NUMA-ICON

一种自适应NUMA调度框架

唐辉
华为操作系统高级工程师

目录

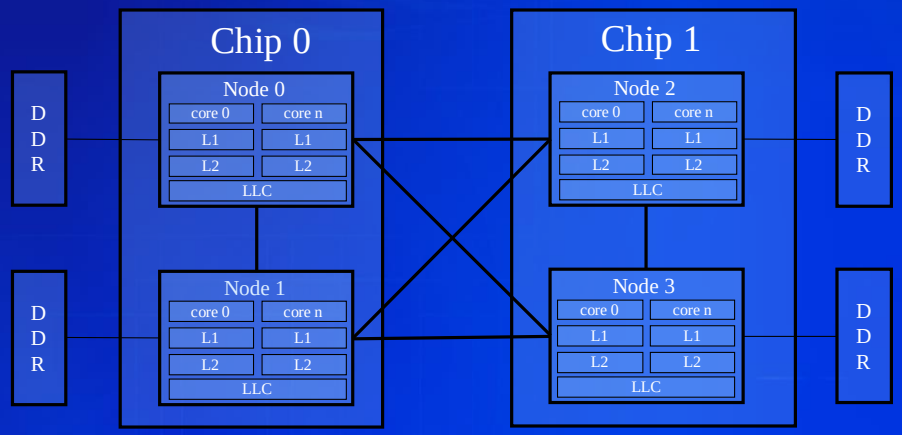
CONTENT

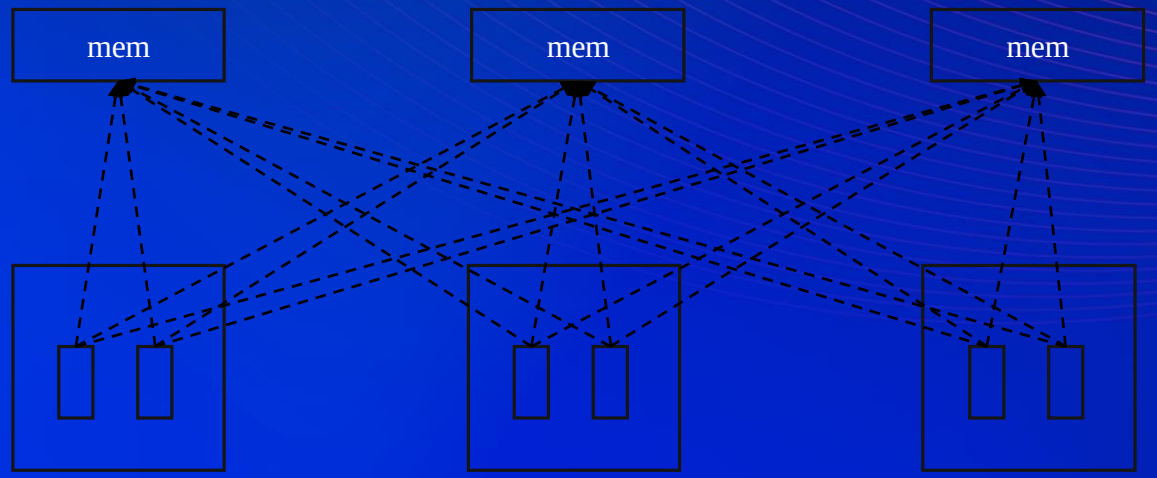
1

Part One

相关背景简介

NUMA问题的认识





跨NUMA访存时延大，比NUMA内时延增大30%以上

- NUMA架构

本地内存访问相比远端内存访问，拥有更高的带宽或更低的延迟。

- 硬件架构发展趋势

摩尔定律失效，硬件架构Scale-up收益空间收窄，Scale-out是硬件架构主要发展的方向。

- Linux 的现状

以吞吐为中心，强调负载均衡，会因为负载均衡的约束，而打破全局跨NUMA访存最优的状态。

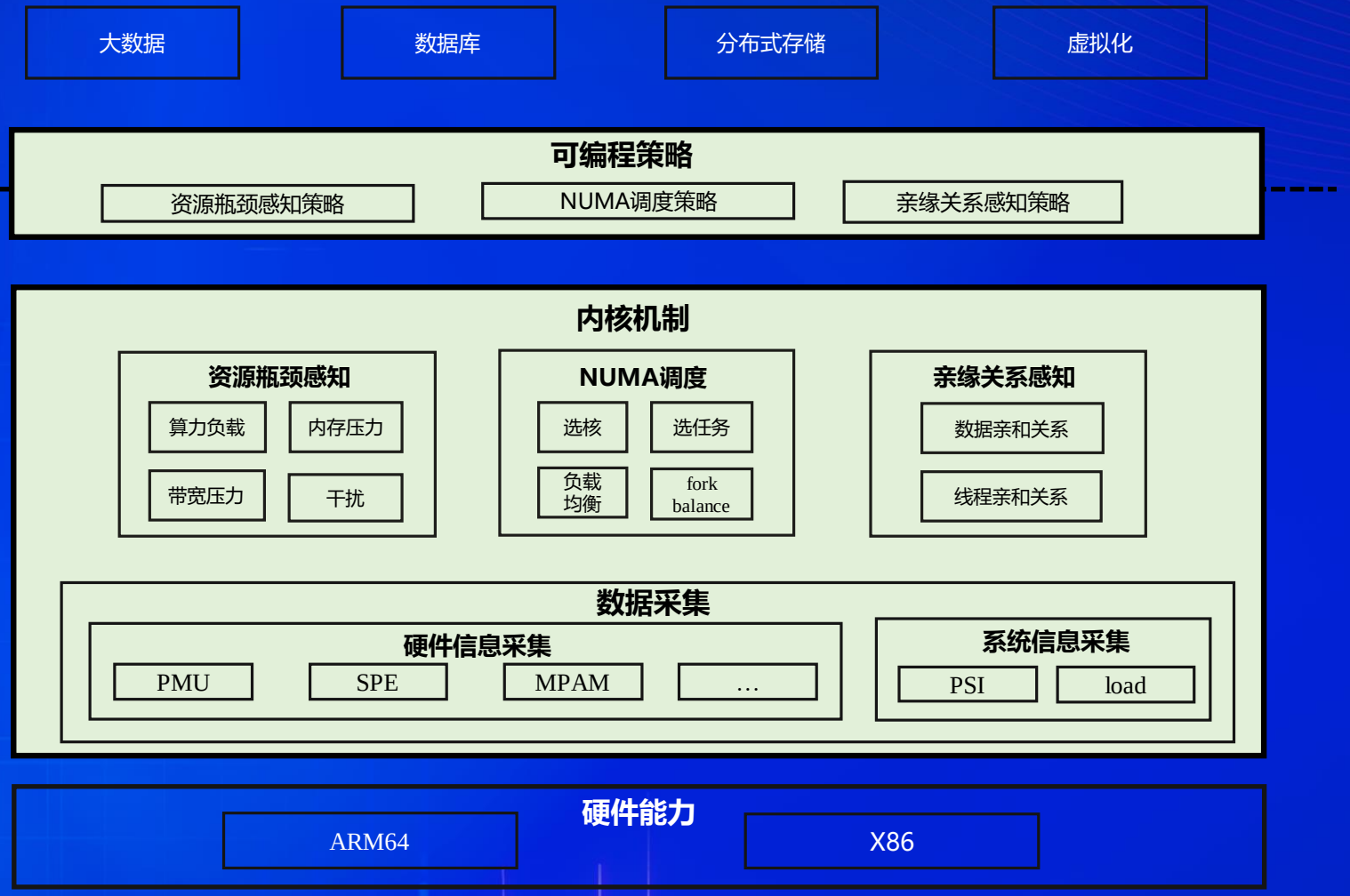
AutoNuma机制遵守负载均衡的约束，追求每个线程跨NUMA最优状态（局部最优），未达到全局跨NUMA访存最优。

2

Part Two

自适应NUMA调度框架

NUMA-ICON逻辑架构



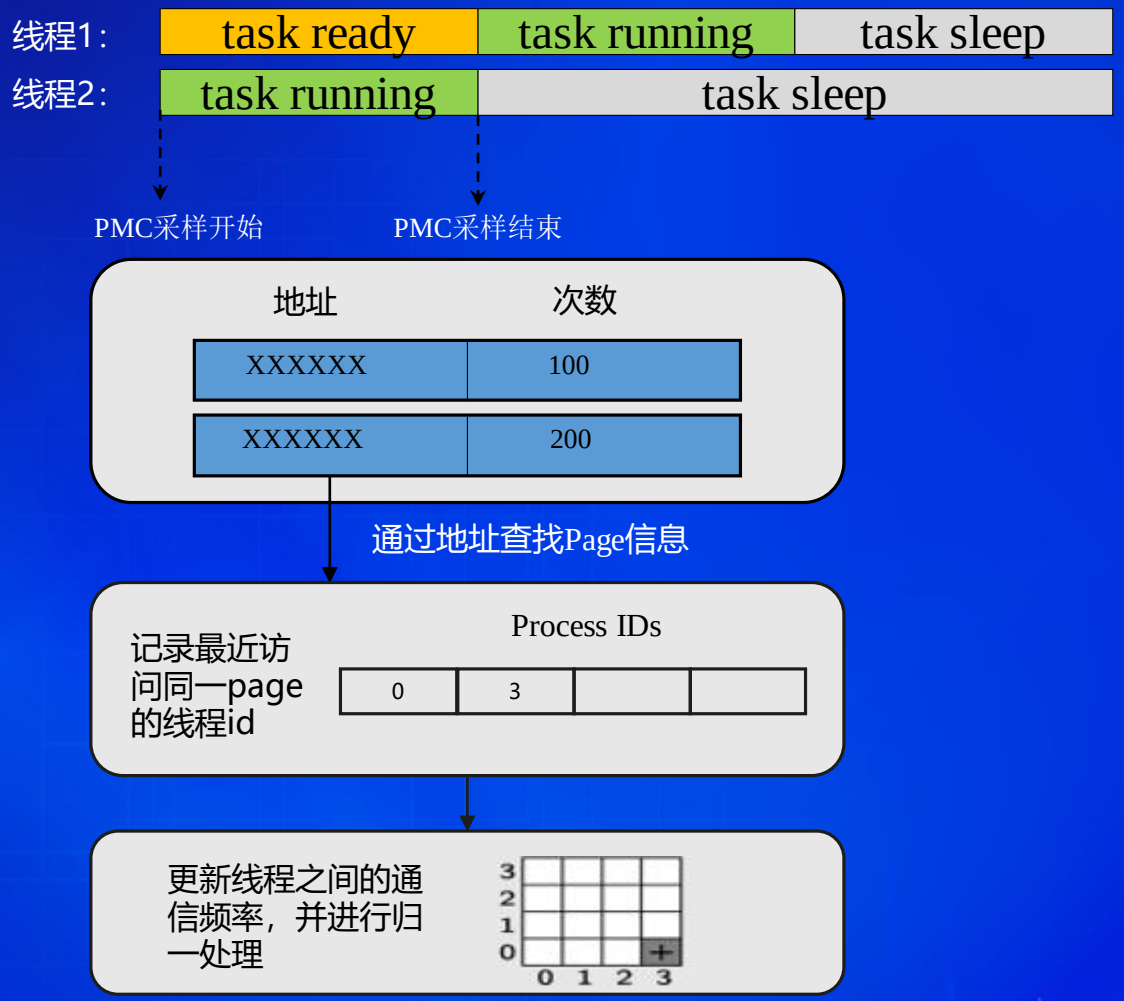
3

Part Three

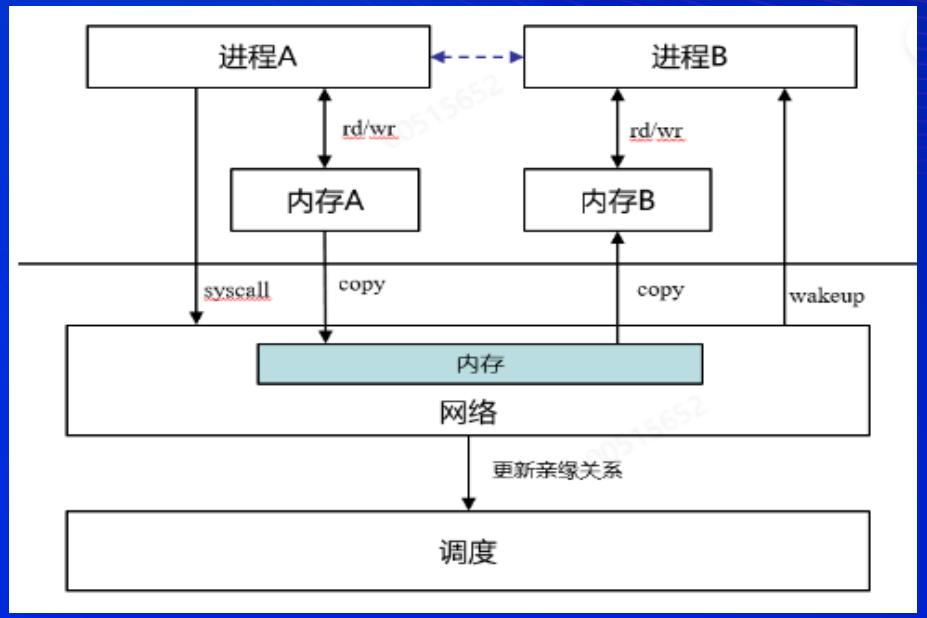
关键技术方案

轻量级采样精准识别线程的亲缘关系及强弱

线程运行轨迹图

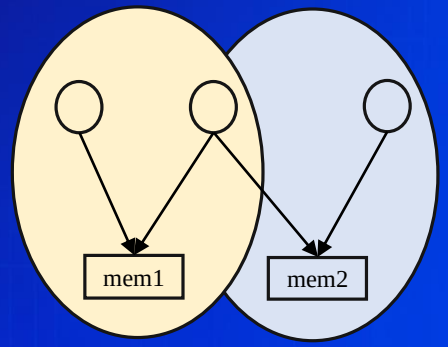


通过网络协议栈四元组关系，识别通信的两个线程之间的亲缘关系。



- 在线程调度上下文，通过PMC采样技术（SPE寄存器）精准识别线程的访存行为，解决pagefault识别线程访存行为精度差，开销高的问题；
- 记录最近访问统一内存页的多个线程，将此多个线程纳入到同一个任务组中；
- 更新线程之间的通信频率，通过通信频率、带宽来表示线程之间的亲缘关系；

亲缘关系组融合与维护



关系组融合

- 线程与线程之间通过访存信息及访存频率识别亲缘关系；
- 两个亲缘关系组如果存在交互，亲缘关系组融合；
- 两个亲缘关系组融合需要考虑融合方向，小负载（算力 + 内存）的向大负载的融合（涉及迁移成本）
- 避免超级巨无霸亲缘关系组，需要设置亲缘关系组的大小，在大于一定条件下（如负载，任务个数，内存占用量等），禁止亲缘关系组的融合；

频次	t1	t2	t3
t1	-	20	30
t2	20	-	40
t3	30	40	-

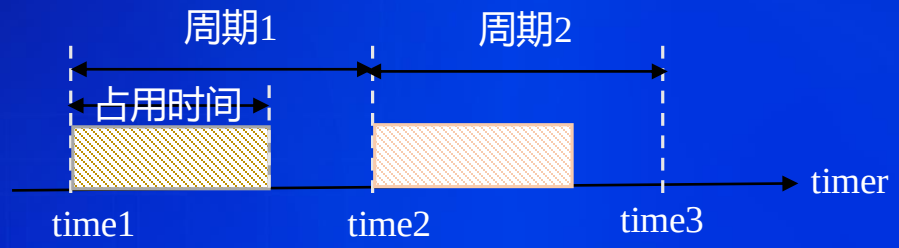
强弱	t1	t2	t3
亲缘	弱	中	强

亲缘关系组维护

- 记录线程访问共享内存的频率；
- 亲缘关系需要衰减，按照标准的衰减算法 s = s/2 ^ time 衰减
- 根据亲缘关系的频次确定亲缘关系强弱，为有效降低开销，在负载均衡中只需要使用亲缘强弱信息；
- 定期维护亲缘关系强弱信息，在调度关键路径更新强弱信息，当亲缘关系衰减为0时，延时XX时间清理出亲缘关系组；

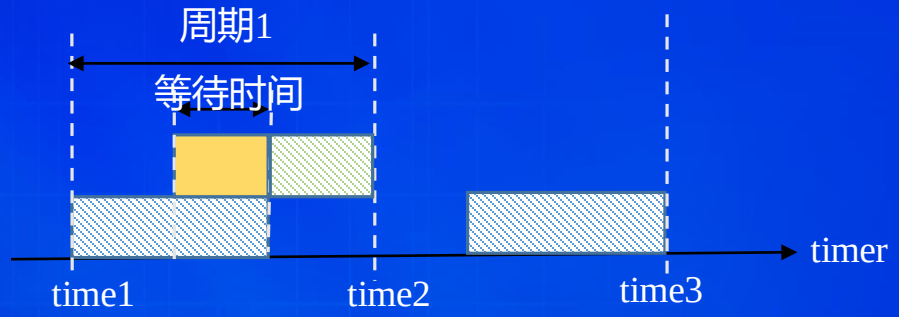
负载和干扰建模，感知NUMA资源瓶颈

资源负载度量



通过资源占用时间/周期，来衡量资源的负载情况，用来度量资源的繁忙程度；

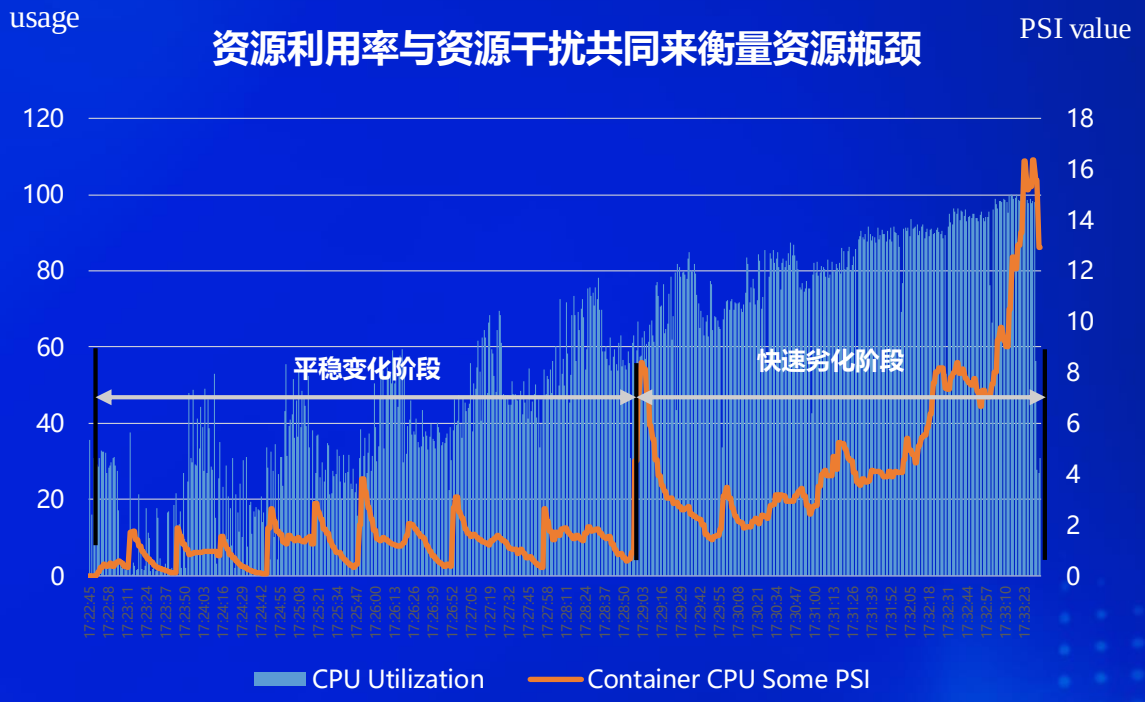
资源干扰度量



通过资源等待时间/周期，来衡量资源的干扰情况，用来度量资源的争抢程度；

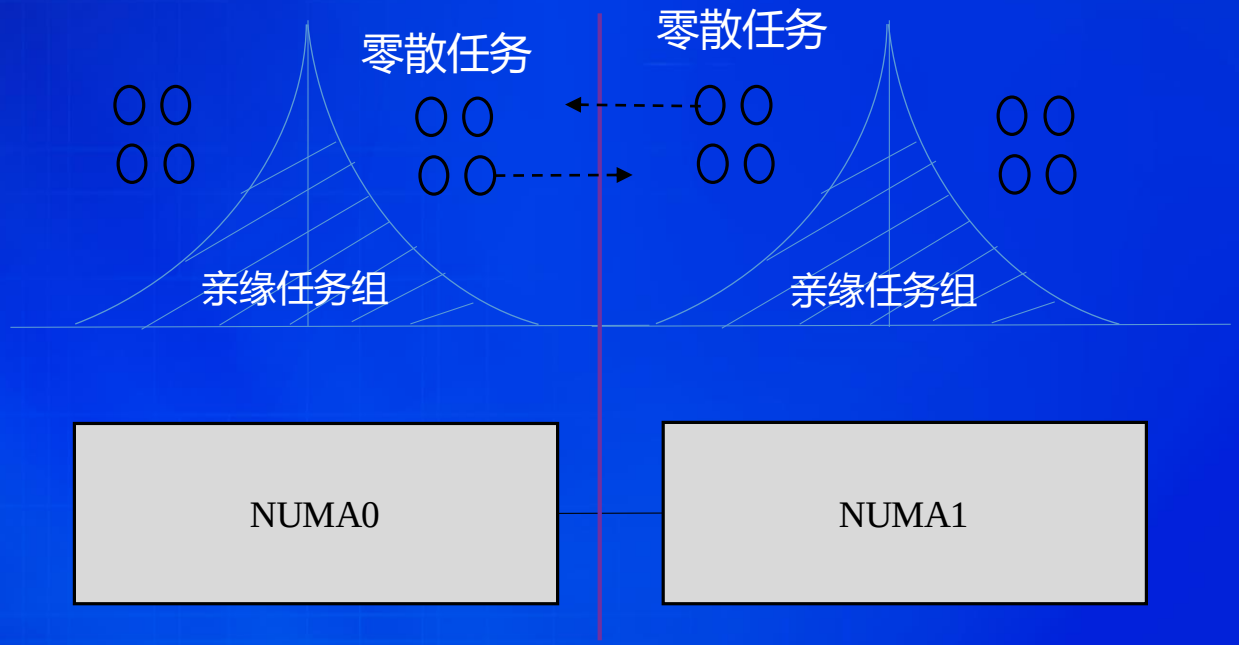
当CPU、内存、IO利用率或者干扰达到阈值，则判定NUMA达到资源瓶颈；

资源利用率与资源干扰共同来衡量资源瓶颈



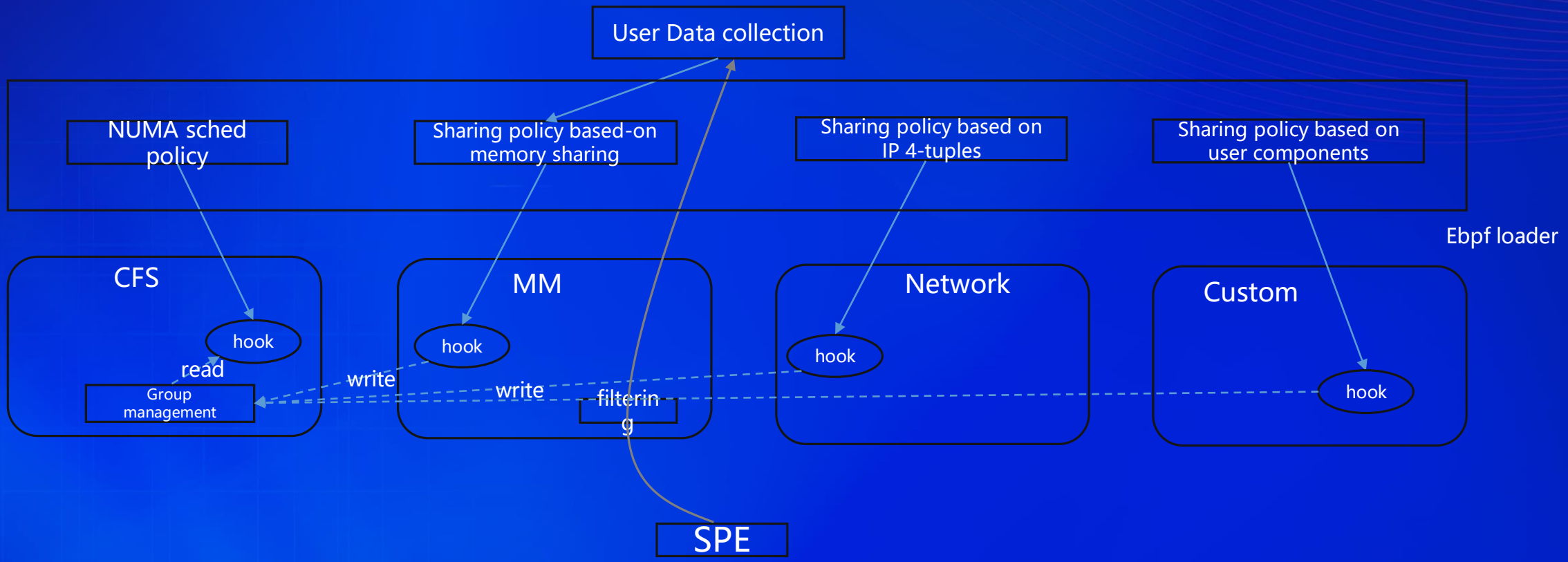
动态决策NUMA内还是NUMA间调度

打破操作系统现有调度模型，根据亲缘关系和资源瓶颈，修改选核与负载均衡机制



- 进程fork时，按照interleave策略，每个进程选择一个NUMA node；
- 在线程唤醒时，根据任务组的访存信息，选择一个最优的node，判断此node是否存在资源瓶颈，如果存在资源瓶颈则选择进程组其他最优的node；最优node选择策略为：此node为进程组访存最频繁的node节点；
- 在负载均衡时，按照访问此NUMA内存频率的强弱对任务进行排序，越弱的先迁移。当任务迁移时，如果此NUMA为任务最优的node节点，判断NUMA是否到达资源瓶颈，如果未达到资源瓶颈，则不允许迁移；

结合ebpf构建多维度亲缘关系和定制调度策略



4

Part Three

效果展示

Ceph测试

B	C	D	E	F	G	H	I	J	N	O	P	Q	R	S	T	U	V
			fiobaselnumactl			fio numa v1			fiobase			(fionuma-fiobasel)/fiobase			(fiobasenumactl-fiobasel)/fiobase		
序号	场景		吞吐IOPS	Latency (lat)	bandwidth MiB/s	吞吐IOPS	Latency (lat)	bandwidth MiB/s	吞吐IOPS	Latency (lat)	bandwidth MiB/s	吞吐IOPS	Latency (lat)	bandwidth MiB/s	吞吐IOPS	Latency (lat)	bandwidth
1	100%读	读	5858.67	170.15	22.9	5690.32	175.19	22.2	5463.86	182.5	21.3	4.14%	-4.01%	4.23%	7.23%	-6.77%	7.51%
2		读	5852.51	170.35	45.7	5534.19	180.18	43.2	5551.52	179.62	43.3	-0.31%	0.31%	-0.23%	5.42%	-5.16%	5.54%
3		读	38034.15	3367.28	148	36889.98	3471.17	144	28863.95	4435.99	113	27.81%	-21.75%	27.43%	31.77%	-24.09%	30.97%
4		读	38092.89	3362.28	297	38038.53	3366.83	297	29734.94	4307.67	232	27.93%	-21.84%	28.02%	28.11%	-21.95%	28.02%
5		读	40025.71	3200.61	9997	38859.11	3296.47	9706	29217.58	4382.36	7301	33.00%	-24.78%	32.94%	36.99%	-26.97%	36.93%
6		读	24083.51	5312.44	23500	24578.14	5210.43	24000	20703.8	6184.51	20200	18.71%	-15.75%	18.81%	16.32%	-14.10%	16.34%
7		读	45262.84	2829.72	11000	36590.57	3499.89	9142	28396.87	4512.24	7091	28.85%	-22.44%	28.92%	59.39%	-37.29%	55.13%
8		读	25141.52	5093.65	24500	24657.36	5195.27	24100	18711.12	6843.7	18300	31.78%	-24.09%	31.69%	34.37%	-25.57%	33.88%
1	7:3读写	读	3792.28	175.87	14.8	3723.59	179.51	14.5	3607.94	184.94	14.1	3.21%	-2.94%	2.84%	5.11%	-4.90%	4.96%
1		写	1626.21	202.9	6.502	1596.87	205.75	6.385	1547.14	213.2	6.185	3.21%	-3.49%	3.23%	5.11%	-4.83%	5.13%
2		读	3720.36	175.83	29	3534.71	185.81	27.6	3556.58	184.21	27.8	-0.61%	0.87%	-0.72%	4.60%	-4.55%	4.32%
2		写	1595.35	214.89	12.5	1515.44	224.69	11.8	1524.85	224.35	11.9	-0.62%	0.15%	-0.84%	4.62%	-4.22%	5.04%
3		读	25078.74	3566.98	97.9	26123.27	3423.91	102	19877.68	4501.94	77.6	31.42%	-23.95%	31.44%	26.17%	-20.77%	26.16%
3		写	10751.19	3592.17	42	11199.16	3450.63	43.7	8519.88	4526.03	33.3	31.45%	-23.76%	31.23%	26.19%	-20.63%	26.13%
4		读	25083.27	3564.89	196	24482.36	3650.65	191	20522.03	4358.27	160	19.30%	-16.24%	19.38%	22.23%	-18.20%	22.50%
4		写	10753.24	3598.55	83.9	10495.45	3689.72	81.9	8796.08	4392.66	68.7	19.32%	-16.00%	19.21%	22.25%	-18.08%	22.13%
5		读	10201.65	7534.38	2548	10139.26	7556.61	2533	10158	7487.72	2537	-0.18%	0.92%	-0.16%	0.43%	0.62%	0.43%
5		写	4372.13	11716.7	1092	4345.11	11840.84	1086	4353.1	11954.41	1087	-0.18%	-0.95%	-0.09%	0.44%	-1.99%	0.46%
6		读	2730.2	30728.52	2727	2725.88	30564.51	2724	2723.2	30630.08	2721	0.10%	-0.21%	0.11%	0.26%	0.32%	0.22%
6		写	1170.51	37784.22	1169	1168.71	38302.85	1168	1167.44	38279.21	1167	0.11%	0.06%	0.09%	0.26%	-1.29%	0.17%
1	5:5读写	读	17805.08	3584.3	69.5	17554.71	3641.19	68.5	14639.39	4361.26	57.1	19.91%	-16.51%	19.96%	21.62%	-17.82%	21.72%
1		写	17798.62	3610.81	69.5	17549.54	3656.72	68.5	14633.63	4389.5	57.1	19.93%	-16.69%	19.96%	21.63%	-17.74%	21.72%
2		读	17366.25	3670.33	136	17086.23	3730.06	133	14207.33	4490.86	111	20.26%	-16.94%	19.82%	22.23%	-18.27%	22.52%
2		写	17359.58	3705.97	136	17079.17	3769.19	133	14201.19	4523.99	111	20.27%	-16.68%	19.82%	22.24%	-18.08%	22.52%
3		读	4518	11637.27	1129	4504.8	12092.93	1125	4507.03	11705.79	1126	-0.05%	3.31%	-0.09%	0.24%	-0.59%	0.27%
3		写	4513.22	16732.26	1127	4500.1	16359.16	1124	4502.25	16735.36	1125	-0.05%	-2.25%	-0.09%	0.24%	-0.02%	0.18%
4		读	1177.91	51444.31	1177	1179.12	50763.88	1178	1179.75	51290.98	1179	-0.05%	-1.03%	-0.08%	-0.16%	0.30%	-0.17%
4		写	1176.36	57390.47	1175	1177.62	57933.03	1177	1178.19	57362.46	1177	-0.05%	0.99%	0.00%	-0.16%	0.05%	-0.17%
1	100%写	写	4933.67	202.15	19.3	4721.5	211.25	18.4	4724.05	211.13	18.4	-0.05%	0.06%	0.00%	4.44%	-4.25%	4.89%
2		写	4633.22	215.3	36.2	4570.28	218.22	35.7	4442.24	224.58	34.7	2.88%	-2.83%	2.88%	4.30%	-4.13%	4.32%
3		写	33537.73	3818.8	131	33736.58	3796.03	132	27643.63	4632.96	108	22.04%	-18.06%	22.22%	21.32%	-17.57%	21.30%
4		写	34344.14	3729.66	268	33994.63	3767.22	265	28249.12	4532.13	221	20.34%	-16.88%	19.91%	21.58%	-17.71%	21.27%
5		写	4633.4	27647.93	1157	4641.15	27604.63	1159	4635.34	27635.78	1158	0.13%	-0.11%	0.09%	-0.04%	0.04%	-0.09%
6		写	1180.35	108514.67	1179	1179.38	108585.72	1179	1180.6	108493.22	1180	-0.10%	0.09%	-0.08%	-0.02%	0.02%	-0.08%
7		写	1180.15	108535.93	1179	1180.65	108507.59	1180	1180.4	108518.62	1179	0.02%	-0.01%	0.08%	-0.02%	0.02%	0.00%
8		写	1180.49	108508.33	1180	1180.65	108.49	1180	1180.34	108516.61	1179	0.03%	-99.90%	0.08%	0.01%	-0.01%	0.08%

FIO（client + 3个server） 验证收益：较原生linux提升**15-24%**

Redis测试

	回环单pipeline		回环多pipeline	
	set	get	set	get
自适应numa调度(avg)	104815	101847	720196	851047
baseline绑node0(avg)	103887	101897	705231	858614
baseline(avg)	78600.2	79673.4	669032	804393
提升	33.35%	27.83%	7.65%	5.80%

redis-server+redis-benchmark， 验证收益：较原生linux提升**5% -30%**

欢迎感兴趣的同学加入群聊，共同探索OS技术的真谛

谢谢聆听！

CHINA LINUX KERNEL
中国Linux内核开发者大会

華中科技大學
网络空间安全学院
School of Cyber Science and Engineering, HUST

第19届中国Linux内核开发者大会

赞助单位

华中科技大学 | HUAWEI | 阿里云 | FUJITSU | 腾讯云
oppo | 火山引擎 | vivo | 龙芯中科 LOONGSON TECHNOLOGY | 蚂蚁集团 ANT GROUP

支持单位

清华大学 Tsinghua University | 迪捷软件 DIGIPROTO | OpenEuler | OpenAnolis 龙蜥社区 | OpenCloudOS

支持社区&媒体

CSDN | 云巅论剑 | sf 思否 | 阅码场 yamcode | InfoQ 极客传媒
51CTO | 开源江湖

2024年10月　湖北 • 武汉

華中科技大學