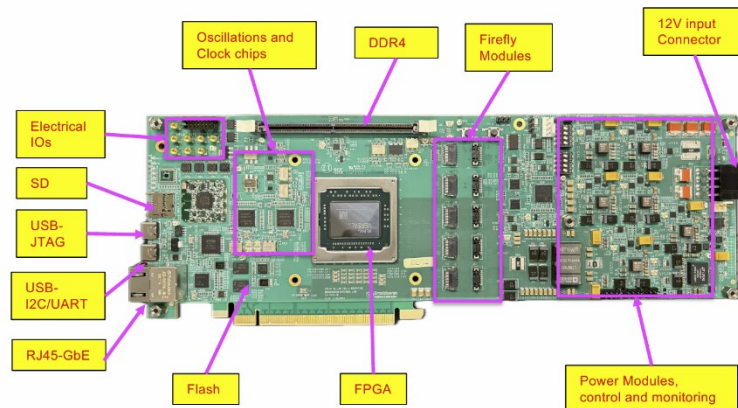


高能物理领域发起的的
开源项目

BNL: 研制基于FPGA的FLEX硬件, 用于数据读出和处理

IHEP: 研制基于FPGA的可计算存储服务器

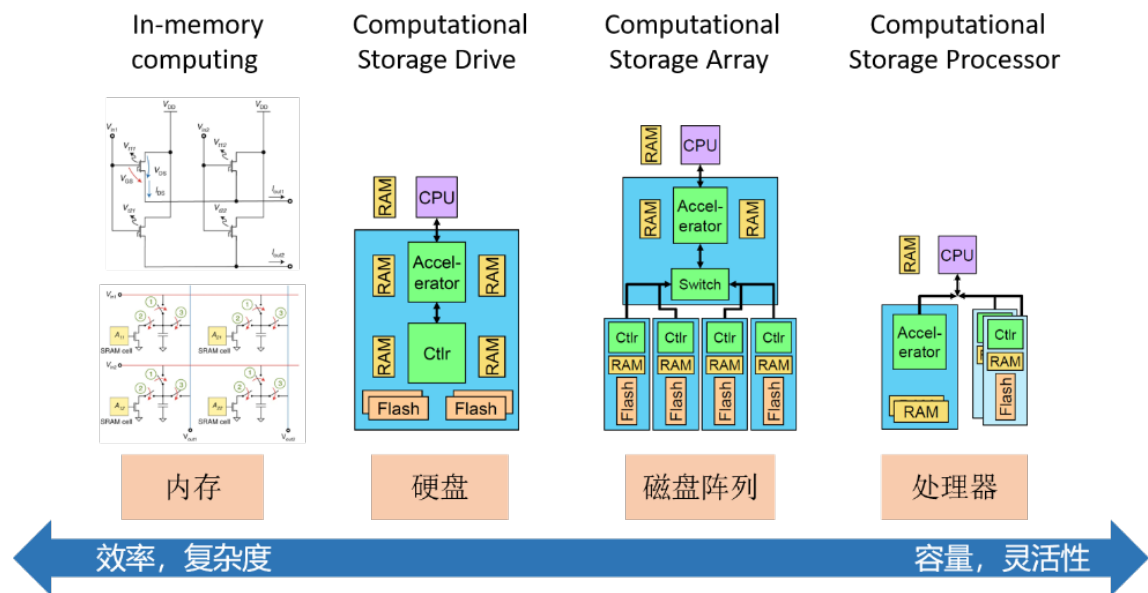
- ✓ 基于PCIe标准, 方便灵活, 内置内存和存储, 适合数据处理



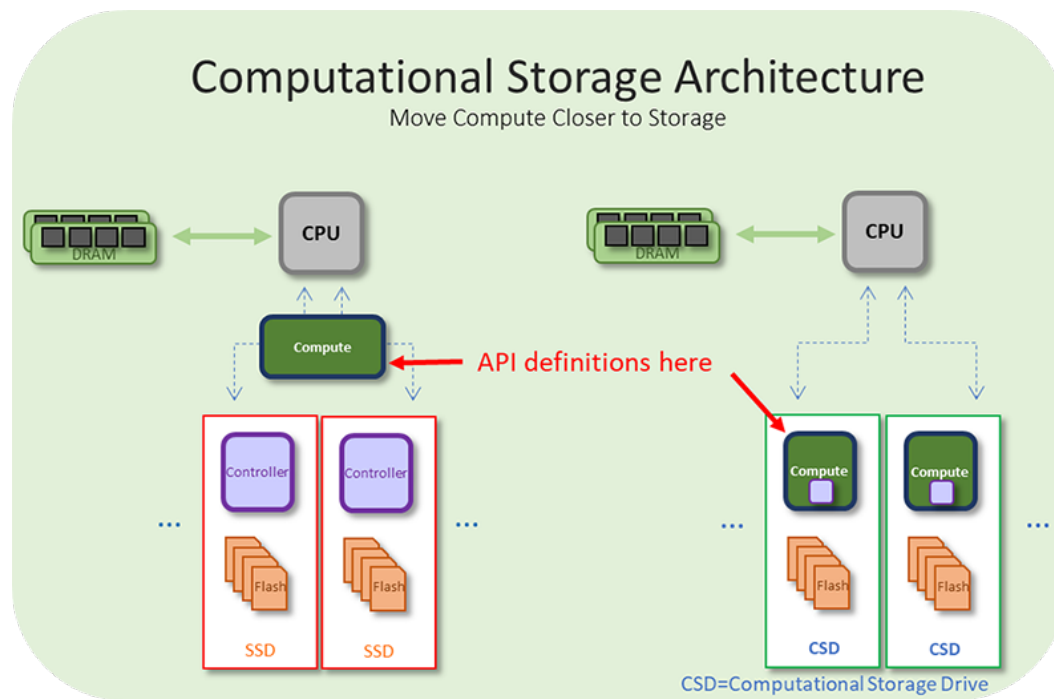
**BNL
FLEX**



可计算存储技术



可计算存储技术应用到数据处理中将带来重大突破！



- ✓ 可计算存储（Computational Storage）作为一种“存算一体”技术，将部分计算能力下沉到数据存储部分，减少数据移动，提高计算效率



可计算存储产品

- SmartSSD CSD

- Samsung SmartSSD, ScaleFlux CSD, NGD Systems CSD
- 深圳大普微 (dapuStor)

- CSP加速卡

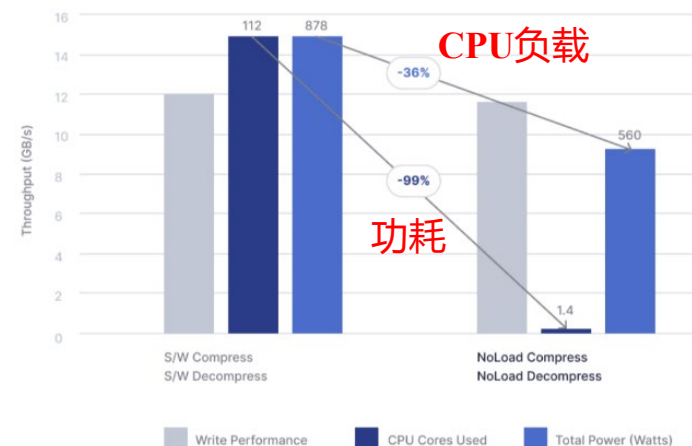
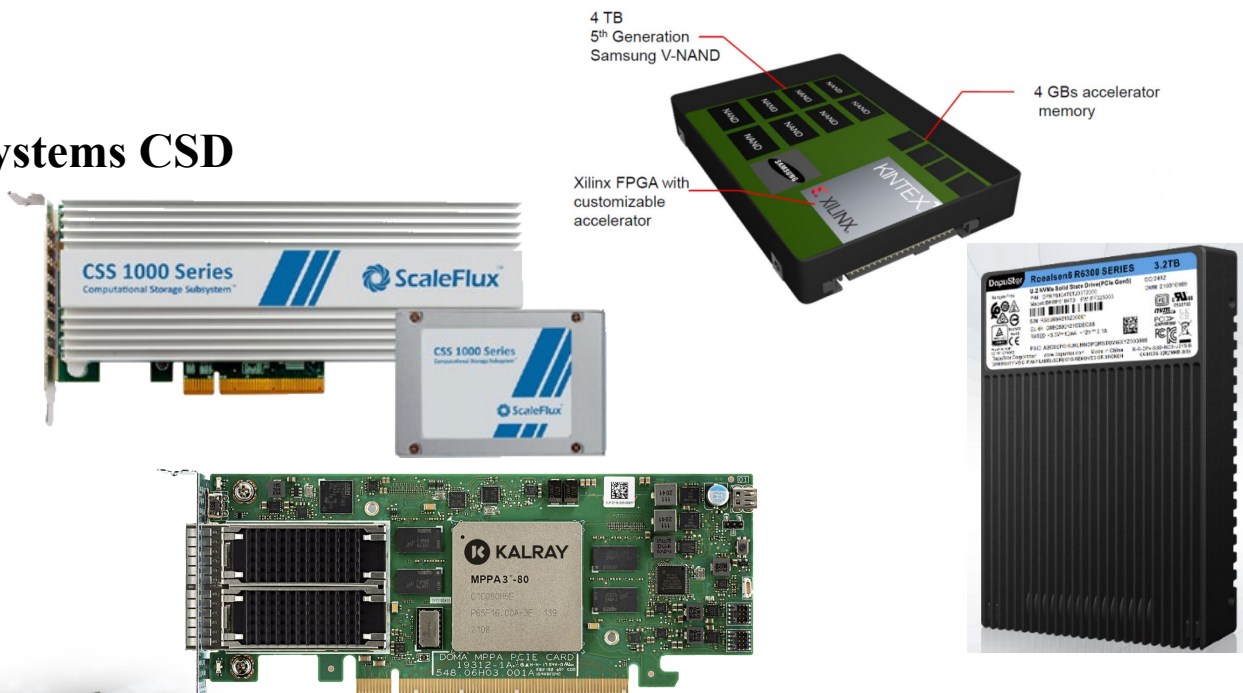
- Pliops XDP, Eideticom NoLoad CSP, Kalray Smart Storage Adapter, Solidigm ...
- 深存科技 (StorageX)

- 主要实现方式

- 在存储设备中嵌入FPGA或者多颗ARM核

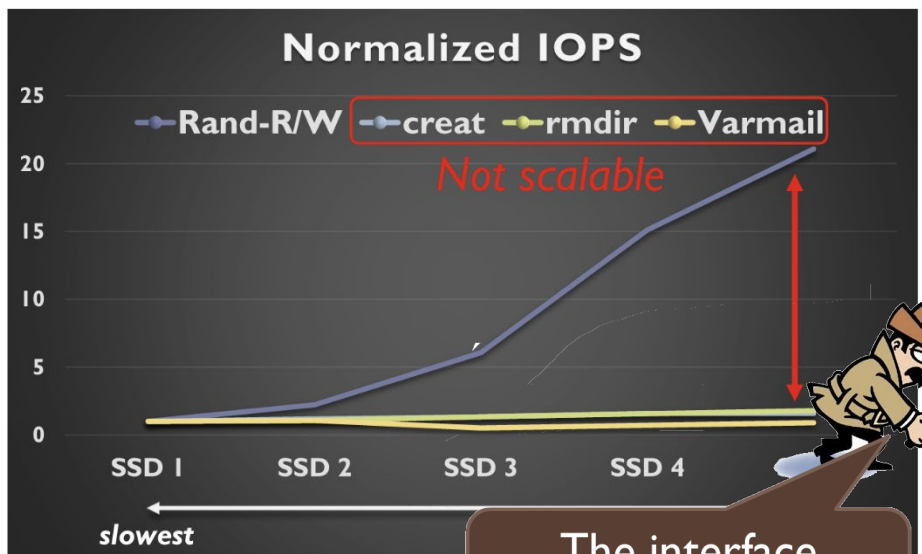
- 主要功能

- 压缩、检索、AI推理加速、...





文件系统卸载: KevinFS



Faster devices *cannot* make

The interface should be changed!

Metadata Management



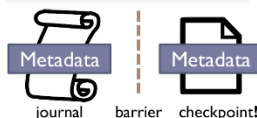
I/O Amplification

Space Management

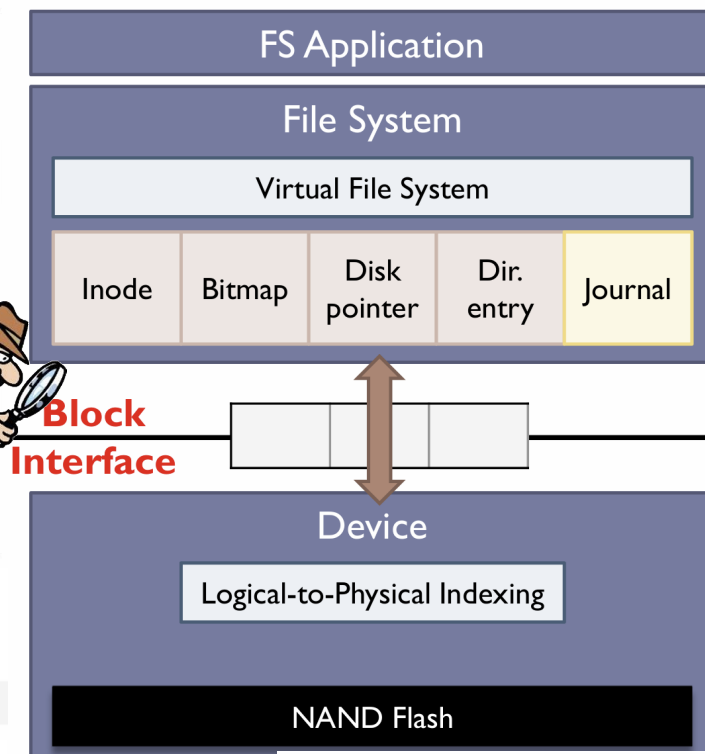


Fragmentation

Crash Consistency

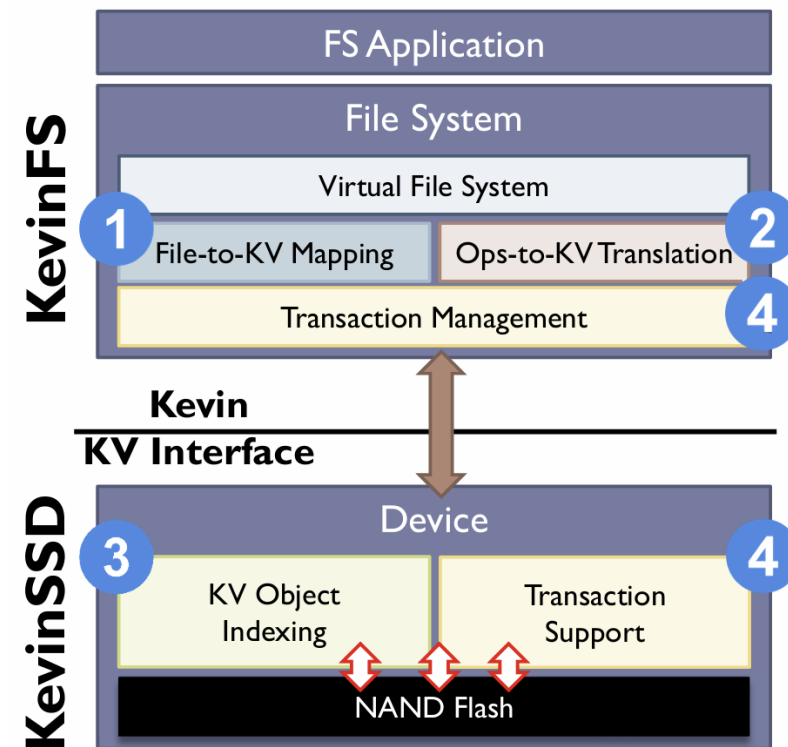


Journaling I/O
(double-writes, I/O delay)



Checklist

- Reducing Meta I/O Amp.? ✓
- Resilient to Fragmentation? ✓
- Reducing Journaling I/O? ✓



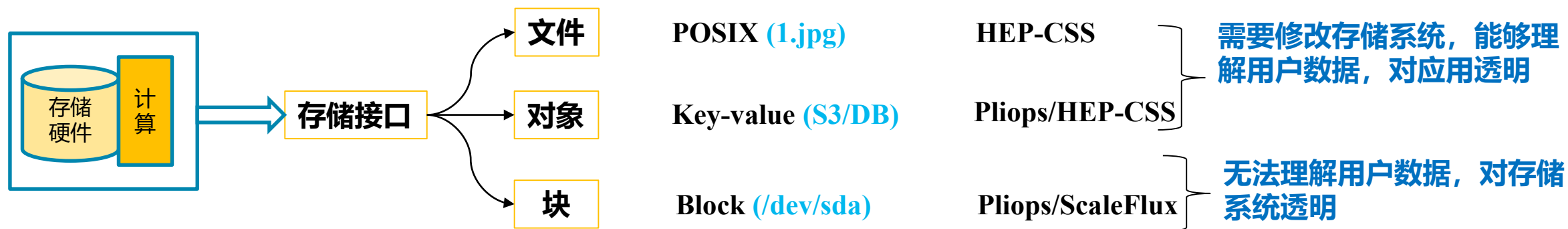
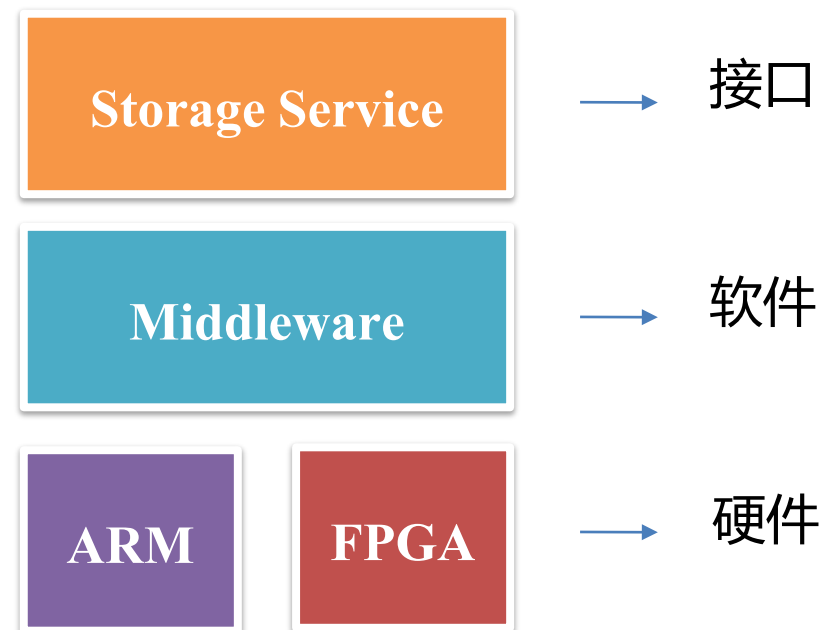
Data transfer for indexing KV objects occurs in storage
→ Lightweight FS with low FS traffic

Koo J, Im J, Song J, et al. Modernizing file system through in-storage indexing, OSDI'21
<https://github.com/dgist-datalab/kevin>



高能物理可计算存储系统

- 采用CSP的方式——**容量大、灵活性好**
- 将“CPU处理效率低下、GPU不合适”的负载卸载到专用芯片
- 硬件：低功耗专用芯片——ARM+FPGA
- 软件：异构计算——软件硬件化将成为常态，基于ARM/FPGA的软件生态，EB级分布式存储
- 接口：兼容标准化的数据访问接口——POSIX, NFS, http, XRootd, Restful, ...



“ 总体架构



“ 硬件研制

✓ 基于FPGA架构，自主研制**可计算存储处理器**

- FPGA: Xilinx Kintex UltraScale+ KU5P
- PCI-e Gen 3 x2, 内置4GB DDR4
- 支持PR功能，动态更新FPGA算法

✓ 基于低功耗ARM，自主研制**可计算存储服务器**

- NXP LS1046A, Cortex®-v8 A72, 4 cores支持DPAA
- 2个万兆, 2个千兆光纤以太网接口
- 12个SATA接口
- 高密度、低功耗、绿色节能

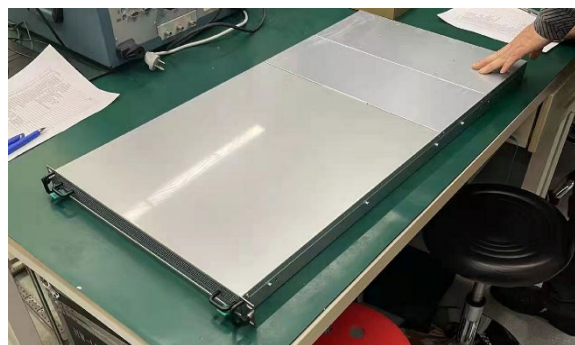
	Device Name	KU3P	KU5P
Logic	System Logic Cells (K)	356	475
	CLB Flip-Flops (K)	325	434
	CLB LUTs (K)	163	217
Memory	Max. Distributed RAM (Mb)	4.7	6.1
	Total Block RAM (Mb)	12.7	16.9
	UltraRAM (Mb)	13.5	18.0
Clocking	Clock Mgmt Tiles (CMTs)	4	4
Integrated IP	DSP Slices	1,368	1,824
	PCIE4 (PCIe® Gen3 x16)	1	1
	PCIE4C (PCIe® Gen3 x16 / Gen4 x8)	0	0
	150G Interlaken	0	0
I/O	100G Ethernet w/ KR4 RS-FEC	0	1
	Max. Single-Ended HD I/Os	96	96
	Max. Single-Ended HP I/Os	208	208
	GTH 16.3 Gb/s Transceivers	0	0
Speed Grades	GTY 32.75 Gb/s Transceivers	16	16
	Extended ⁽¹⁾	-1 -2 -2L -3	-1 -2 -2L -3
	Industrial	-1 -1L -2	-1 -1L -2



主板及相关部件



可计算存储数据处理器CSP



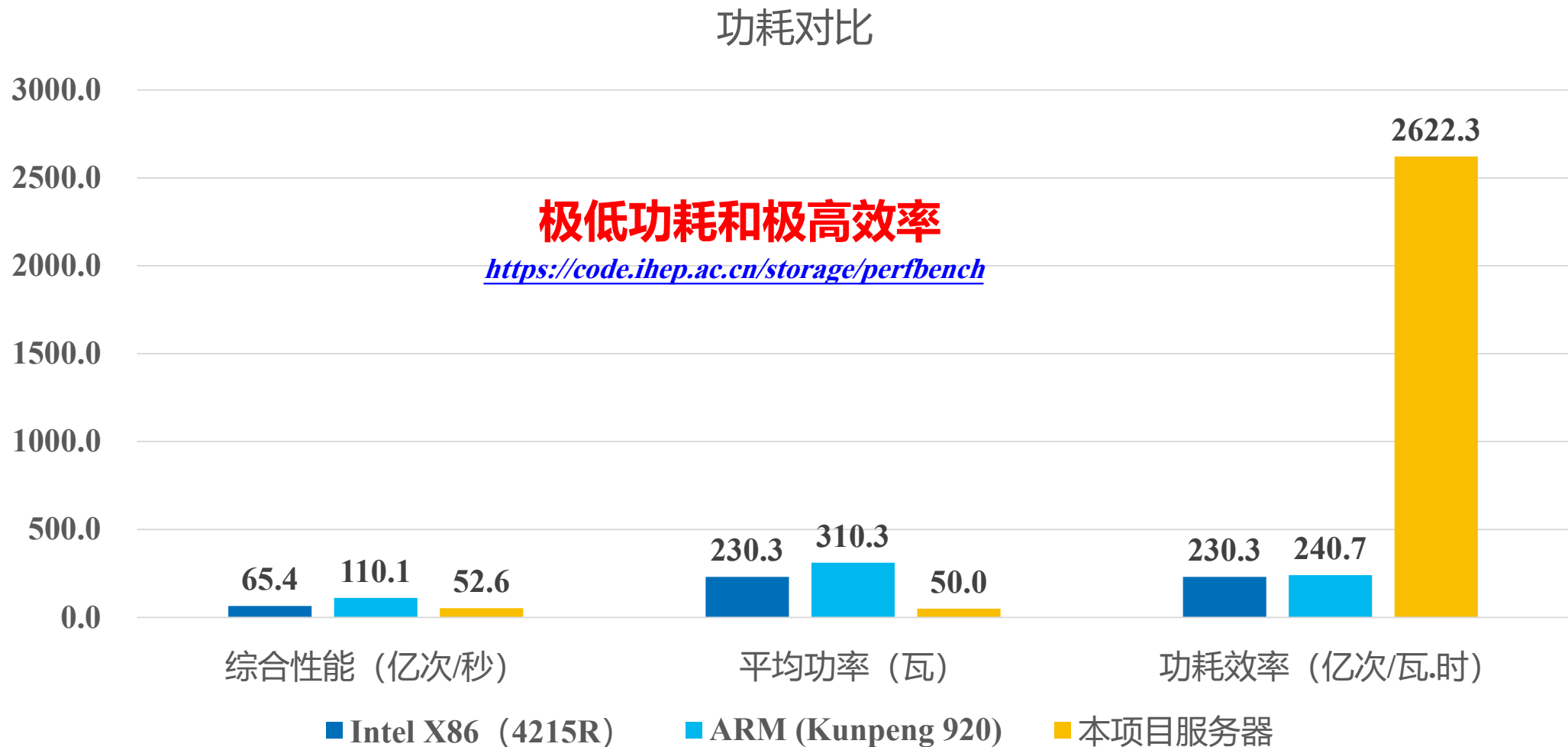
可计算存储服务器HPS



上线的设备



性能与功耗



“可计算存储软件

解决了可计算存储的“标准化”问题

基于CERN EOS进行扩展，2025年4月并入EOS的主线代码（<https://gitlab.cern.ch/dss/eos>）

存储客户端：

兼容标准访问协议，调用时加个参数即可，比如：

```
cat runid.txt&css_alg=sort;
```

```
open( "run011.root&css_app=decode" ,  
O_RDONLY)
```

计算任务调度：

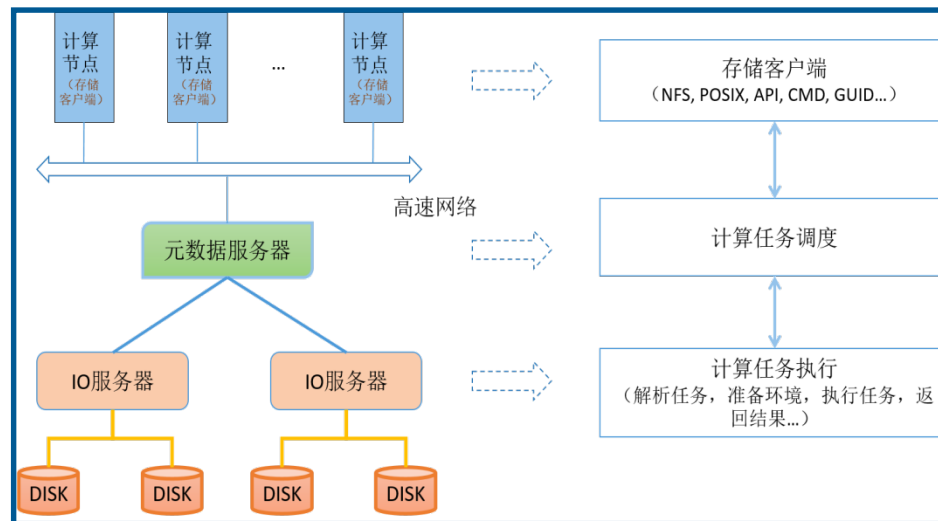
通过元数据查找物理文件位置

IO服务：

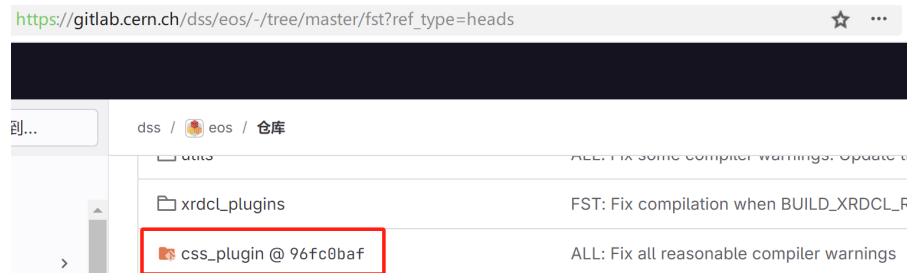
任务解析，执行等

框架特点

兼容性好、**EB级可扩展**、易于支持各类应用与算法



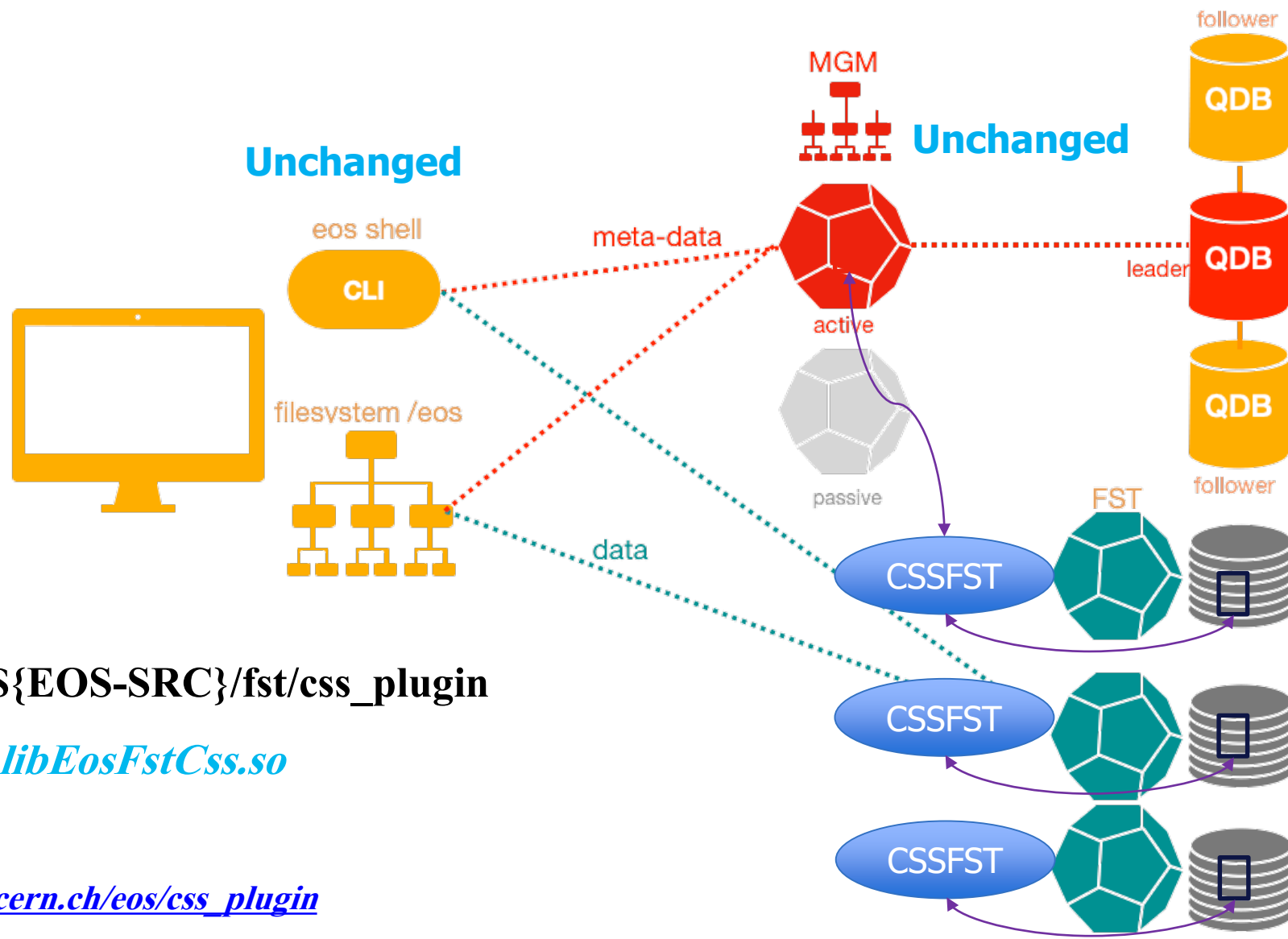
存算一体化，EB级可扩展



https://gitlab.cern.ch/eos/css_plugin



软件架构与设计



代码位置: `${EOS-SRC}/fst/css_plugin`

编译生成: *libEosFstCss.so*

https://gitlab.cern.ch/eos/css_plugin



软件部署

- 修改/etc/xrd.cf.fst配置文件
~~`xrootd.fslib -2 libXrdEosFst.so`~~
`xrootd.fslib -2 libEosFstCss.so -2 libXrdEosFst.so`
- 修改FST节点上的配置文件/etc/eoscss.conf，定制可计算存储函数CSF

```
{ "sort" : {  
  "name" : "sort",  
  "path" : "/usr/local/libexec/cssfst/sort.sh",  
  "out" : false },  
  "km2a_decode" : {  
    "name" : "km2a-decode",  
    "path" : "/usr/local/libexec/cssfst/km2a-decode.sh",  
    "out" : true,  
    "postfix" : "root"},  
  "zstd": {  
    "name": "zstd",  
    "path" : "/usr/local/libexec/cssfst/zstd.sh",  
    "out" : true,  
    "postfix" : "zst"  
  }  
}
```

```
#cat sort.sh  
/usr/bin/sort -n $1 > $2  
if [ $? -eq 0 ];then  
  echo "EXEC_SUCCESS"  
  exit 0  
echo "EXEC_FAILED"  
exit 1
```

```
#cat km2a-decode.sh  
container=/usr/local/libexec/cssfst/km2adecode.sif  
apptainer exec --bind $dirn $container  
/root/km2a/km2a-decode/decode_sort $1 $2  
if [ $? -eq 0 ];then  
  echo "EXEC_SUCCESS"  
  exit 0  
echo "EXEC_FAILED"  
exit 1
```

“ 如何使用

✓ 两种方法来使用可计算存储系统，一个是标准的xrdcp命令，一个是专用的cssclient

1) 使用xrdcp命令，将CSF的名字附加在文件名后面，例如：

```
#xrdcp root://eosbak02.ihep.ac.cn//eos/user/chyd/data.txt?css=sort -
```

2) 使用cssclient，这个工具调用了XrdPosixXrootd

```
#Export EOS_MGM_URL=root://eosbak02.ihep.ac.cn/
```

```
[root@eosbak02 css]# xrdcp root://eosbak02.ihep.ac.cn//eos/user/chyd/data.txt -
1-01, 266.0
1-06, 145.9
2-04, 188.8
1-03, 183.1
2-01, 122.2
2-03, 199.1
1-04, 119.3
1-01, 222.3
2-05, 183.6
1-05, 180.3
1-04, 155.8
1-02, 190.5
1-20, 223.5
2-02, 130.9
1-02, 189.1
1-03, 185.1
2-03, 199.2
[204B/204B][100%][=====][204B/s]
```

传统的模式，结果显示文件的内容

```
[root@eosbak02 css]# xrdcp root://eosbak02.ihep.ac.cn//eos/user/chyd/data.txt?css=sort -
1-01, 222.3
1-01, 266.0
1-02, 189.1
1-02, 190.5
1-03, 183.1
1-03, 185.1
1-04, 119.3
1-04, 155.8
1-05, 180.3
1-06, 145.9
1-20, 223.5
2-01, 122.2
2-02, 130.9
2-03, 199.1
2-03, 199.2
2-04, 188.8
2-05, 183.6
[204B/204B][100%][=====][204B/s]
```

可计算存储的模式，结果显示排序后的内容



典型应用-KM2A Decode

```
bash-4.2$ time aptainer exec /home/chyd/km2adecode.sif /home/km2a/decode_sort_xrootd.sh /eos/user/c/chyd/km2a/20220701003242.670.dat /eos/user/c/chyd/km2a/20220701003242.670.dat.root
```

real 0m54.875s
user 0m31.467s
sys 0m8.258s

在计算节点上直接运行解码程序

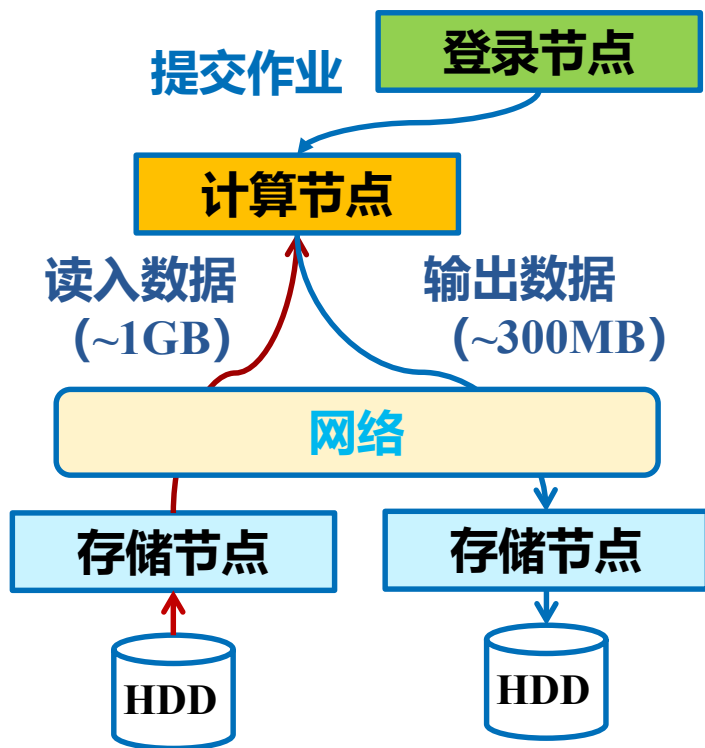
```
bash-4.2$ time cssclient -f /eos/user/chyd/km2a/20220701003242.670.dat -c km2a_decode file '/eos/user/chyd/km2a/20220701003242.670.dat.root' created
```

real 0m26.347s
user 0m0.015s
sys 0m0.015s

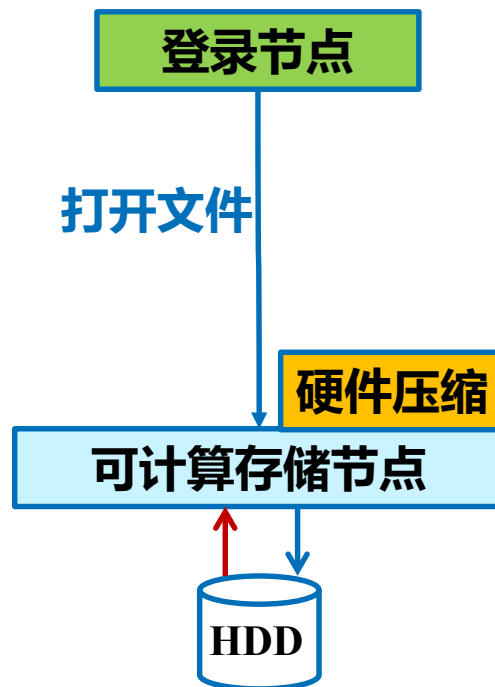
从登录节点调用可计算存储功能，时间节约1/2以上

```
bash-4.2$ eos ls -l /eos/user/chyd/km2a
-rw-r--r-- 1 chyd u07 1001554470 Nov 14 21:22 20220701003242.670.dat
-rw-r--r-- 1 chyd u07 446552932 Apr 20 05:26 20220701003242.670.dat.root
```

传统方式需要从一个IO节点读取文件，计算后将输出文件写入到另外一个IO节点



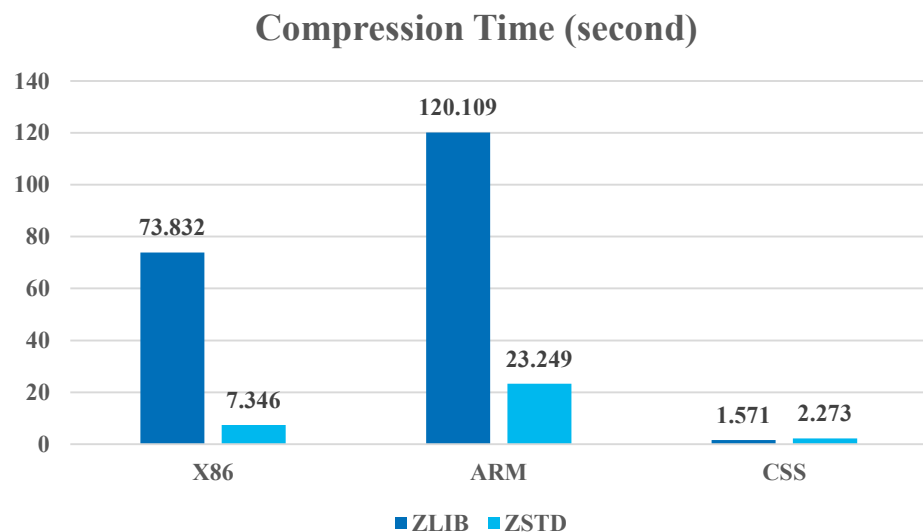
可计算存储方式将任务分配到输入文件所在的IO节点，计算后再写入同一个节点





典型应用-硬件压缩

- ✓ 采用传统的数据压缩算法(比如Zlib) , 采用FPGA硬件加速, 比ARM CPU提高100X, 比Intel CPU提高50X



- ✓ 修改ROOT源代码, 将硬件压缩功能整合到ROOT中, 应用程序只需要指定相应的压缩算法和级别既可以透明调用



Accelerating ROOT compression with Intel ISA-L Library

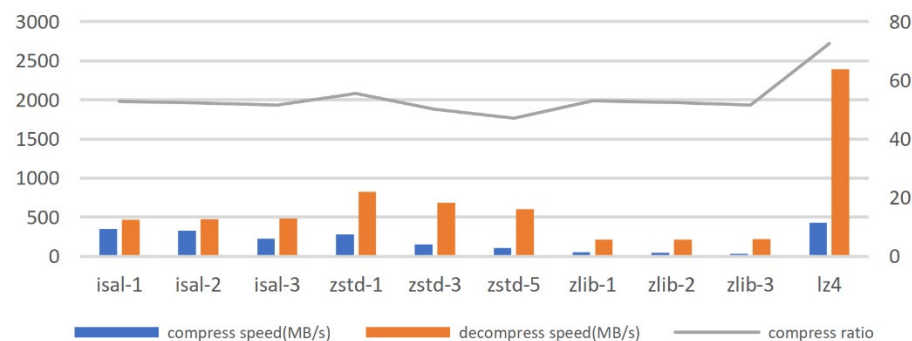
Yu Gao^{1,2}, Yaodong Cheng^{1,2}, Yaosong Cheng¹, Xiaoyu Liu^{1,2}

¹ Institute of High Energy Physics, Chinese Academy of Science, Beijing 100049, China

² University of Chinese Academy of Sciences, Beijing 100049, China



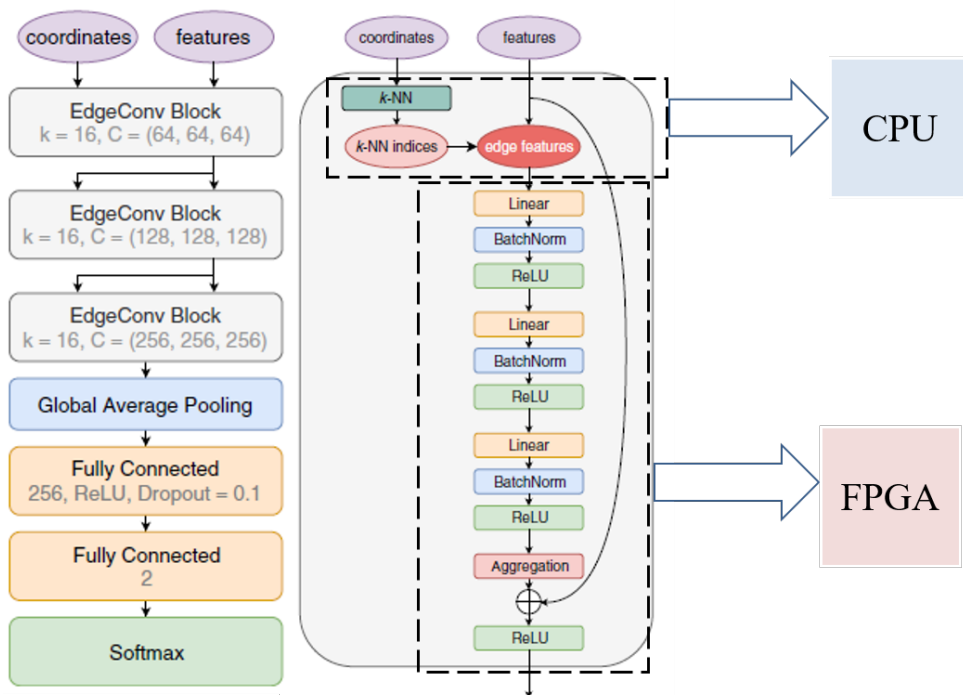
Compression Test on Intel(R) Xeon(R) Silver 4215R CPU@3.2GHz



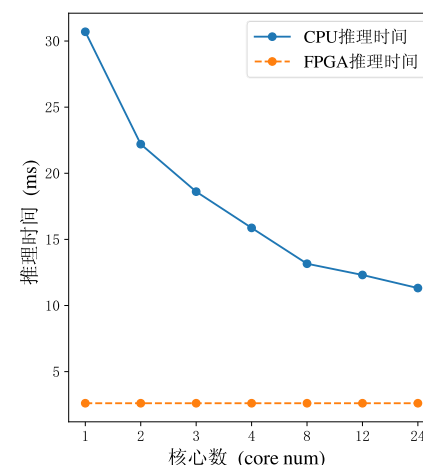


典型应用-ParticleNet

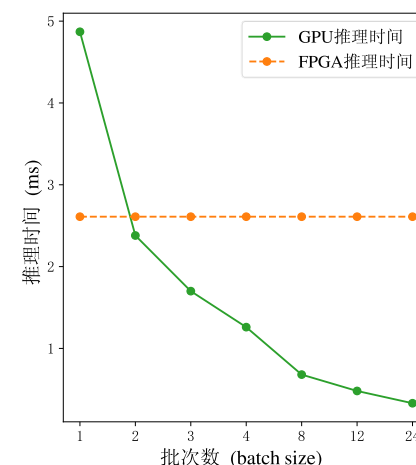
- 粒子物理探测器收集到的一般都是次级粒子，需要通过神经网络算法找到**原初粒子**
- 图卷积神经网络ParticleNet是广泛使用的模型
- 采用FPGA技术，推理时间是**CPU的1/10**，功耗只有**1/8**



硬件平台以及型号	功耗	推理时间
CPU Intel(R) Xeon(R) Silver 4116 CPU @ 2.10GHz (单核)	85W	30ms
HPS/FPGA	34.3W	2.6ms
GPU Tesla V100 (batch size=1)	250W	4.7ms



(a) CPU与FPGA推理时间对比



(b) GPU与FPGA推理时间对比

“典型应用-Particle Transformer

➤ Particle Transformer是新的一种分类模型

➤ 模型优化与压缩

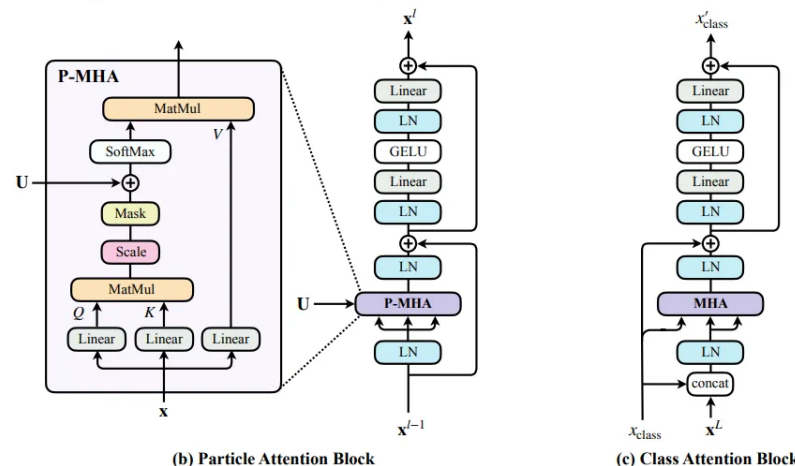
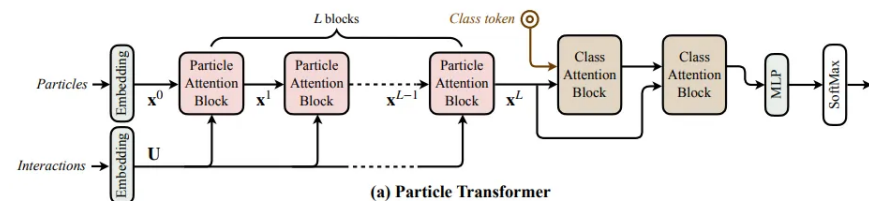
- 量化、剪枝、蒸馏等

Particle Attention Block

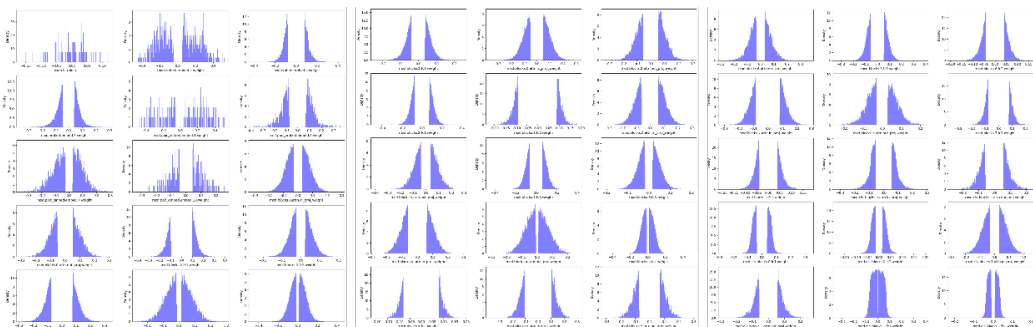
在应用Softmax之前，在P-MHA层中添加粒子相互作用(U) 作为偏置

Class Attention Block

查询矩阵是从类标记导出的，用于建模类到粒子的关系



Bit size	Model size	Accuracy
32-bit	8.17 MB	93.998%
8-bit	2.04 MB	93.986%
4-bit	1.02 MB	93.352%
2-bit	0.51 MB	64.687%



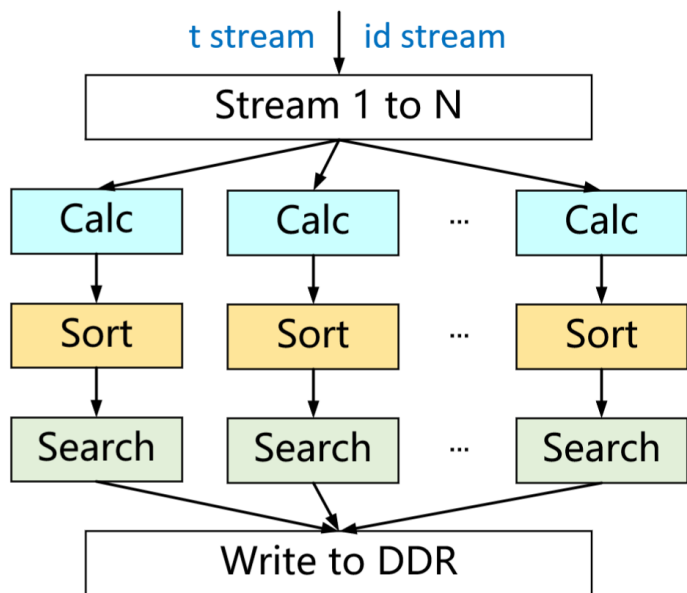
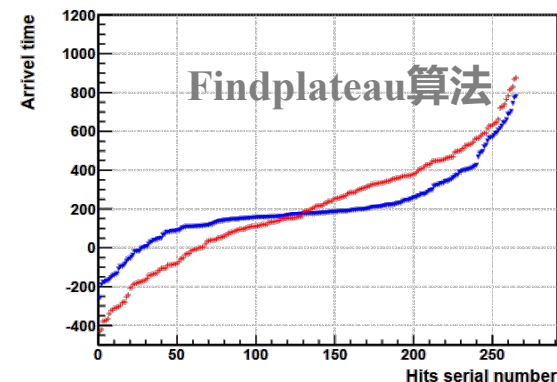
ParT剪枝后:
Size: 4.96M, **48.53%**
精度: 93.998% →
93.936%

“ 典型应用-LHAASO无触发数据处理

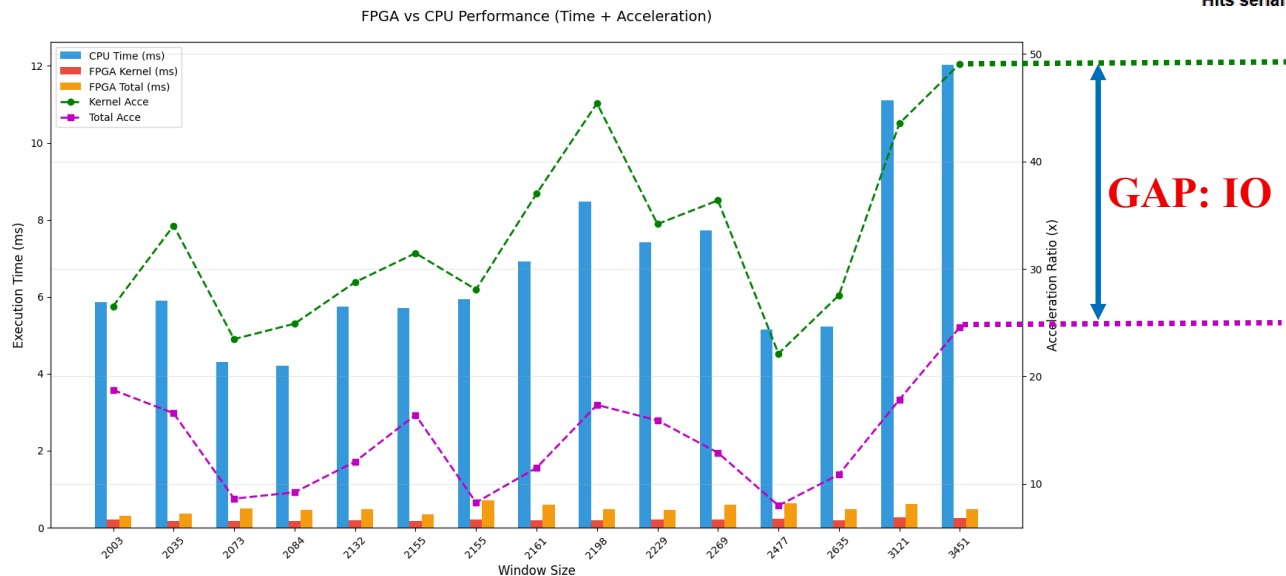
➤ LHAASO无触发数据获取模式是其探测技术的核心创新之一，有效提升了伽马爆等实时天文现象的观测能力，但是计算量和存储量巨大

➤ 核心算法是findplateau，找到并输出事例

➤ 基于FPGA硬件计算，可以大大加速



FPGA并行计算



相对于 CPU 算法，kernel 的加速比约为22.12~49.08

总加速比约为 7.99~24.59。IO性能优化仍在进行



可计算存储的挑战

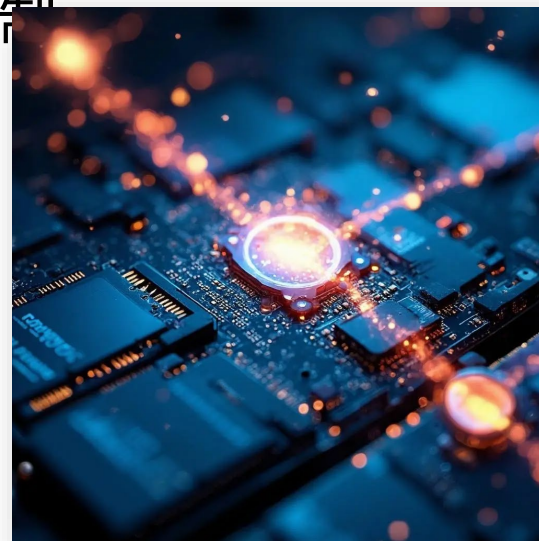
- 还没有真正的“杀手级应用程序”
 - SmartSSD主要做压缩，但是多核CPU性能也不差
 - 推理加速对网络模型尺寸要求很高（剪枝、量化、蒸馏等）
- 复杂性是一个非常令人担忧的问题
 - 复杂性低，效益一般
 - 高复杂性，高效益
- 标准化
 - SNIA制定了CS标准规范和API接口，各种CS硬件的兼容性？
 - 与文件、对象和块存储接口的兼容性
- 安全
 - 恶意CSF、数据泄密、设备DoS攻击等
- 调整生态系统
 - 存储（无生态）+ 计算（有生态）= 可计算存储（有生态）





总结

1. 高能物理计算与AI技术快速发展，急需先进技术架构——领域专用计算与应用优化协同
2. 可计算存储系统能够减少数据搬运，提高计算效率，且可按需定制
3. 已完成高能物理可计算存储软硬件系统原型研制及相关验证
4. 现有设计倾向于采用低功耗组件，拟开发高性能版本
5. 力争找到适合的场景，支持更多的应用，完善生态





中国科学院高能物理研究所
Institute of High Energy Physics Chinese Academy of Sciences

敬请批评指正!

