

FusionStor

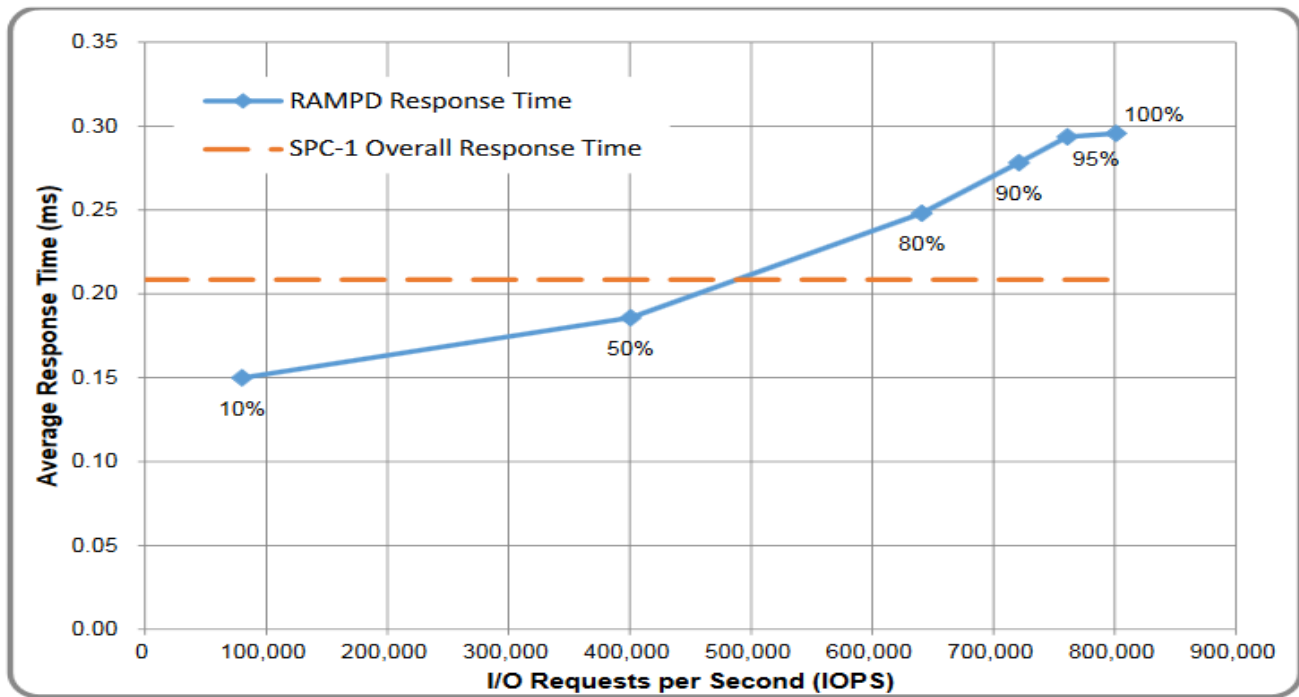
分布式块存储系统性能优化之路

亓武强
分布式存储架构师

关于华云网际

- ◆ 国内首个超融合 (**FusionStack**[®]) 应用案例
- ◆ SPC (存储性能理事会) 4家中国成员之一
- ◆ SPC认证全球性价比最高的分布式存储**FusionStor**[®]

Response Time and Throughput Graph



基于SPDK的fusionstor的spc1性能

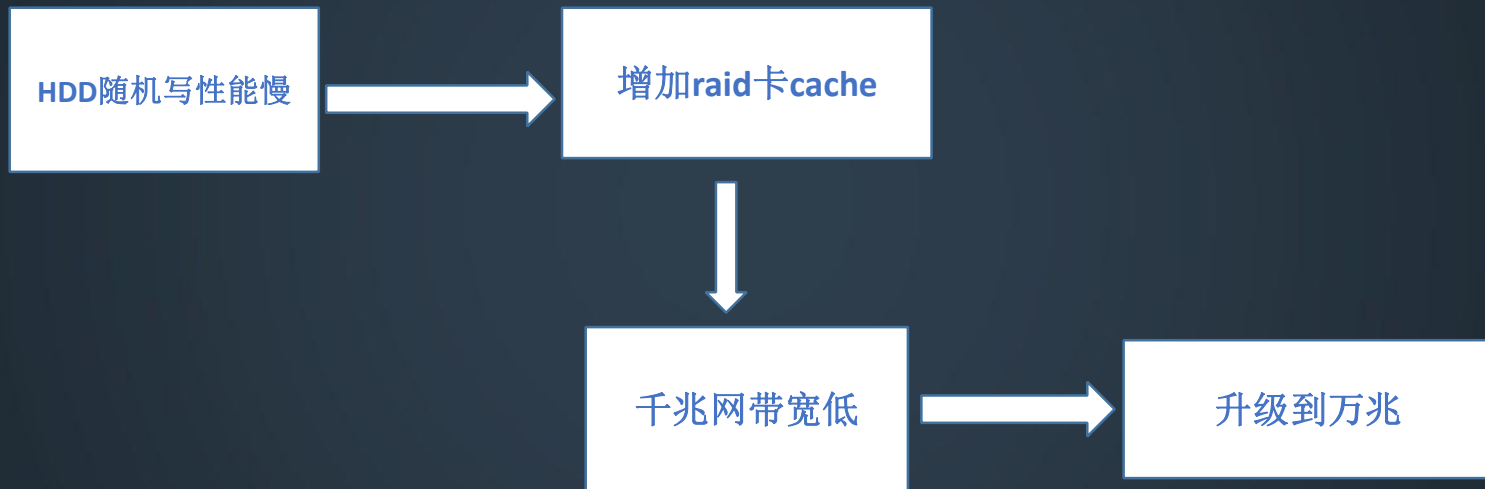
衡量存储产品好坏的标准？

高可用

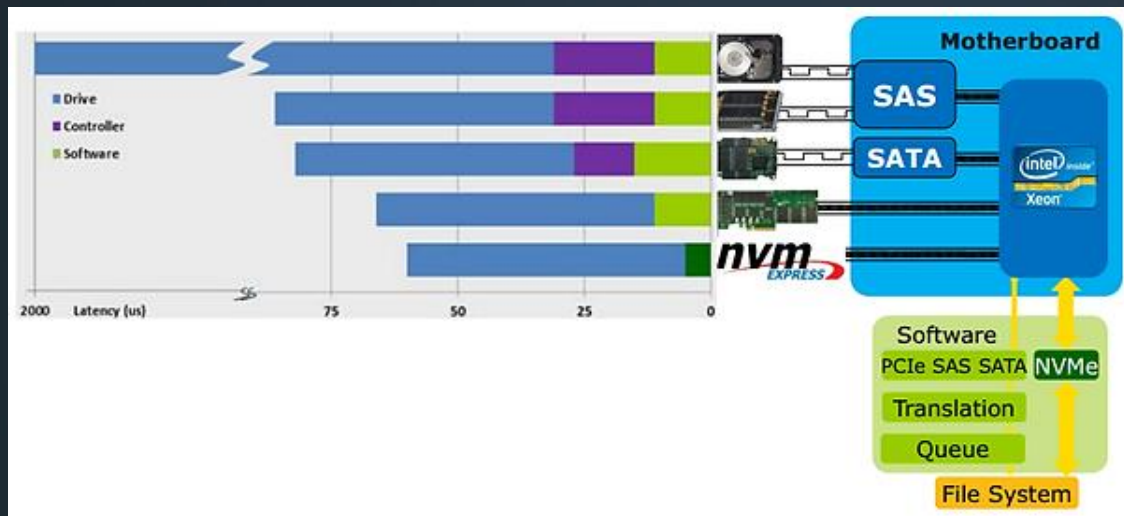
功能

性能

HDD时代fusionstor主要的性能优化方法



NVME SSD比SATA SSD的性能又大幅提升



PCI-E接口消除了硬件上的延时

NVMe协议大幅降低了协议开销

存储软件又该如何应对？？？

```
procs -----memory----- --swap-- -----io----- -system-- -----cpu-----
 r b  swpd  free  buff  cache  si  so   bi   bo  in   cs us sy id wa st
21  1 1243872 8250852 425144 16003796  0  0   11  2704   1   5 34 11 54  1  0
 2  0 1243872 8236568 425144 16021568  0  0    0 63976 47803 91819 22  7 70  1  0
33  0 1243872 8239912 425144 16012796  0  0    0 68364 103067 184394 37 16 46  1
76  2 1243872 8219680 425144 16033296  0  0    0 84552 126484 231530 61 17 31  1
40  0 1243872 8216288 425144 16036904  0  0    0 82968 112019 217908 41 15 44  1
```


与内核相关的两个性能杀手

上下文切换

内存复制

上下文切换的时间消耗

Context switching - times in microseconds - smaller is better

```
-----  
Host          OS  2p/0K 2p/16K 2p/64K 8p/16K 8p/64K 16p/16K 16p/64K  
              ctxsw ctxsw  ctxsw ctxsw  ctxsw  ctxsw  ctxsw  
-----  
stor01  Linux 3.10.0- 2.1500 2.2300 2.0100 2.9100 4.2300 3.44000 3.46000
```

2M task ,10 cpu core, context switching 需要消耗多少时间

影响上下文切换的因素

时间片用完，CPU正常调度下一个任务
被其他优先级更高的任务抢占（无节制使用线程）

执行任务碰到IO阻塞，调度器挂起当前任务，切换执行下一个任务（阻塞性系统调用）

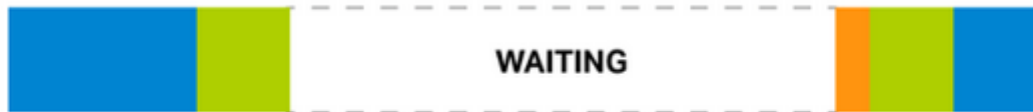
用户代码主动挂起当前任务让出CPU时间

多任务抢占资源，由于没有抢到被挂起（互斥锁）

硬件中断

任务阻塞

**BLOCKING
OPERATION**

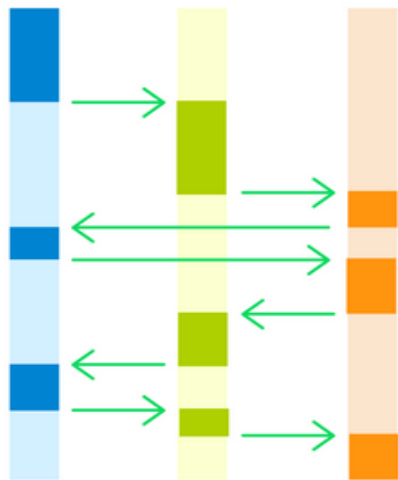


WAITING



TRADITIONAL SERVER

PROCESS 1 PROCESS 2 PROCESS 3



NGINX WORKER

PROCESS



TASK SWITCHES



PROCESSING REQUEST 1



PROCESSING REQUEST 2

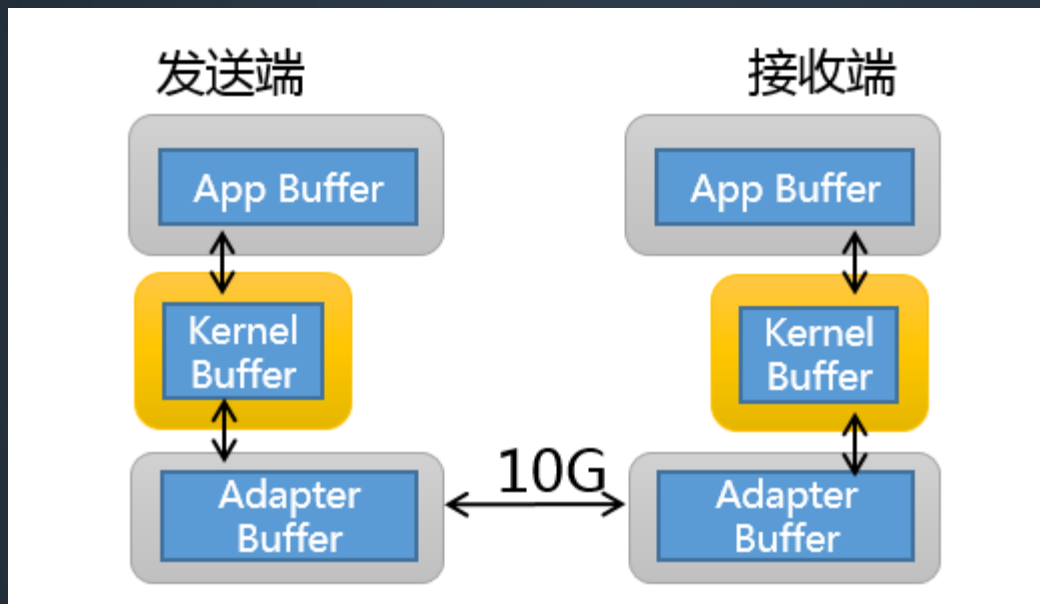


PROCESSING REQUEST 3

内存复制的时间消耗

```
*Local* Communication bandwidths in MB/s - bigger is better
-----
Host          OS   Pipe AF      TCP   File   Mmap   Bcopy   Bcopy   Mem   Mem
              OS   Pipe AF      TCP   reread reread (libc) (hand) read write
-----
stor01      Linux 3.10.0- 3772 7802 7679 5885.7 8796.0 7048.5 4767.1 9380 5865.
```

2M iops ,4k blocksize , 10 cpu core, memcpy需要消耗多少时间?



网络层内存复制

影响性能的其他因素

- 软件算法
- 锁
- Cpu (cache, tlb,分支预测)
- 内存(page fault, numa)
- 编译器

避免使用互斥锁（无锁架构）

避免使用阻塞io系统调用（所有io操作全部设置为非阻塞）

避免使用过多的线程(主io路径的线程数和设定的cpu数量相同)

尽量减少io路径下的内存复制（rdma/dpdk zero copy）

尽量降低硬件中断数量（rdma/dpdk）

The kernel isn' t the solution. The kernel is the problem

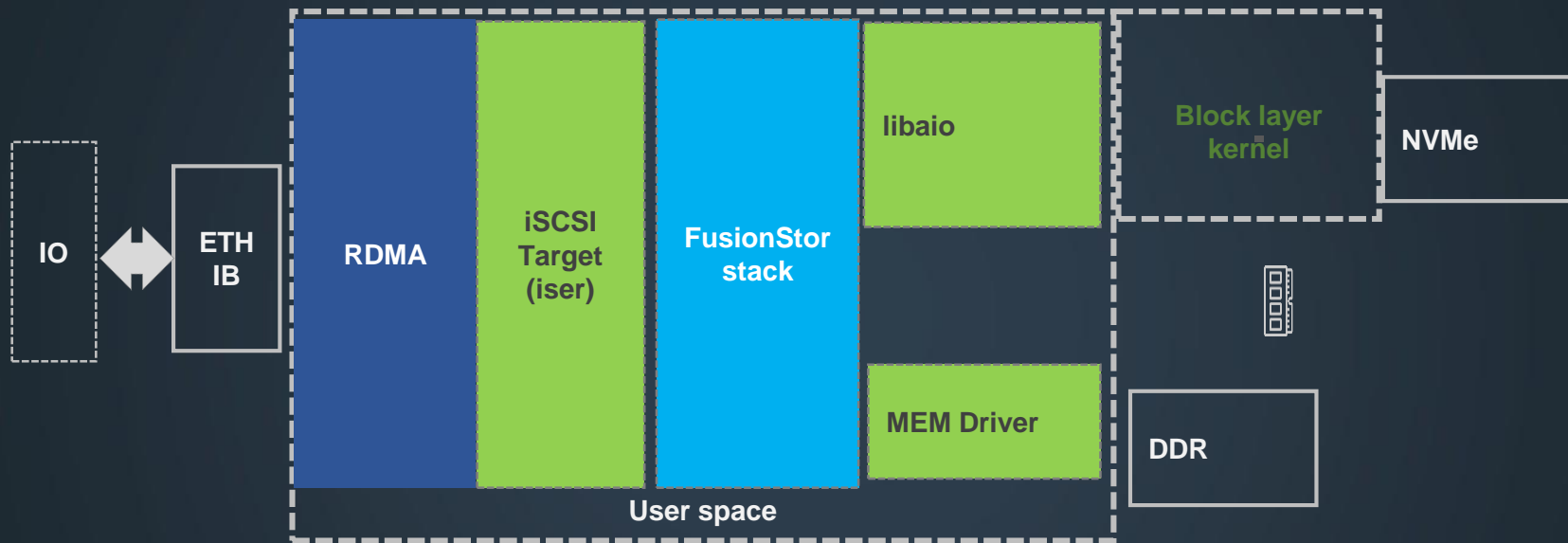
编程模型：同步非阻塞

- 任务调度：coroutine
- 事件处理：polling
- 多核同步：session based hash

数据路径：kernel-bypass

- Network：RDMA / DPDK
- Flash：libaio
- Mem：hugepage

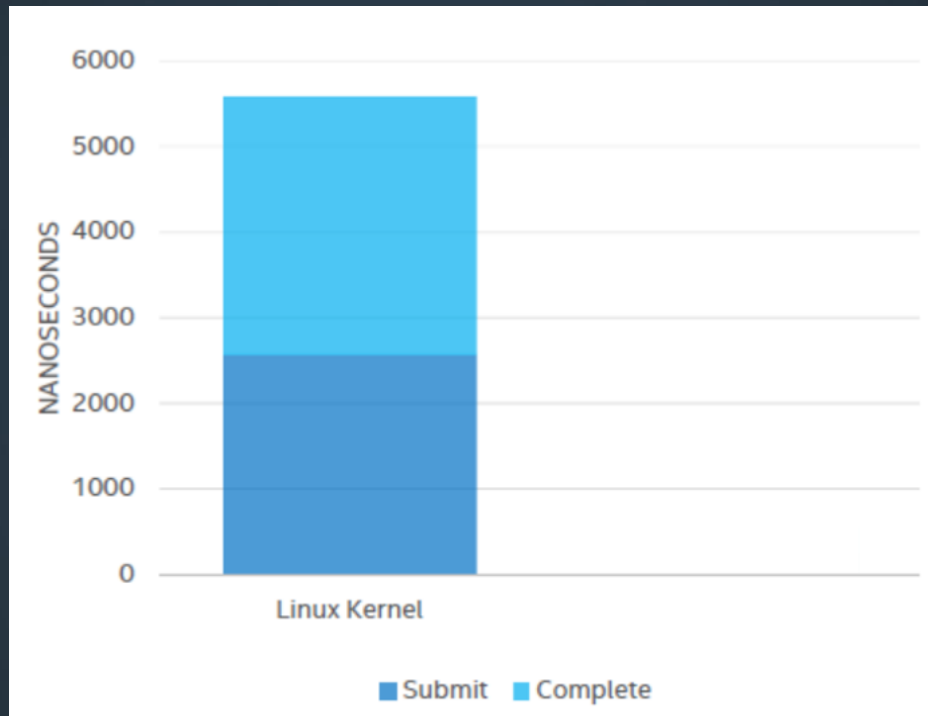
基于rdma的存储架构



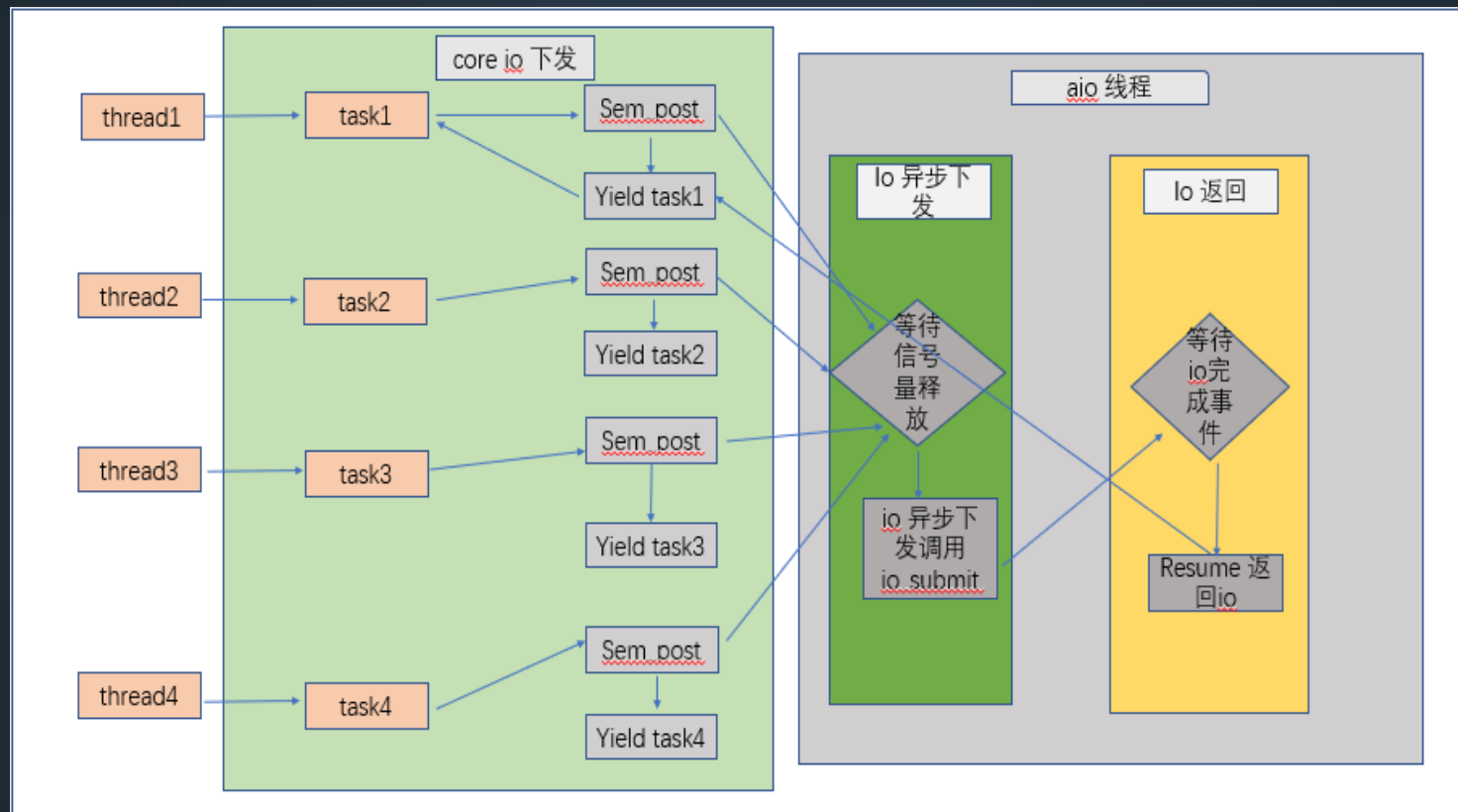
- ✓ 完美支持DPDK网卡、NVMe闪存硬件，基于Intel SDPK软件架构
- ✓ 用户态运行设备驱动，零拷贝机制，大幅降低内外切换性能损耗
- ✓ Polling定点轮循驱动，降低中断开销，高效并发IO处理



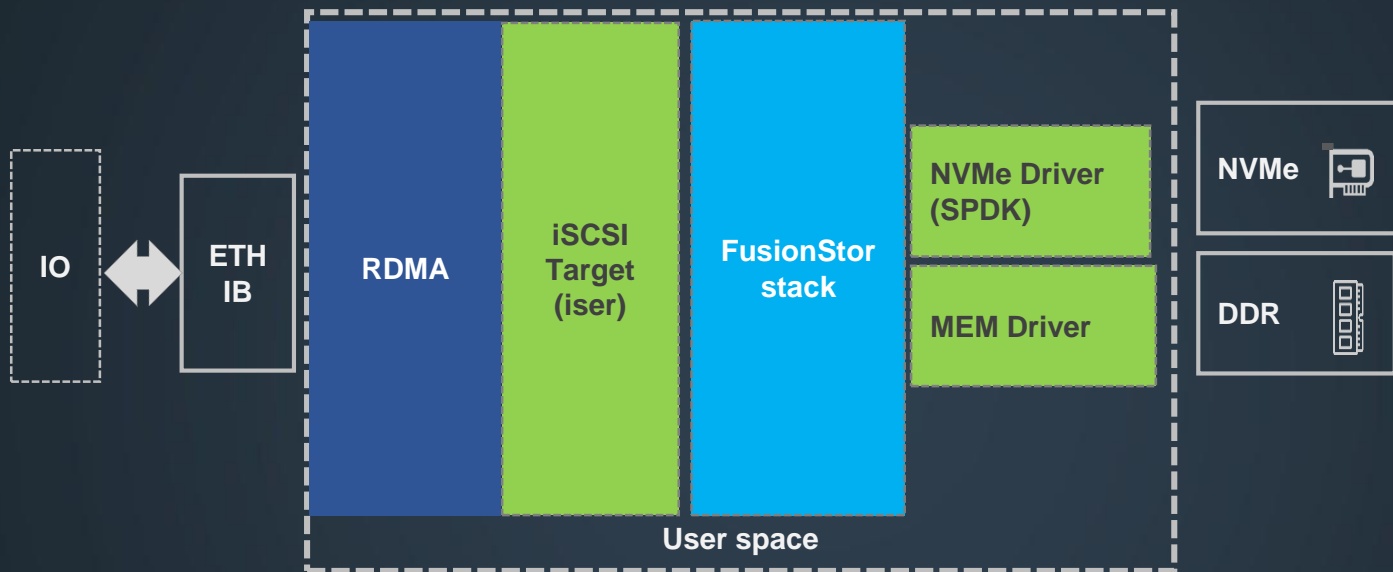
IO 在KERNEL 时间消耗



Fusionstor aio 处理流程

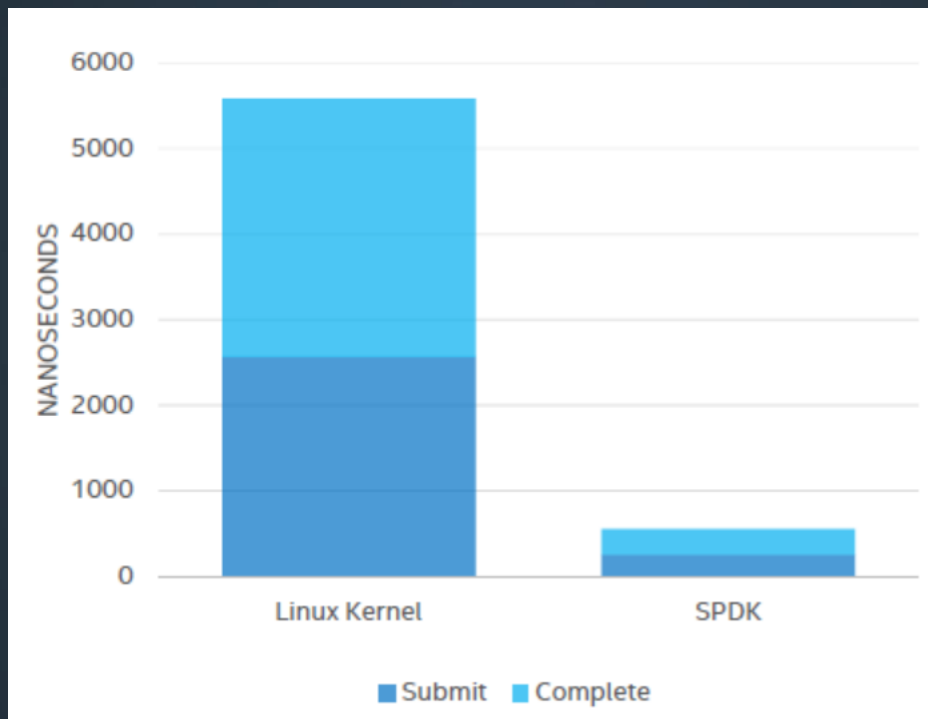


Kernel-bypass架构释放NVMe闪存极致性能

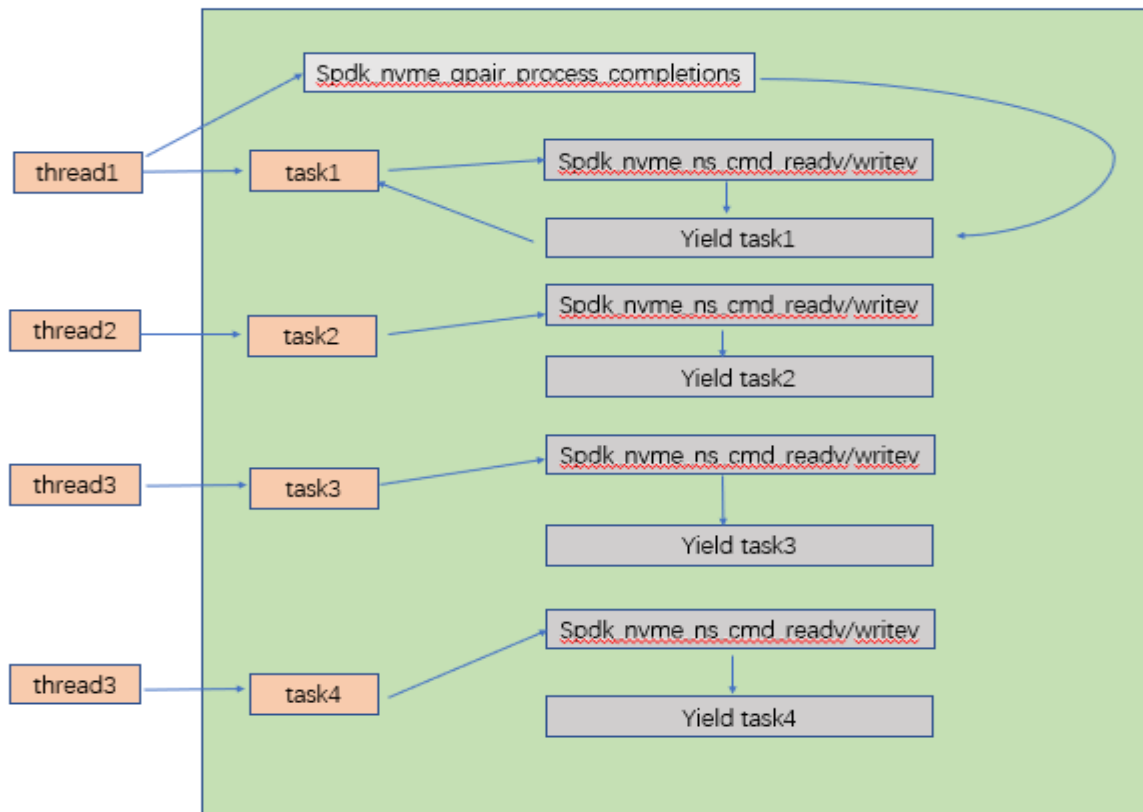


- ✓ 完美支持RDMA、NVMe闪存硬件，基于Intel SDPK软件架构
- ✓ 用户态运行设备驱动，零拷贝机制，大幅降低内核上下文切换性能损耗
- ✓ Polling定点轮循驱动，降低中断开销，高效并发IO处理

SPDK用户态NVMe驱动性能对比



Fusionstor 使用SPDK 处理IO的流程



测试场景	持续时间	Vdbench性能均值												
		i/o	MB/sec	bytes	read	resp	read	write	resp	resp	queue	cpu%	cpu%	
		rate	1024**2	i/o	pct	time	resp	resp	max	stddev	depth	sys+u	sys	
7.3随机读写 性能测试,单主机压测, 8个LUN 520Gx8(未启动SPDK)														
4K数据块,7.3随机读写,队列=16测试,IOPS不限制	30分钟	492023.71	1921.97	4096	70	0.258	0.275	0.22	728.122	0.302	127	25	19.6	
4K数据块,7.3随机读写,队列=32测试,IOPS不限制	30分钟	557196.07	2176.55	4096	70	0.457	0.448	0.479	1358.381	0.689	254.8	34.9	28.7	
4K数据块,7.3随机读写,队列=64测试,IOPS不限制	30分钟	559168.42	2184.25	4096	70	0.913	0.845	1.071	897.202	0.702	510.4	38.1	31.9	
8K数据块,7.3随机读写,队列=8测试,IOPS不限制	30分钟	295836.34	2311.22	8192	70	0.215	0.259	0.112	551.854	0.348	63.5	11.7	8.6	
8K数据块,7.3随机读写,队列=16测试,IOPS不限制	30分钟	433176.71	3384.19	8192	70	0.293	0.339	0.188	1044.549	0.41	127.1	21	16	
8K数据块,7.3随机读写,队列=64测试,IOPS不限制	30分钟	511724.05	3997.84	8192	70	0.997	0.949	1.111	819.263	0.707	510.4	32.9	26.5	
32K数据块,7.3随机读写,队列=8测试,IOPS不限制	30分钟	142374.6	4449.21	32768	70	0.447	0.565	0.174	1006.528	0.661	63.7	6.6	4.5	
32K数据块,7.3随机读写,队列=16测试,IOPS不限制	30分钟	170767.7	5336.49	32768	70	0.747	0.8	0.623	1296.495	0.813	127.5	8.5	5.9	
32K数据块,7.3随机读写,队列=64测试,IOPS不限制	30分钟	170001.66	5312.55	32768	70	3.008	2.66	3.82	1413.265	1.688	511.4	8.6	5.5	
256K数据块,7.3随机读写,队列=4测试,IOPS不限制	30分钟	21053.28	5263.32	262144	70.01	1.511	1.744	0.968	1535.661	1.083	31.8	2.7	1.6	
256K数据块,7.3随机读写,队列=8测试,IOPS不限制	30分钟	21525.95	5381.49	262144	70.01	2.963	2.905	3.1	875.016	1.273	63.8	2.9	1.7	
7.3随机读写 性能测试,单主机压测, 每个主机8个LUN 520Gx8(启动SPDK)														
4K数据块,7.3随机读写,队列=16测试,IOPS不限制	30分钟	770306.81	3009.02	4096	70	0.16	0.2	0.06	23.19	0.29	126.4	39.2	31.6	
4K数据块,7.3随机读写,队列=32测试,IOPS不限制	30分钟	873889.2	3413.63	4096	70	0.29	0.31	0.25	24.84	0.32	253.7	54.8	46.8	
4K数据块,7.3随机读写,队列=64测试,IOPS不限制	30分钟	859147.85	3356.05	4096	70	0.59	0.57	0.64	26.83	0.4	508.7	59.1	51.8	
8K数据块,7.3随机读写,队列=8测试,IOPS不限制	30分钟	324842.1	2537.83	8192	70	0.2	0.25	0.06	21.28	0.37	63.5	12.4	9.1	
8K数据块,7.3随机读写,队列=16测试,IOPS不限制	30分钟	504113.12	3938.38	8192	70	0.25	0.33	0.08	21.58	0.45	127	23	17.3	
8K数据块,7.3随机读写,队列=64测试,IOPS不限制	30分钟	612571.34	4785.71	8192	70	0.83	0.83	0.85	23.45	0.58	509.9	41.1	33.8	
32K数据块,7.3随机读写,队列=8测试,IOPS不限制	30分钟	134259.02	4195.59	32768	70	0.48	0.63	0.11	13.29	0.67	63.7	6.1	4.1	
32K数据块,7.3随机读写,队列=16测试,IOPS不限制	30分钟	170655.97	5333	32768	70	0.75	0.87	0.47	19.67	0.67	127.6	8.4	5.8	
256K数据块,7.3随机读写,队列=4测试,IOPS不限制	30分钟	20256.87	5064.22	262144	70	1.57	1.93	0.72	12.61	1.04	31.8	2.6	1.5	
256K数据块,7.3随机读写,队列=8测试,IOPS不限制	30分钟	21529.83	5382.46	262144	70	2.96	3.04	2.78	17.6	1.16	63.8	3	1.7	
大变化IO size, 随机读写 性能测试,2个主机并发压测, 每个主机8个LUN 520x8														

Fusionstor 启动Spdk和不启动spdk性能对比

编程模型：同步非阻塞

- 任务调度：coroutine
- 事件处理：polling
- 多核同步：session based hash

数据路径：kernel-bypass

- Network：RDMA / DPDK
- Flash：SPDK
- Mem：hugepage

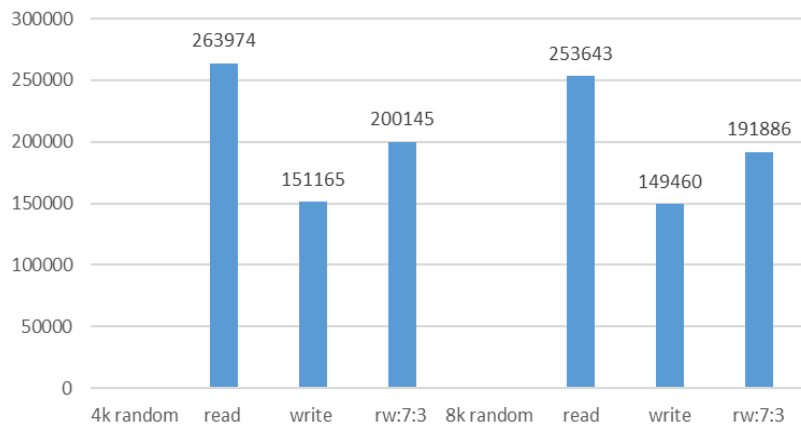
procs		memory				swap		io		system			cpu			
r	b	swpd	free	buff	cache	si	so	bi	bo	in	cs	us	sy	id	wa	st
15	0	14009496	227716		0 702480	0	0	0	0	0 14714	1747	58	0	42	0	0
14	0	14009496	228212		0 702480	0	0	0	0	0 14688	1753	58	0	42	0	0
14	0	14009496	228292		0 702400	0	0	0	0	0 14681	1803	58	0	42	0	0
14	0	14009496	228464		0 702360	0	0	0	0	0 14837	1665	58	0	42	0	0
14	0	14009496	228464		0 702360	0	0	0	0	0 14940	2247	58	0	42	0	0
14	0	14009496	227352		0 702392	0	0	0	0	0 17803	1794	58	0	42	0	0
14	0	14009496	226360		0 702392	0	0	0	56	53277	1844	58	0	42	0	0
14	0	14009496	226388		0 702396	0	0	0	0	0 83691	1872	58	0	42	0	0
14	0	14009496	226360		0 702396	0	0	0	0	0 15310	1778	58	0	42	0	0
14	0	14009496	227052		0 702396	0	0	0	0	0 14698	1755	58	0	42	0	0
14	0	14009496	227052		0 702396	0	0	0	0	0 15189	1909	58	0	42	0	0
14	0	14009496	227232		0 702400	0	0	0	0	0 14800	1748	58	0	42	0	0

华云网际

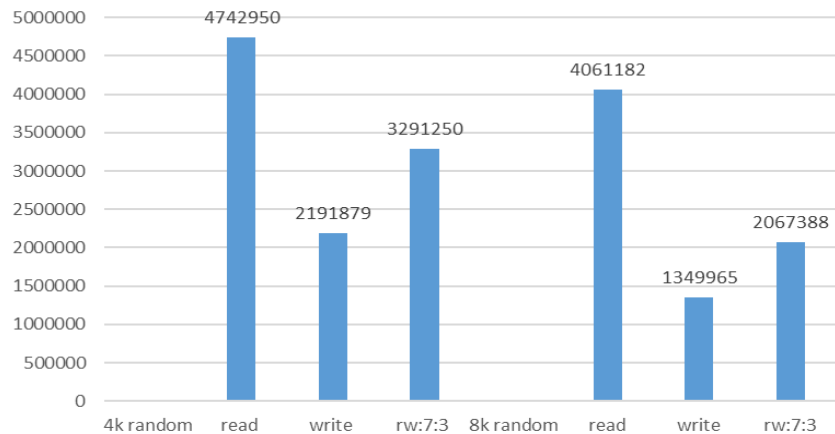
完全消除上下文切换

Kernel-bypass带来的实际性能表现

1 volume iops



24 volumes iops





THANK YOU