SuperFS: 高速硬件时代的文件系统

陆游游

清华大学

提纲

一 背景

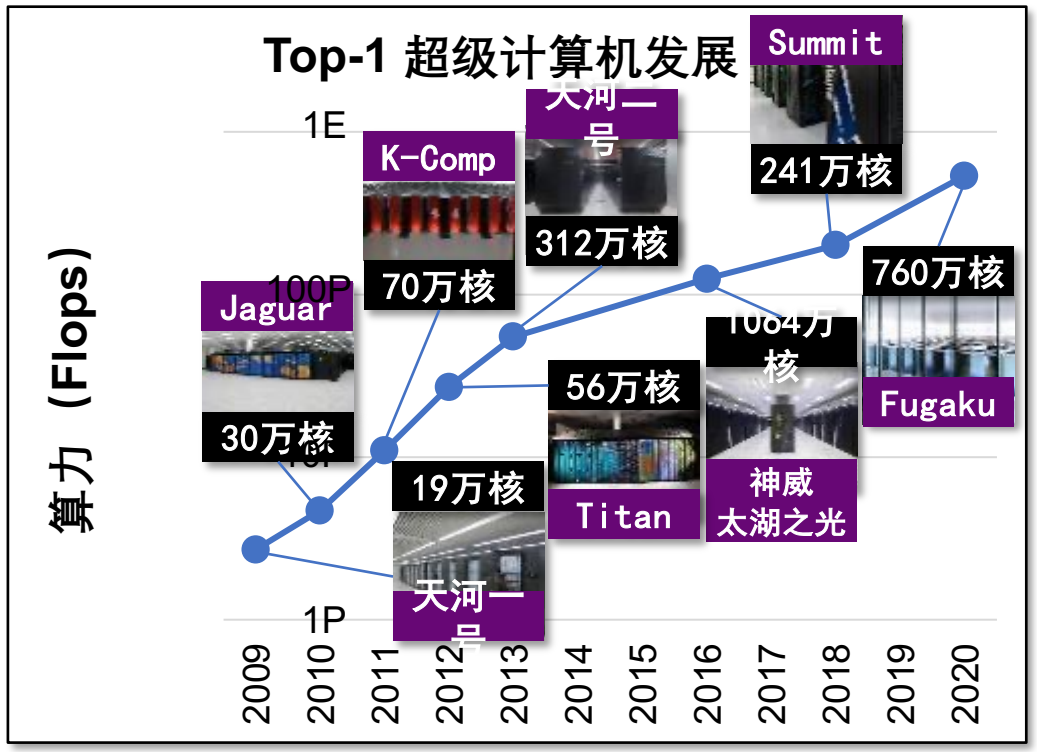
二 文件系统数据部分

三 文件系统元数据部分

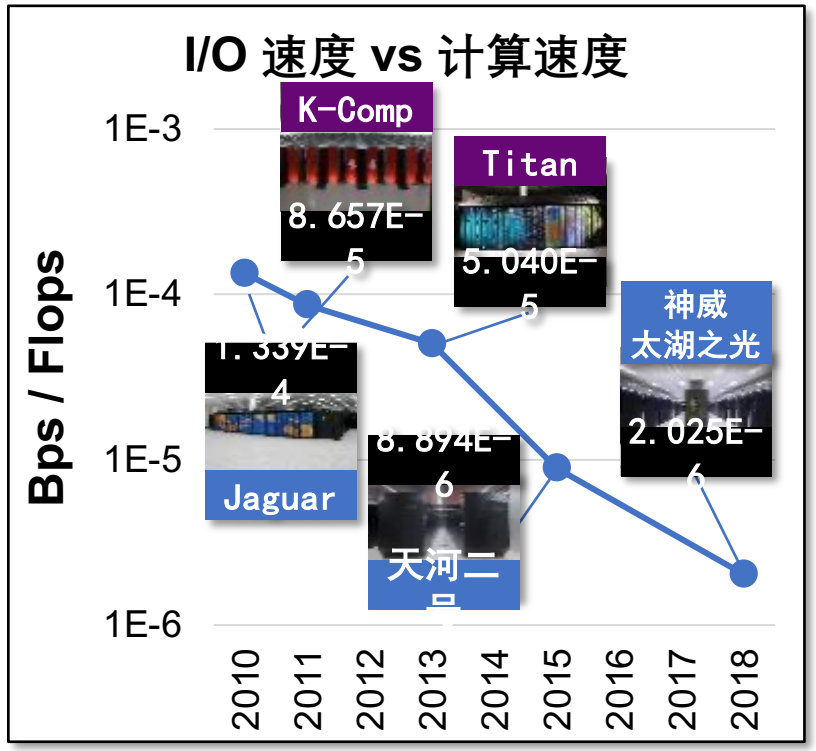
四 总结

存储性能成为瓶颈

□ 超级计算步入千万核时代，存储墙问题日益严峻



超算步入千万核时代，对存储并发需求高



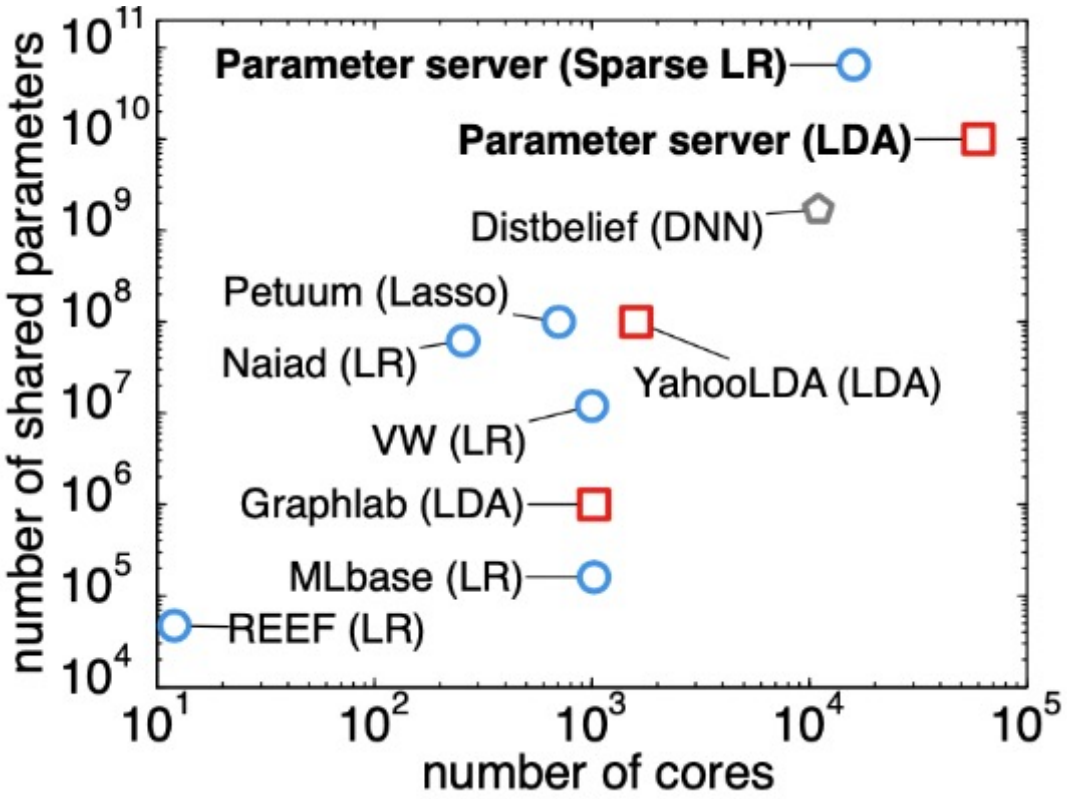
存储发展远远落后计算发展

例子

- ■全机规模的“神图”应用每次需要半小时以上读入超过200TB数据
- ■大地震模拟应用每次需要半小时以上写出超过120TB的数据

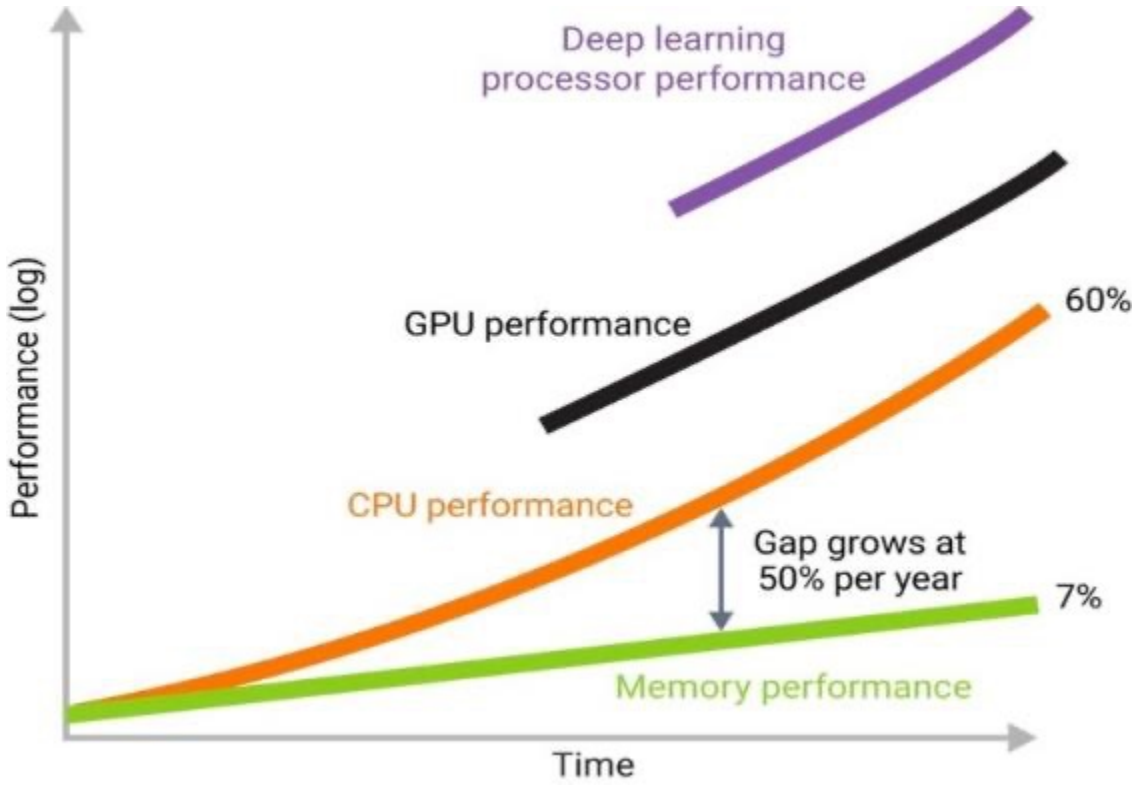
存储性能成为瓶颈

- 存储-计算的需求同时变高



数十万计算核心，百亿级参数

- 存储–计算的性能差异越来越大

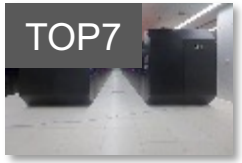
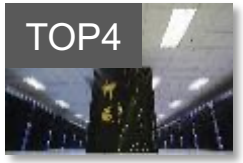
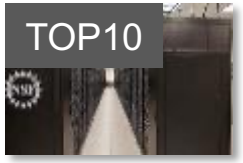


存储落后于计算，差距每年50%递增

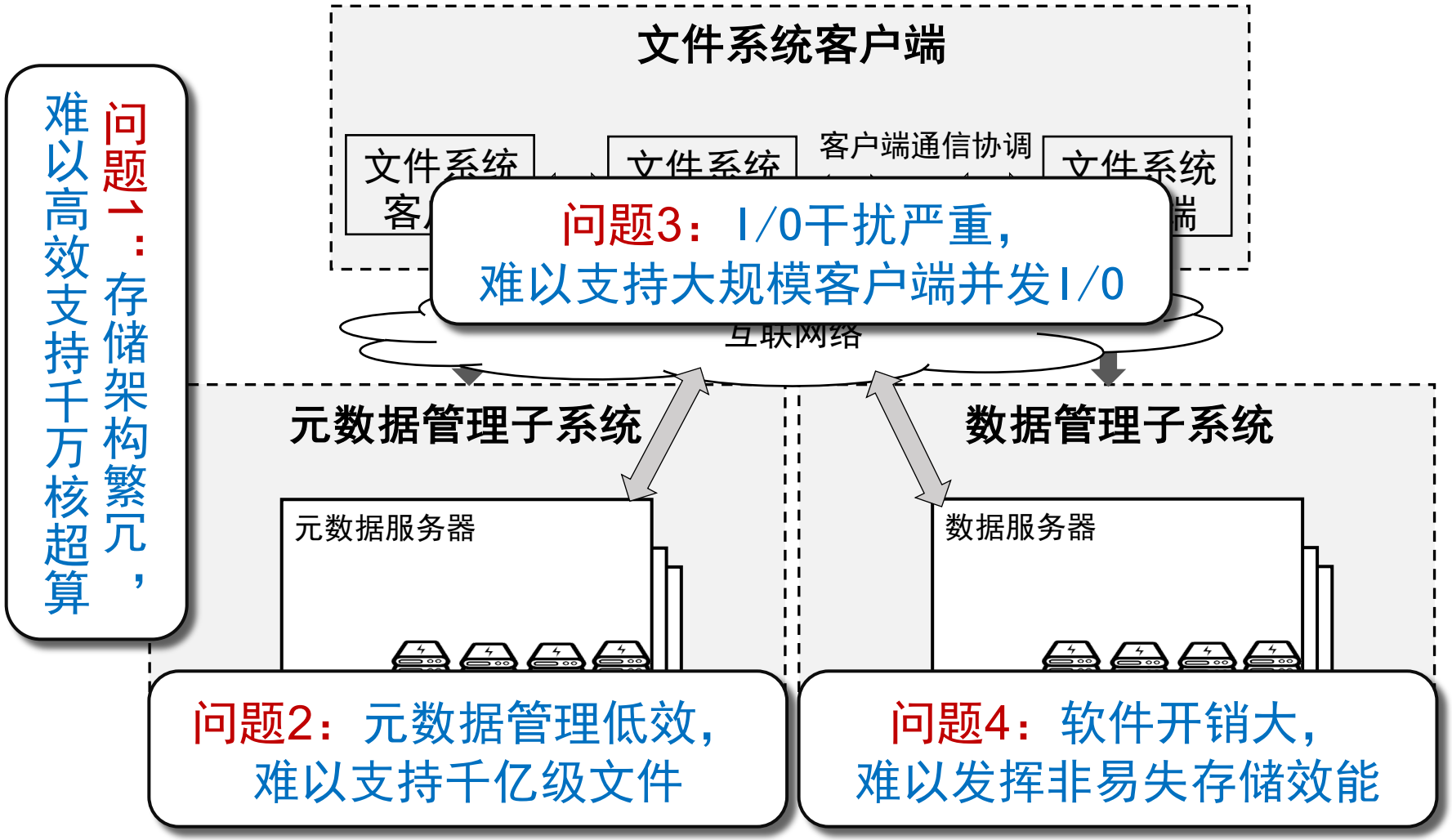
[1] Li, Mu, et al. "Scaling distributed machine learning with the parameter server." OSDI. 2014.
[2] Ken Brock. "Building Efficient Deep Learning Accelerators from the Foundation Up"

国际现状

- 国际超级计算机TOP500排行榜前10名机器均采用Lustre或IBM Spectrum Scale分布式文件系统

TOP1 	Fugaku	Lustre	TOP6 	Selene	Lustre
TOP2 	Summit	IBM Spectrum Scale	TOP7 	天河2号	Lustre
TOP3 	Sierra	IBM Spectrum Scale	TOP8 	JEWELS Booster Module	IBM Spectrum Scale
TOP4 	神威·太湖之光	Lustre	TOP9 	HPC5	Lustre
TOP5 	Perlmutter	Lustre	TOP10 	Frontera	Lustre

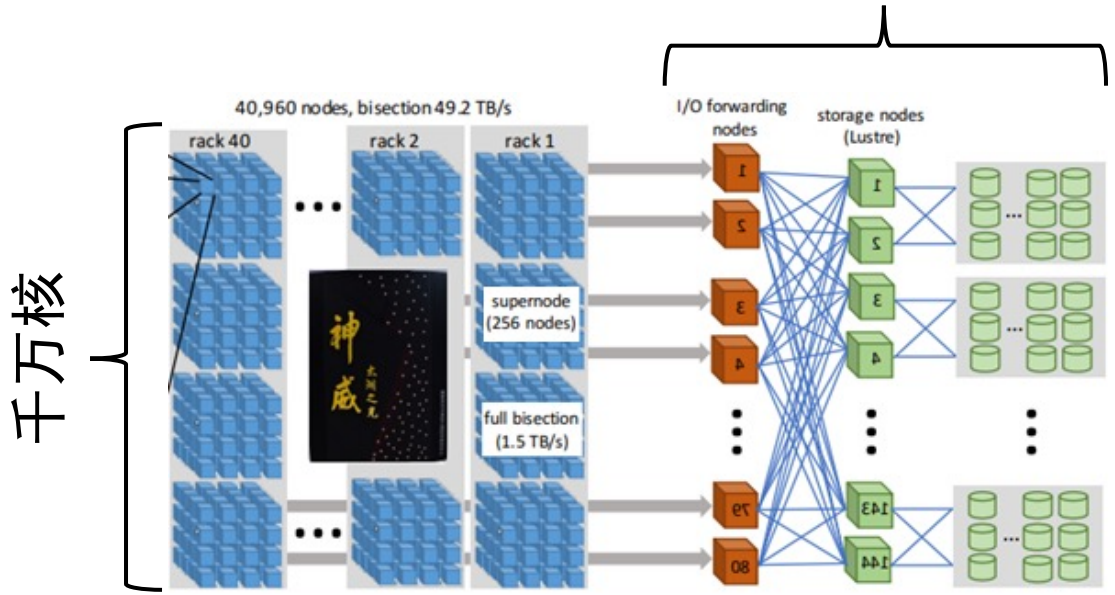
研究现状—现有存储架构及主要问题

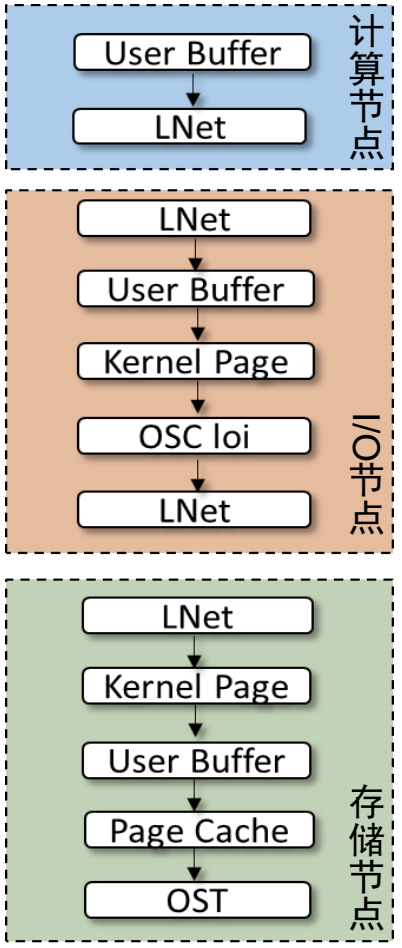


研究现状—现有存储架构的问题1

□ 存储架构繁冗，难以高效支持千万核超算

- 神图全机应用1048.6万个计算核心
- 对应224个存储和I/O服务器，带宽288GB/s
- 最高带宽仅70GB/s，I/O时间超过半小时





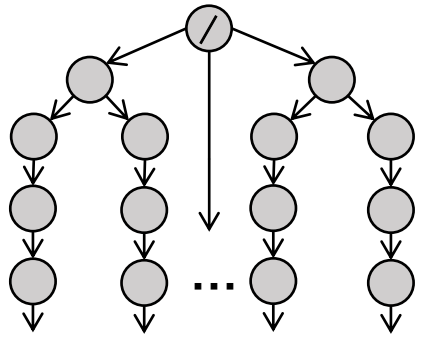
完整I/O路径

I/O路径过长，严重浪费硬件性能

研究现状—现有存储架构的问题2

- 元数据管理低效，难以支持千亿级文件

单机元数据服务

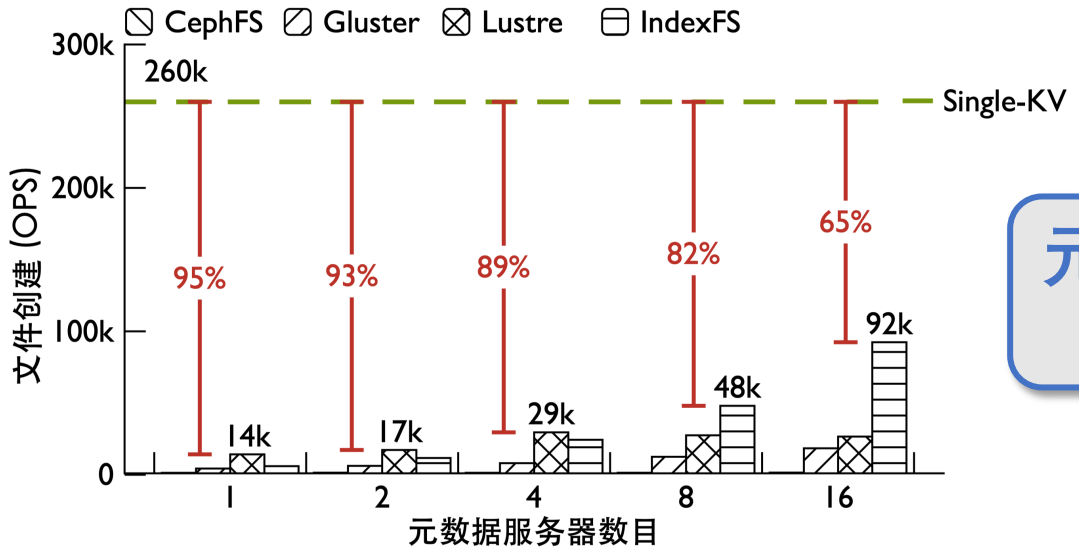


单MDS文件系统	最大文件数
HDFS	1 亿
Lustre	40 亿

远小于1000亿

文件数目受限

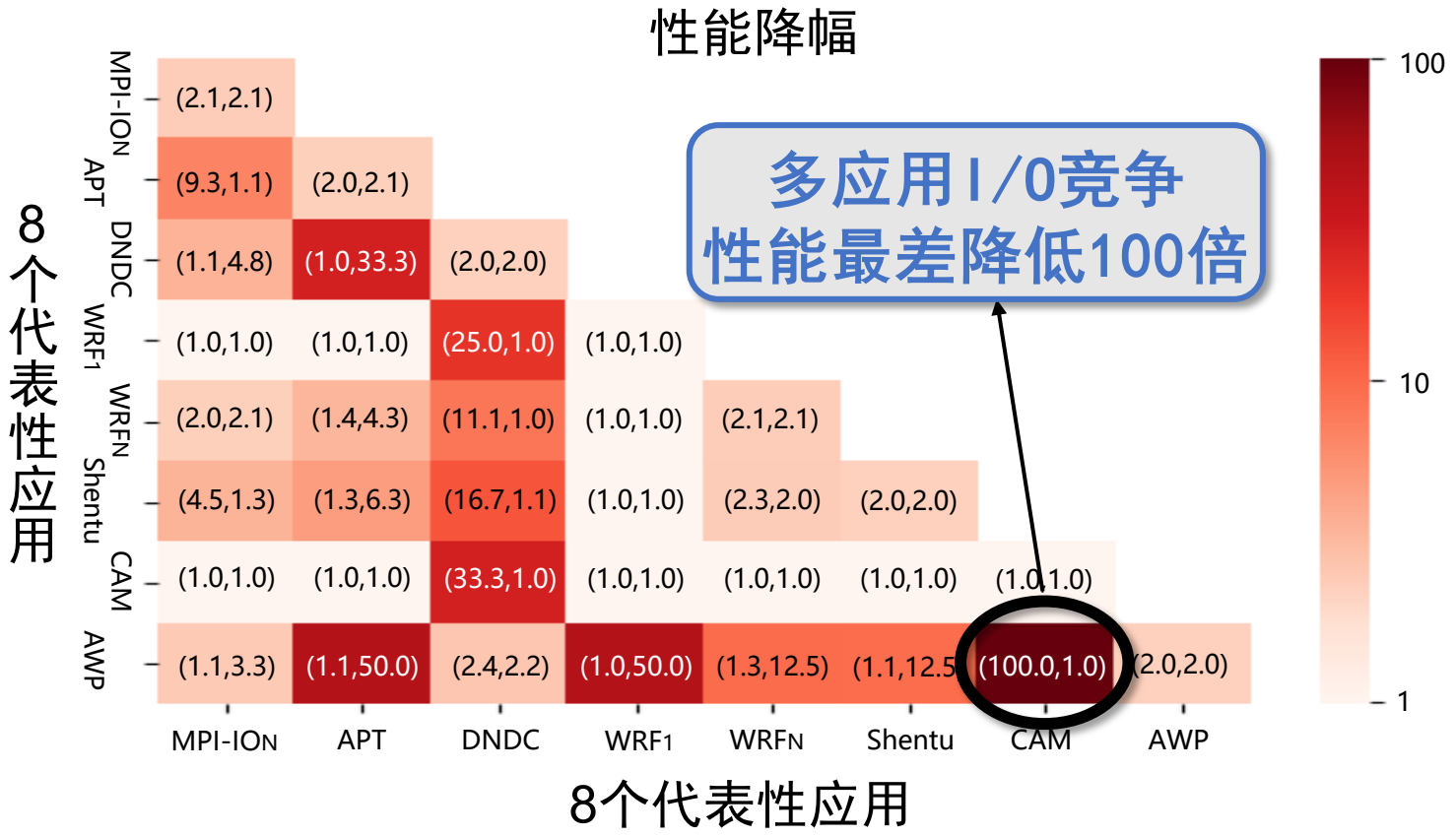
多机元数据服务



元数据依赖复杂
性能难扩展

研究现状—现有存储架构的问题3

▫ I/O干扰严重，难以支持大规模高并发访问

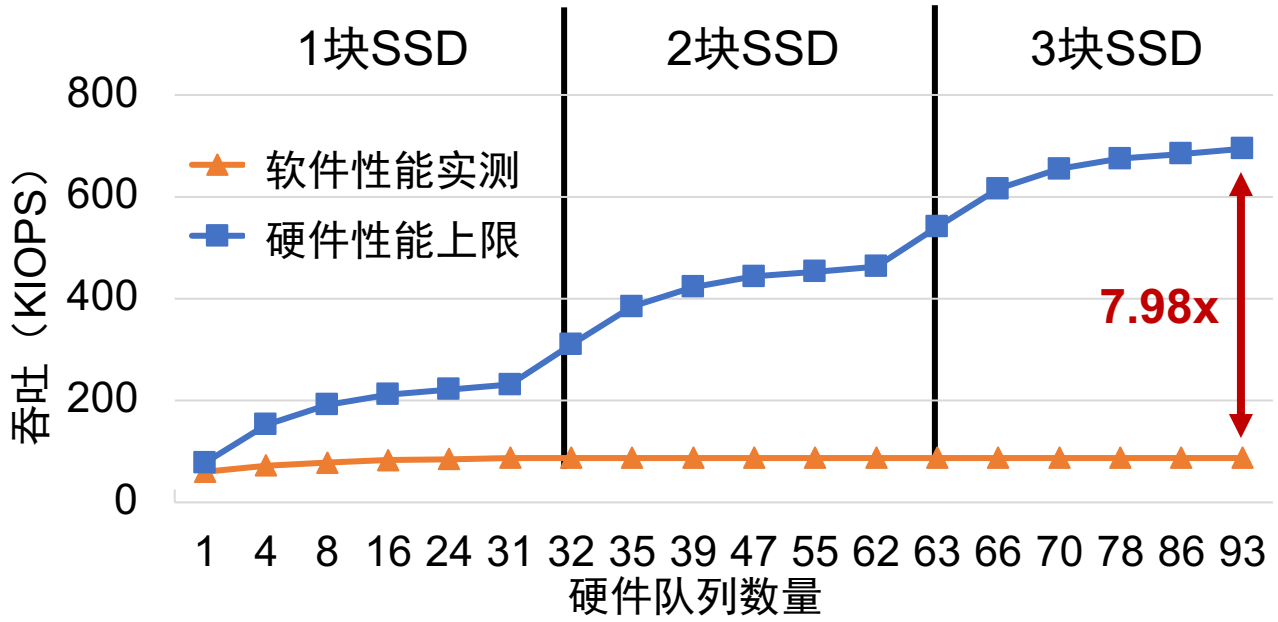


研究现状—现有存储架构的问题4

□ 软件开销大，难以发挥非易失存储硬件效能

	传统磁盘	闪存SSD	非易失内存
延迟	~1ms	~10us	~100ns
带宽	80MB/s	~3GB/s	>40GB/s

硬件性能显著提升



软件未能充分发挥

提纲

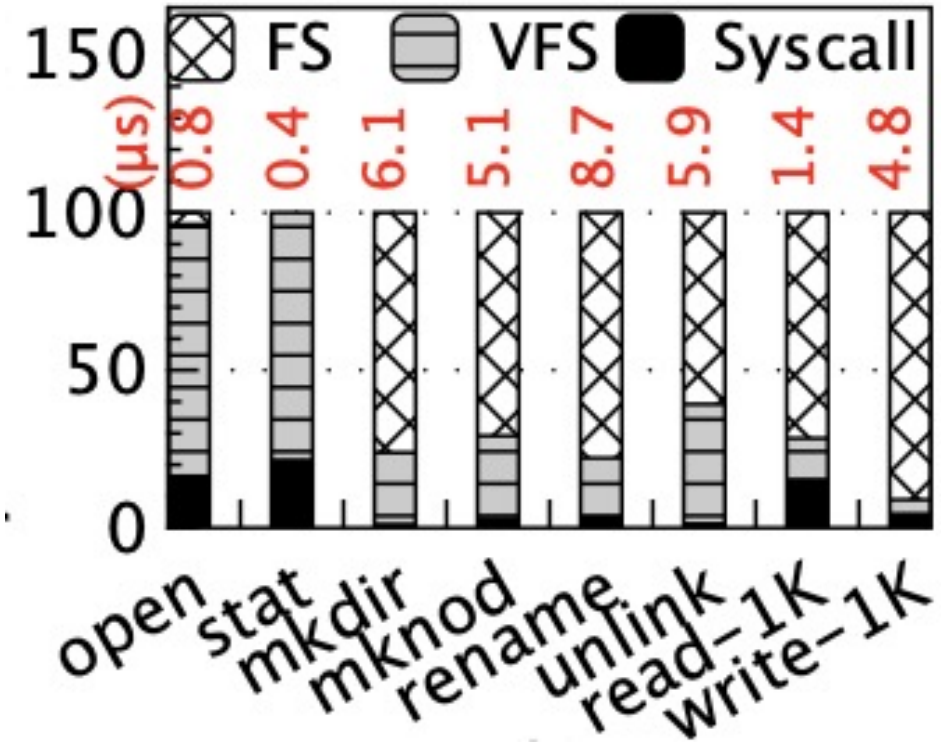
一 背景

二 文件系统数据部分

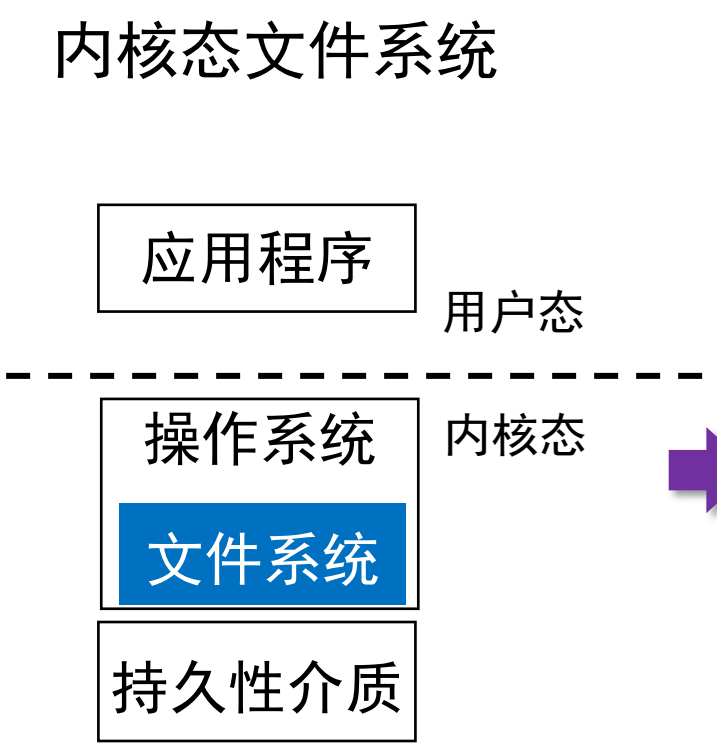
三 文件系统元数据部分

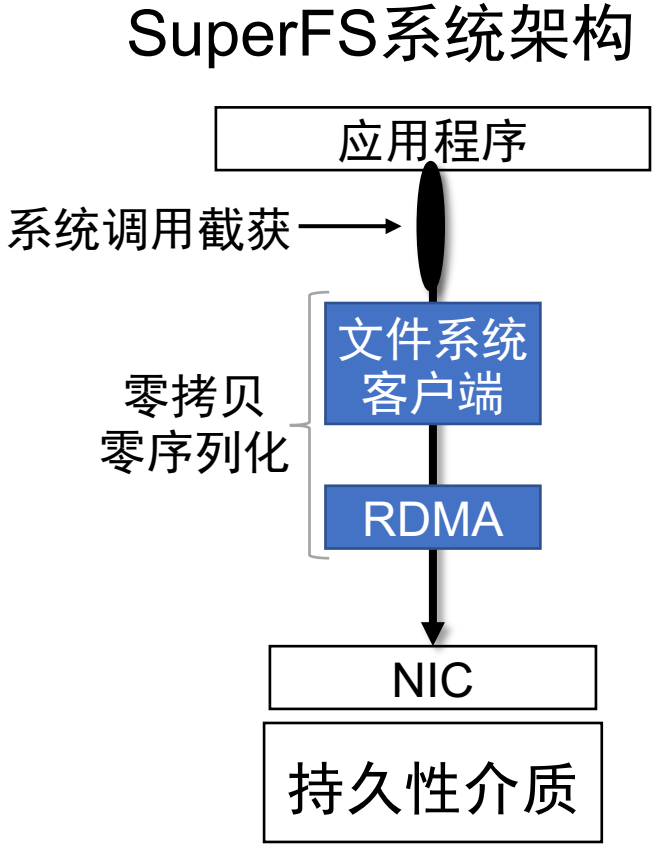
四 总结

数据（1）：用户态直访 – 减少操作系统的额外开销



操作	FS 用时	VFS 用时	Syscall 用时
stat	0.01µs/0%	0.32µs/80%	0.08µs/20%
read	1.01µs/73%	0.18µs/13%	0.21µs/14%

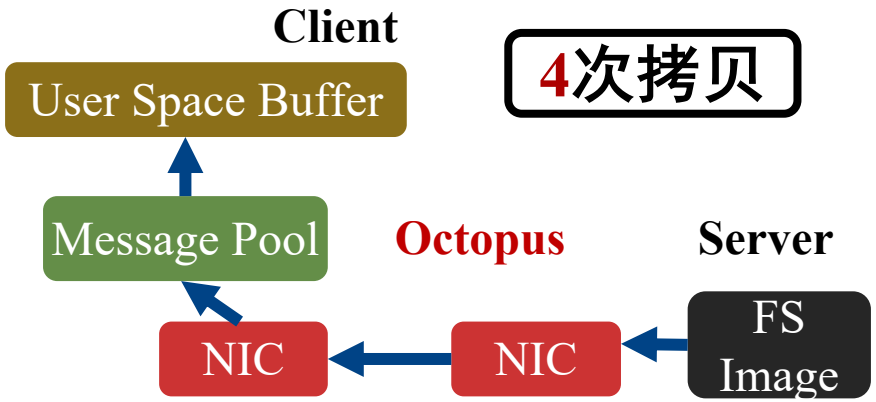


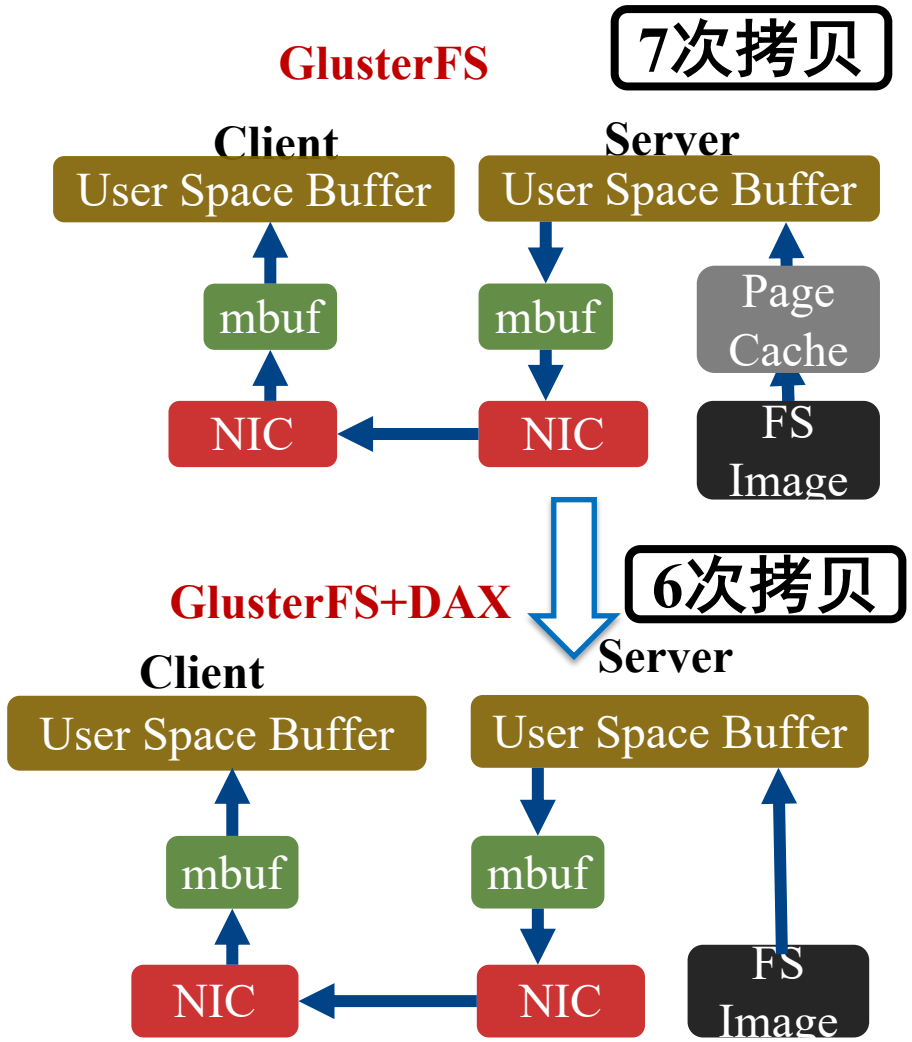


[1] Youmin Chen, Youyou Lu, etc. “Scalable Persistent Memory File System with Kernel-Userspace Collaboration”, **FAST 2021**

数据（2）：数据零拷贝 – 减少拷贝与序列化的开销

- 降低数据路径拷贝与序列化开销
  - 分布式文件系统中冗余拷贝影响性能
  - 参数零序列化，减少序列化次数
  - 引用传递，减少应用层拷贝次数

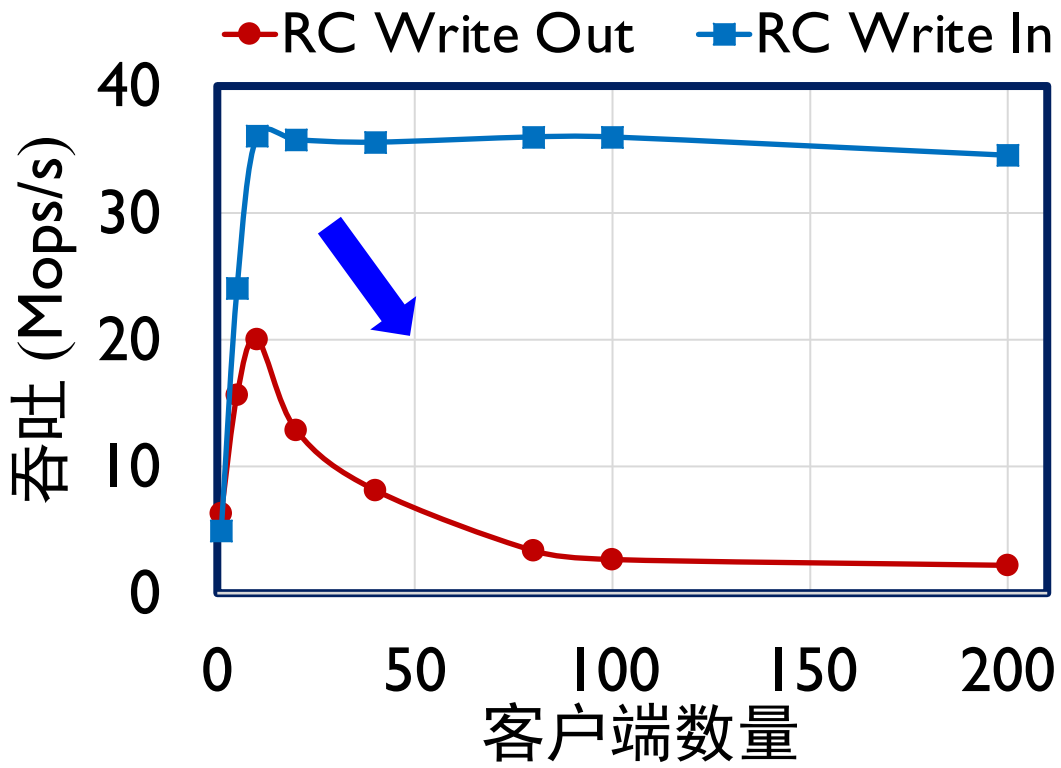


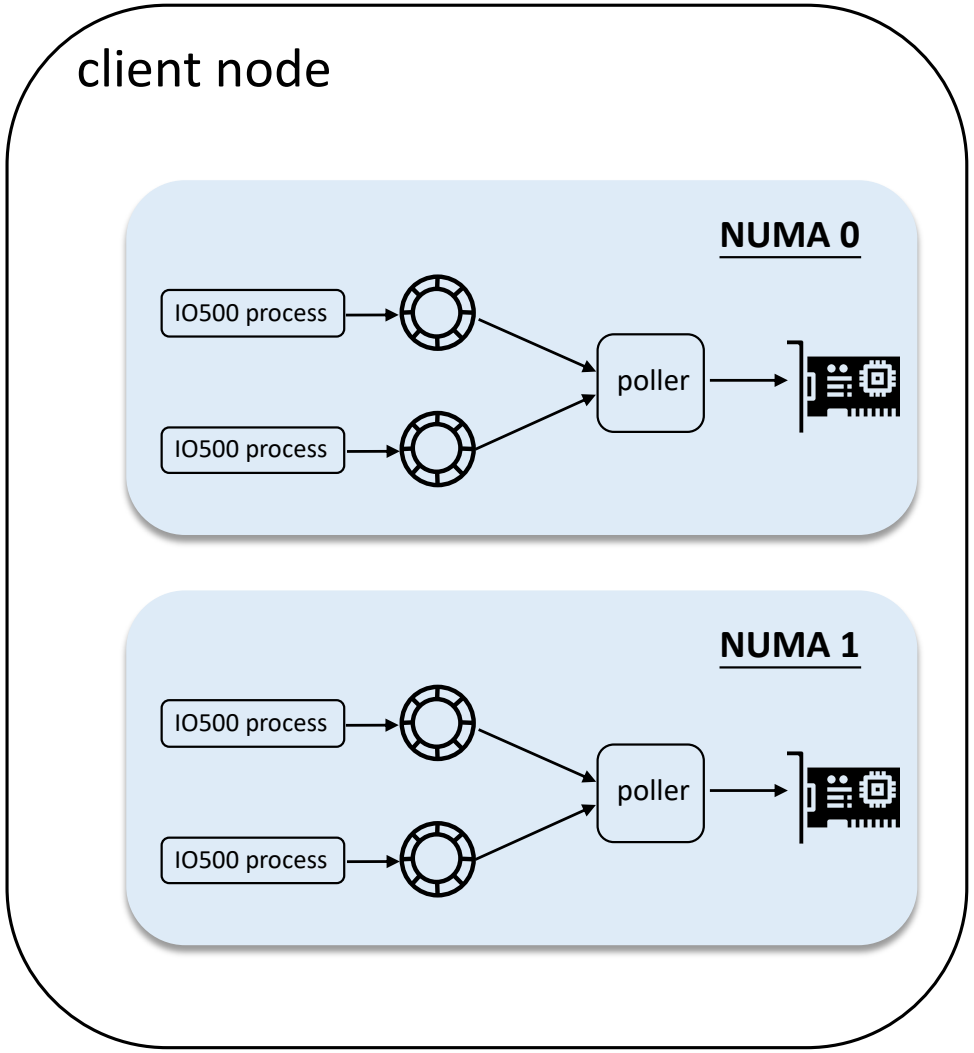


[1] Youyou Lu, Jiwu Shu, etc., *Octopus: an RDMA-enabled Distributed Persistent Memory File System*. In **USENIX ATC 2017**

数据（3）：可扩展的数据通路 – 大规模下性能上得去

- 连接代理机制
- NUMA感知的进程分配与代理转发
- 减少物理连接数量





[1] Youmin Chen, Youyou Lu, etc. “Scalable RDMA RPC on Reliable Connection with Efficient Resource Sharing”, **EuroSys 2019**

提纲

一 背景

二 文件系统数据部分

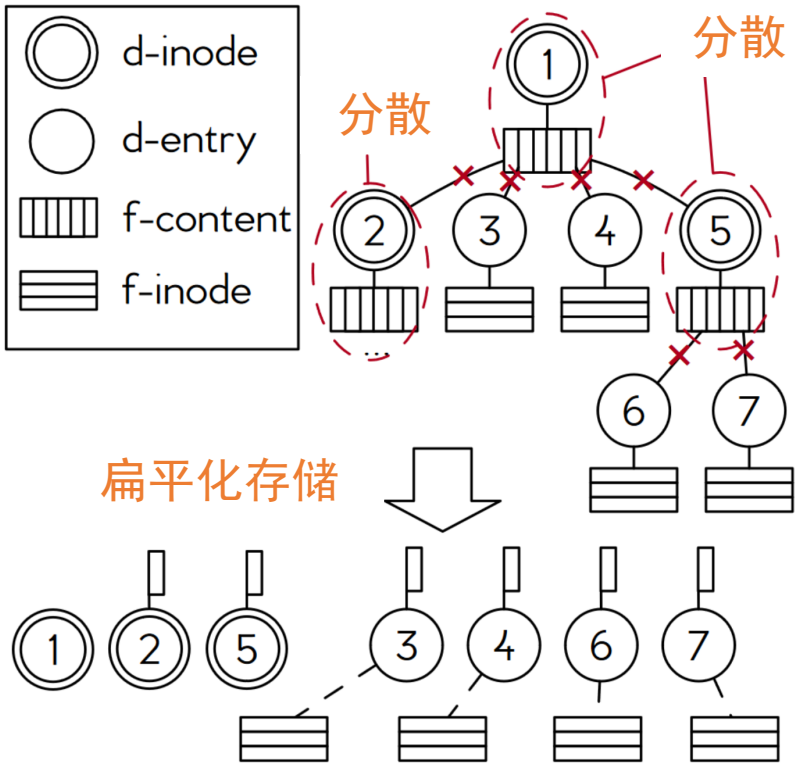
三 文件系统元数据部分

四 总结

元数据（1）：扁平化目录树存储 – 减少逻辑依赖关系

- 核心思想
  - 削弱目录树依赖关系
  - 扁平化组织结构
- 目录项组织
  - 消除目录项数据块
  - 目录项与其关联inode共同放置

削弱元数据依赖关系
适应键值存储结构

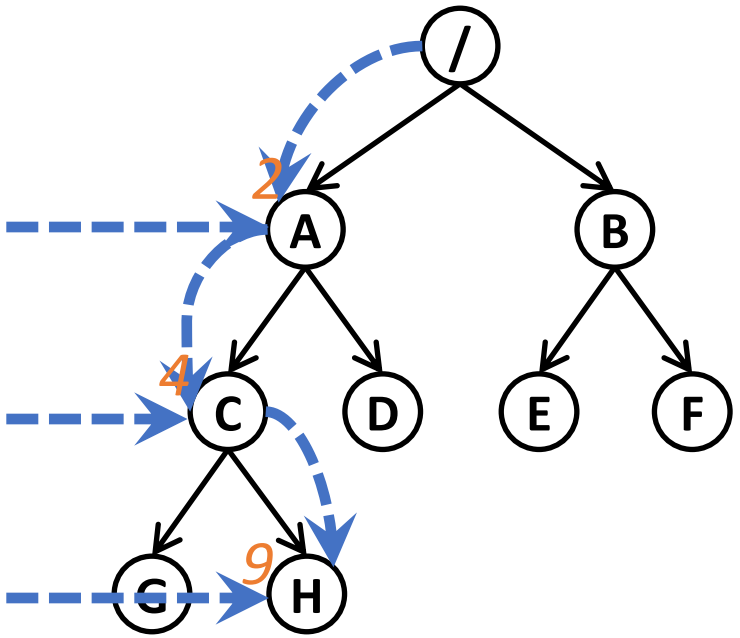


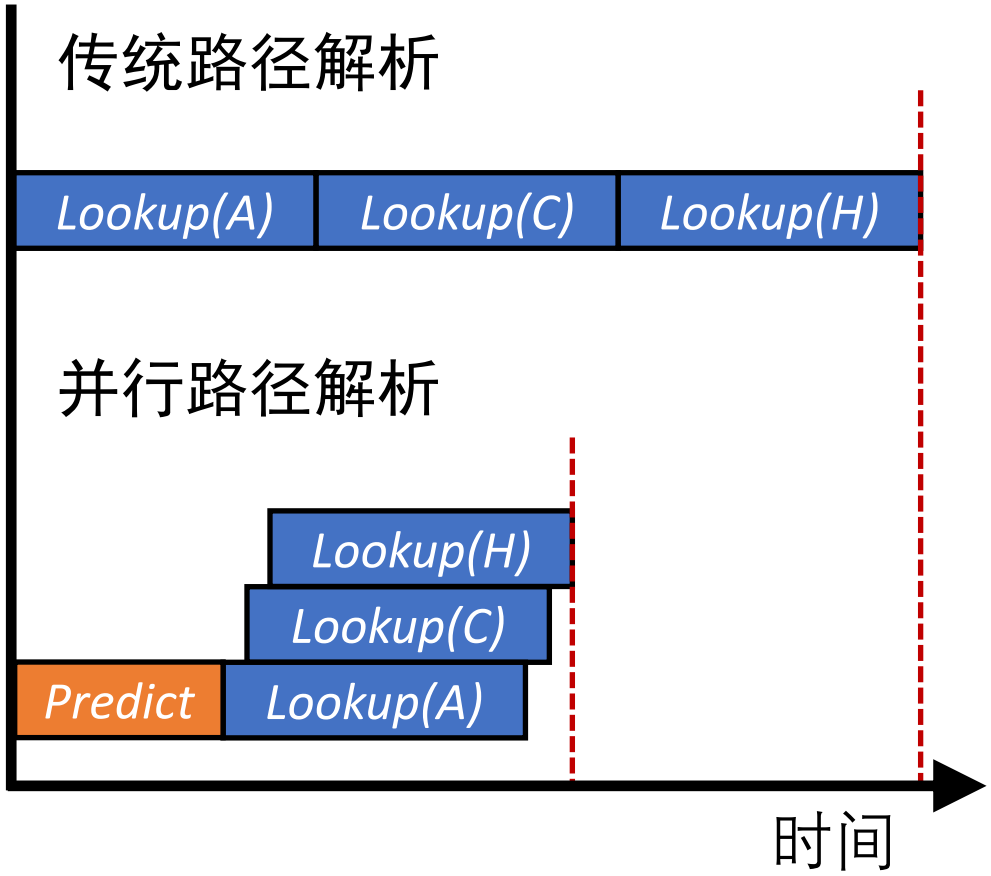
扁平化目录树存储机制

[1] Siyang Li, Youyou Lu, etc. “Locofs: A loosely-coupled metadata service for distributed file systems”, **SC 2017**

元数据（2）：并行路径解析 – 加速元数据访问性能

- 预测目录元数据ID实现并行路径解析





[1] Wenhao Lv, Youyou Lu, etc. “InfiniFS: An Efficient Metadata Service for Large-Scale Distributed Filesystems”, **FAST 2022**

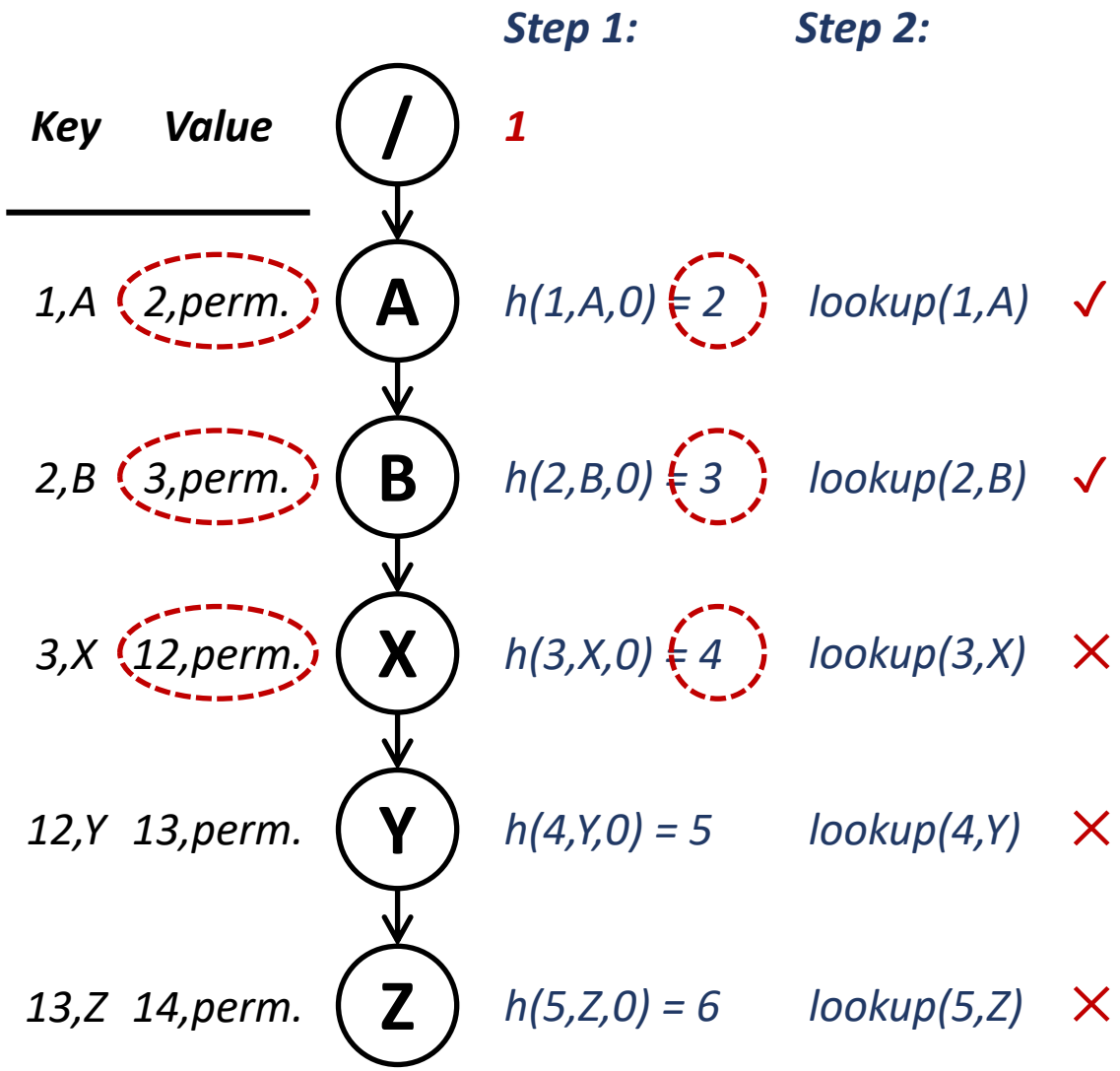
并行路径解析

可预测目录ID

- 利用哈希生成目录ID
- 利用版本号处理冲突

并行路径解析过程

- 预测目录ID
- 并行路径解析
  - 进行权限检查
  - 验证预测ID正确性



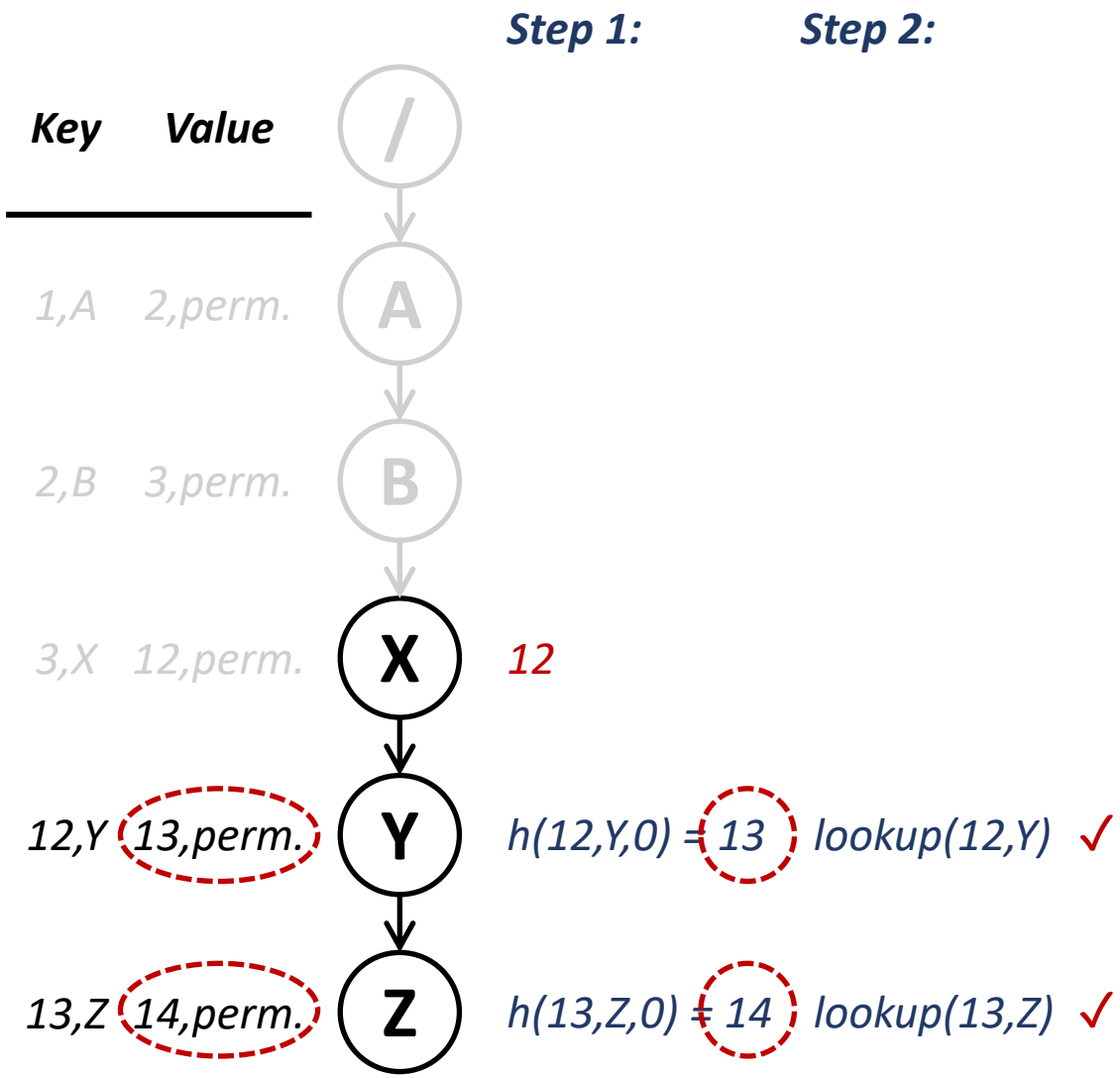
并行路径解析

可预测目录ID

- 利用哈希生成目录ID
- 利用版本号处理冲突

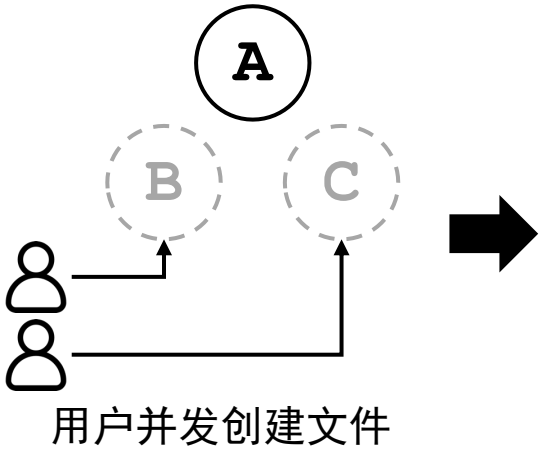
并行路径解析过程

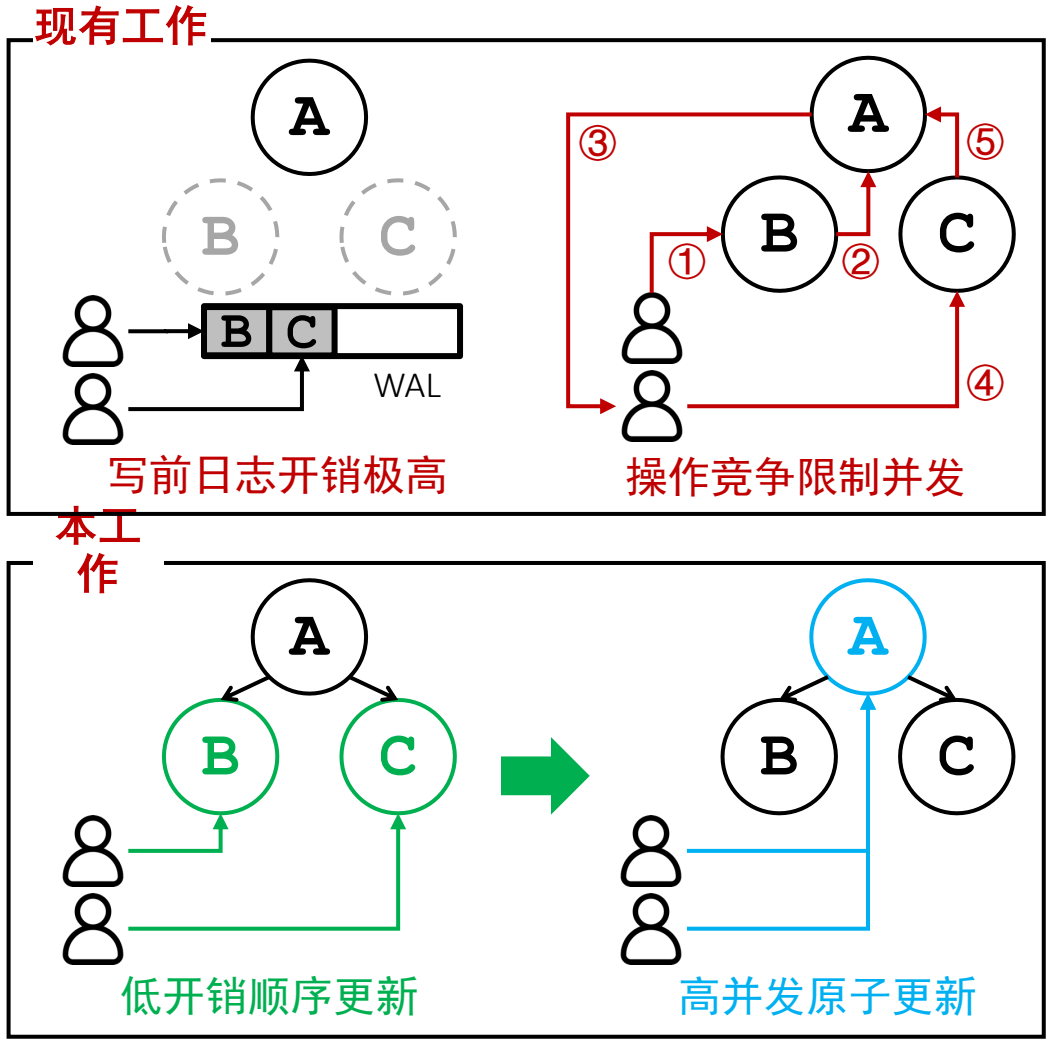
- 预测目录ID
- 并行路径解析
  - 进行权限检查
  - 验证预测ID正确性
- 循环直至完成



元数据（3）：单机性能优化 – 挖掘单节点效率

- 挖掘单机性能
- NUMA管理
- 顺序更新
- 并发控制





[1] Hao Guo, Youyou Lu, etc. “SingularFS: A Billion-Scale Distributed File System Using a Single Metadata Server”, **USENIX ATC 23**

提纲

一 背景

二 文件系统数据部分

三 文件系统元数据部分

四 总结

高性能文件系统SuperFS

- IO500是超算领域存储系统的评测
- 2022年11月 SC超算大会上获得10节点元数据第一名

2022年11月IO500

- 我们用5台服务器

		INFORMATION									IO500		
#	CUSTOM EQUATION	SYSTEM	INSTITUTION	STORAGE VENDOR	FILESYSTEM TYPE	CLIENT NODES	CLIENT TOTAL PROCS	DS NODES	DS STORAGE DEVICES	DS VOLATILE MEMORY CAPACITY	SCORE	BW	MD
1	169,515.95	SuperStore	Tsinghua Storage Research Group	Tsinghua Storage Research Group	SuperFS	10	1200	5	8	153.6 TiB	5,517.73	179.60	169,515.95
2	106,042.93	ParaStor	Sugon Cloud Storage Laboratory	Sugon	ParaStor	10	2560	80	12	3 PiB	8,726.42	718.11	106,042.93
3	88,491.65	StarStor	SuPro Storteck	SuPro Storteck	StarStor	10	2560	80	10	3 PiB	6,751.75	515.15	88,491.65
4	60,119.50	Shanhe	National Supercomputing Center in Jinan	PDSL	flashfs	10	2560				3,534.42	207.79	60,119.50
5	34,777.27	Pengcheng Cloudbrain-II on Atlas 900	Pengcheng Laboratory	Pengcheng	MadFS	10	1800	50			2,595.89	193.77	34,777.27
6	18,235.71	Athena	Huawei HPDA Lab	Huawei	OceanFS	10	1720	80	800		2,395.03	314.56	18,235.71
7	17,224.05	HPC-OCI	Cloudam HPC on OCI	Cloudam	BurstFS	10	720	30	1	114 TiB	1,285.21	95.90	17,224.05
8	16,664.88	OceanStor Pacific	Olympus Lab	Huawei	OceanFS	10	1720	40	400		2,298.69	317.07	16,664.88
9	9,827.09	Kongming	BPFS Lab		BPFS	10	800	35	280		972.60	96.26	9,827.09
10	8,671.65	Endeavour	Intel	Intel	DAOS	10	1440	40	8		1,859.56	398.77	8,671.65

2023年5月IO500

SuperFS部署于智算中心(鹏城云脑II)
获得了IO500全球第一

# ↑	INFORMATION					CLIENT NODES	TOTAL CLIENT PROC.	SCORE ↑	BW (GIB/S)	MD (KIOP/S)
	BOF	INSTITUTION	SYSTEM	STORAGE VENDOR	FILE SYSTEM TYPE					
1	ISC23	Pengcheng Laboratory	Pengcheng Cloudbrain-II on Atlas 900	Pengcheng Laboratory and Tsinghua University	SuperFS	300	36,000	210,254.98	4,847.48	9,119,612.35
2	ISC23	JNIST and HUST PDSL	Cheeloo-1 with OceanStor Pacific	Huawei	OceanFS2	10	9,600	137,100.02	2,439.37	7,705,448.04
3	SC22	Argonne National Laboratory	Aurora Storage	Intel	DAOS	260	27,040	20,694.50	6,048.69	70,802.51
4	SC22	Sugon Cloud Storage Laboratory	ParaStor	Sugon	ParaStor	10	2,560	8,726.42	718.11	106,042.93
5	SC22	SuPro Storteck	StarStor	SuPro Storteck	StarStor	10	2,560	6,751.75	515.15	88,491.65

总结

- 数据 驱动了 信息技术的发展。高性能计算、大数据处理、人工智能均离不开数据存储与处理。

- 在大规模超算/智算中心里，数据存取性能影响很大。其中，数据存取所面临的挑战主要有两个：
  - 大规模场景下的可扩展性
  - 高性能硬件上的软件效率

- SuperFS是高性能硬件上新一代存储系统的探索。我们可以而且有必要去做。

谢谢

陆游游

清华大学计算机系

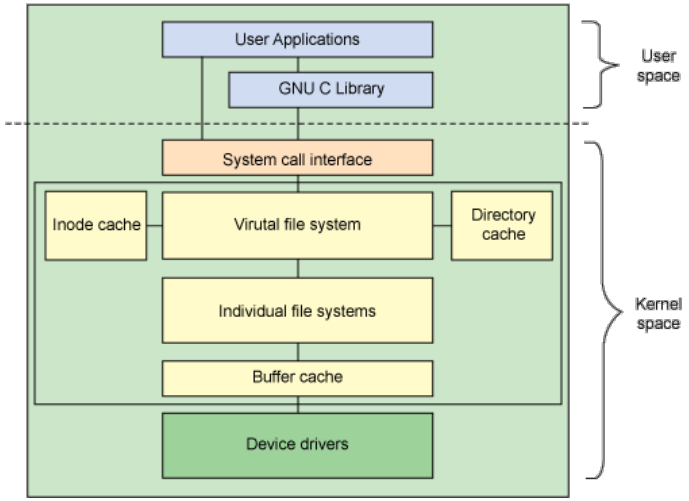
luyouyou@tsinghua.edu.cn

http://storage.cs.tsinghua.edu.cn/~lu

软件面临更大的挑战：软件效率问题

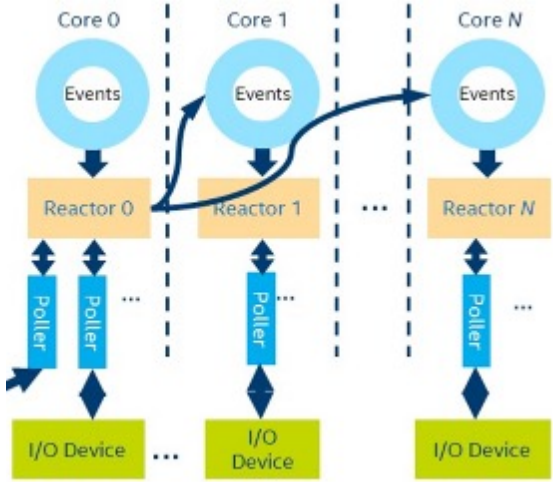
硬件慢，软件复杂

ms scale



硬件快，软件高效

10-100us scale



- 软件直管闪存[1]
  - 以高效架构与软件挖掘闪存的潜力
- 网络互连内存[2]
  - 以高效架构与软件挖掘NVM的潜力

[1] Youyou Lu, Jiwu Shu, Weimin Zheng, *Extending the lifetime of flash-based storage through reducing write amplification from file systems*. In **FAST 2013**.

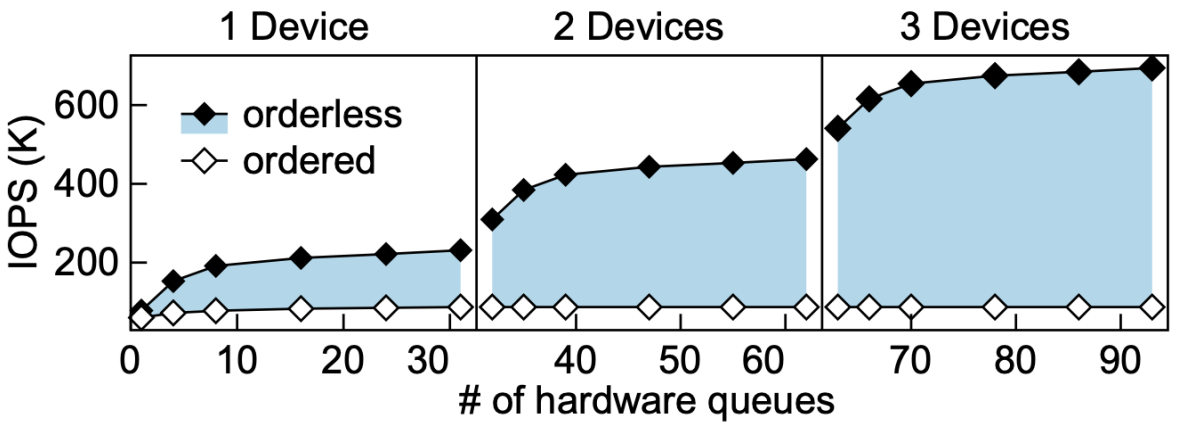
[2] Youyou Lu, Jiwu Shu, etc., *Octopus: an RDMA-enabled Distributed Persistent Memory File System*. In **USENIX ATC 2017**

臃肿的软件栈无法发挥新硬件性能

- 新型硬件延迟低

	SSD	持久性内存	RDMA
延迟	~10μs	~100ns	~1μs
带宽	5GB/s	>40GB/s	200Gbps

- 软件系统难以发挥SSD硬件性能

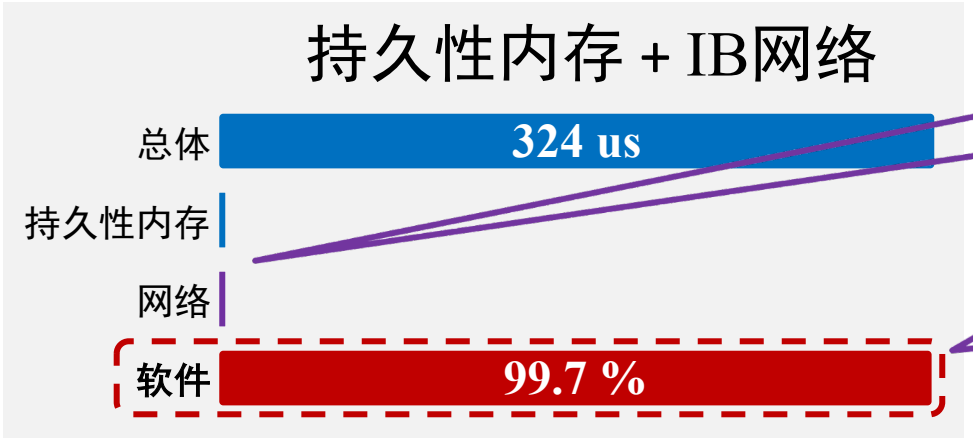


臃肿的软件栈无法发挥新硬件性能

- 新型硬件延迟低

	SSD	持久性内存	RDMA
延迟	~10μs	~100ns	~1μs
带宽	5GB/s	>40GB/s	200Gbps

- 软件系统难以发挥内存性能

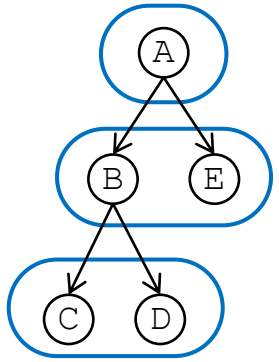


RDMA和持久性内存的硬件延迟极低

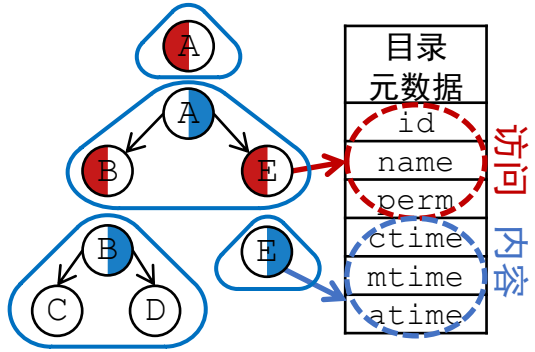
存储软件的开销占比高达99.7%！

NUMA感知元数据组织

- 将目录元数据拆分为两部分
  - 访问元数据：与目录树访问相关
  - 内容元数据：与子节点更新相关
- 将相关的元数据置于相同NUMA节点
  - 访问元数据与父目录，内容元数据与子文件



传统方法：文件创建 / 删除
无法保证NUMA局部性



本方法：文件创建 / 删除
保证NUMA局部性

InfiniFS: An Efficient Metadata Service for Large-Scale Distributed Filesystems
SingularFS: A Billion-Scale Distributed File System Using a Single Metadata Server

高并发共享元数据更新

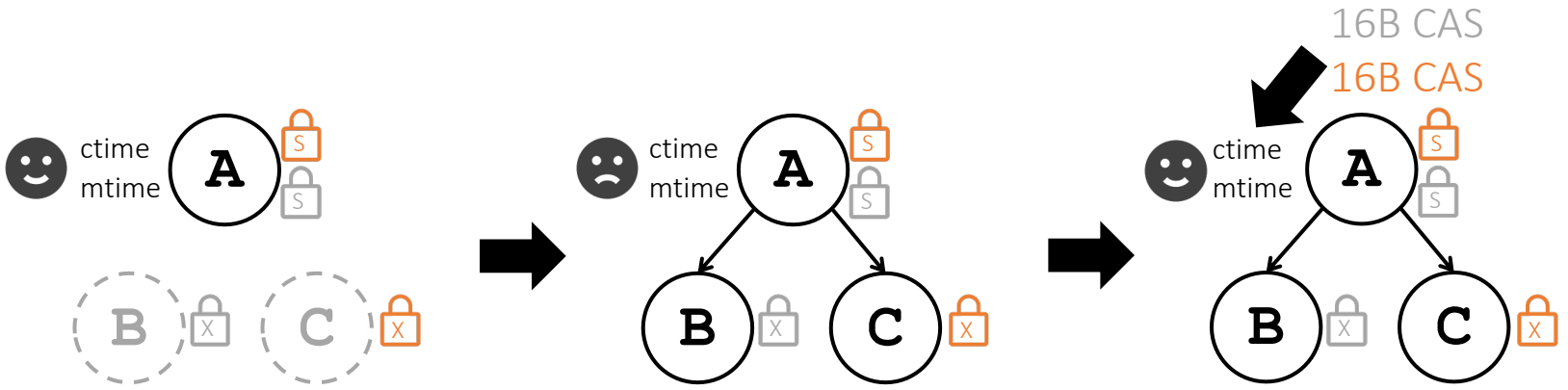
将竞争元数据更新临界区缩短至单次原子操作

共享目录下文件创建 (删除) 操作间并发控制

1. 获取目标inode的写锁和父目录的读锁
2. 并行插入 (删除) 元数据键值对
3. 利用单次原子操作更新父目录元数据

共享文件写操作间并发控制

1. 并行写入数据区段
2. 利用单次原子操作更新文件元数据



操作：线程1创建/A/B，线程2并发创建/A/C

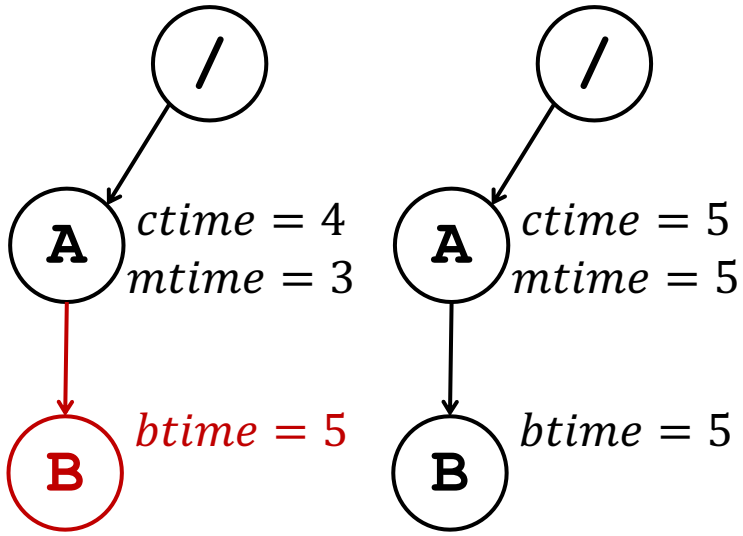
低开销崩溃一致性保证

- 利用元数据依赖关系减少崩溃一致性开销

将大部分元数据操作转换为无事务依赖的KV操作

inode创建

- **写入目标inode元数据同时附带写入其创建时间 (btime)**
  - 崩溃检测：父目录ctime < max(子inode的btime)
  - 崩溃恢复：父目录ctime = mtime = max(子inode的btime)
- **更新父目录的ctime和mtime为目标inode的创建时间 (btime)**

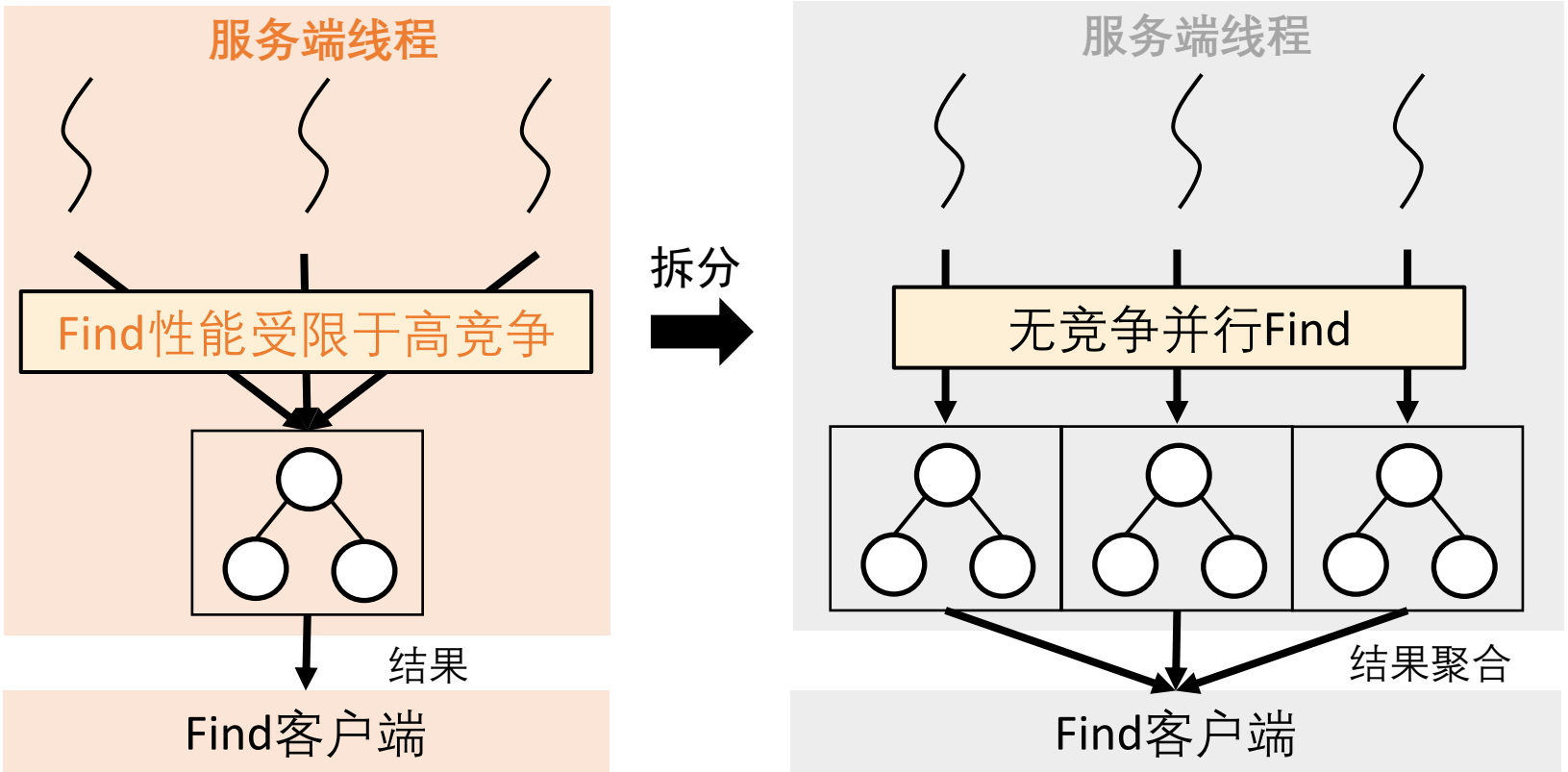


操作: $t_0 = 5$时创建/A/B

并行元数据查找

Find操作性能受限于数据结构竞争

拆分元数据实现服务端并行查找



乐观元数据缓存

元数据缓存一致性维护开销大

元数据服务器端乐观处理缓存失效