QCon
全球软件开发大会

InfoQ 极客传媒

超节点时代的开源基础软件构建实践

胡欣蔚

openEuler 技术委员会主席

目录

InfoQ 极客传媒

北京

QCon 全球软件开发大会

会议时间：4月10-12日

- 大模型赋能 AIOps
- 越挫越勇的大前端
- 云上业务架构演进
- 多模态大模型及应用
- 海外 AI 应用创新实践

北京

AiCon 全球人工智能开发与应用大会

会议时间：6月27-28日

- 端侧智能
- AI Agent
- 多模态大模型
- 金融+大模型

上海

QCon 全球软件开发大会

会议时间：10月23-25日

- AI Agent
- 大模型训练推理
- 端侧 AI
- 搜推深度融合
- 数智金融

4月 — 5月 — 6月 — 8月 — 10月 — 12月

上海

AiCon 全球人工智能开发与应用大会

会议时间：5月23-24日

- 通用大模型
- AI Agent
- 垂直领域应用
- 模型可解释性

深圳

AiCon 全球人工智能开发与应用大会

会议时间：8月22-23日

- 模型效率与部署
- 多模态大模型
- 大模型安全
- 智能硬件

北京

AiCon 全球人工智能开发与应用大会

会议时间：12月19-20日

- 通用大模型
- 智能硬件
- LMOps
- 具身智能

极客邦科技 2025 年会议规划

促进软件开发及相关领域知识与创新的传播

参会咨询

查看会议

通用计算：需要提升资源利用率，优化或卸载IO处理，解决CPU瓶颈

虚拟化：云厂商资源利用率普遍较低

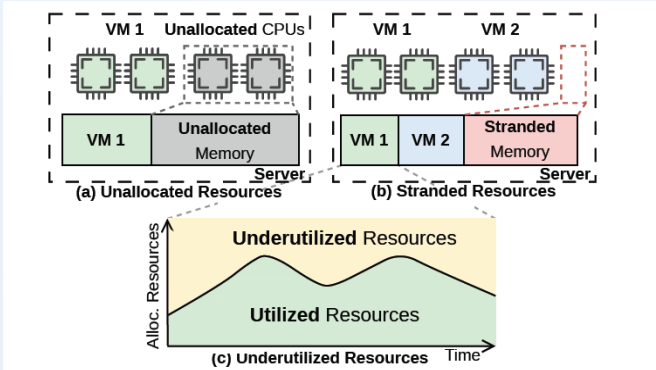


Figure 1. Examples of the causes of low resource utilization.

a) **资源未分配**：平台未出售或保留资源
b) **资源搁浅**：由于服务器中缺少其他资源而无法分配
c) **资源未充分利用**：已分配给虚拟机，但未使用

From：Coach: Exploiting Temporal Patterns for All-Resource Oversubscription in Cloud Platforms

大数据：业务负载动态变化，资源无法精准预测，导致过度分配

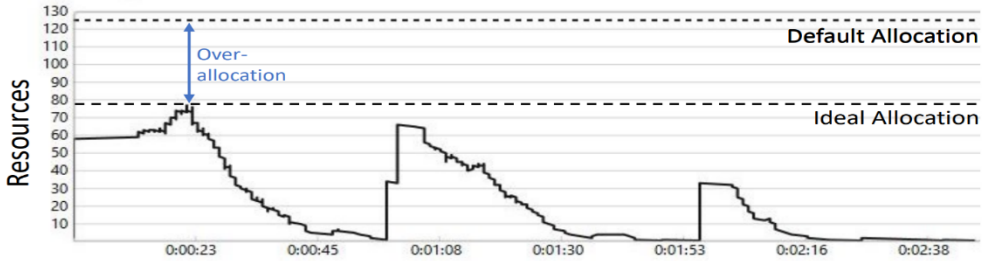
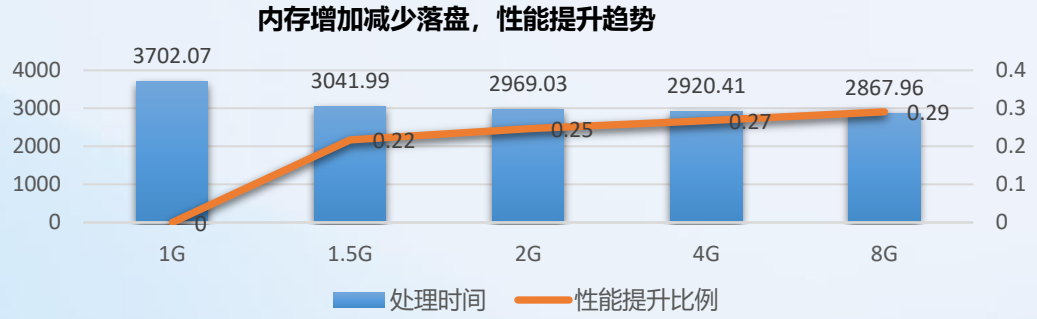


Figure 1: Resource usage in a typical SCOPE job over time.

默认资源分配和实际峰值间存在巨大差距，过度配置会导致资源浪费，降低集群利用率

From：AutoToken: Predicting Peak Parallelism for Big Data Analytics at Microsoft

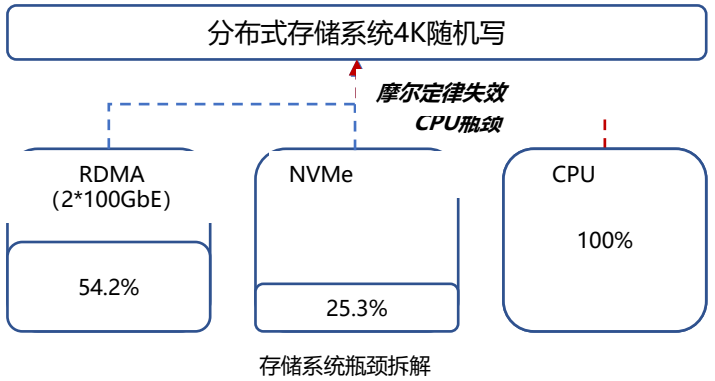
数据库：内存不足时数据落盘，导致SQL执行效率大幅下降



提升Work Memory大小，减少数据落盘，复杂SQL执行性能将提升

From：开源数据库GreenPlum实测数据

分布式存储：高速介质与高性能网络发展，CPU成为瓶颈



NVMe SSD性能 + 网络性能提升幅度大于CPU性能提升，目前在存储系统中CPU已成为瓶颈，预计后续发展CPU依然是瓶颈

From：分布式存储软件测试数据

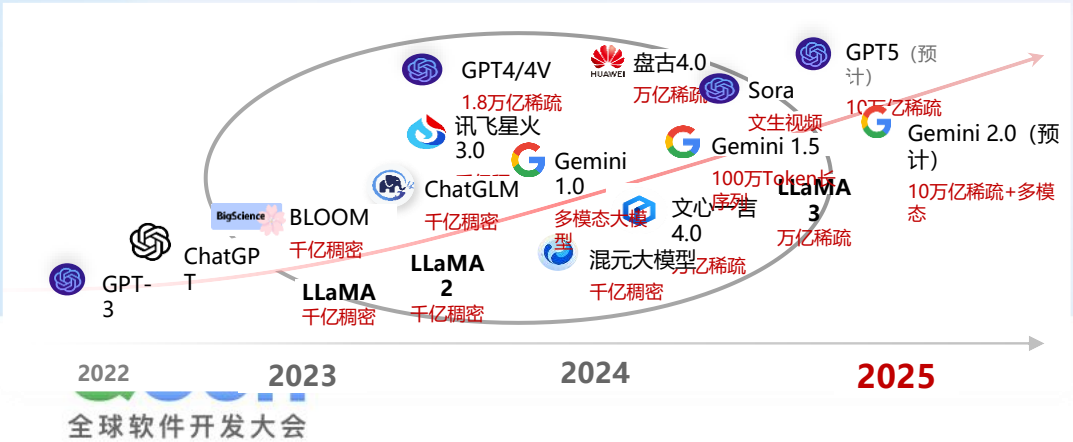
智能计算：需要高带宽、低时延、规模组网、高可靠的互联技术

1、高带宽：训练单模型的算力规模持续指数级别增长，由于GPU/NPU内存容量和算力的限制，需要将训练任务切分到多卡上进行并行训练，这就引入了额外的并行通信，卡间通信量随着模型参数量的增加急剧增长。

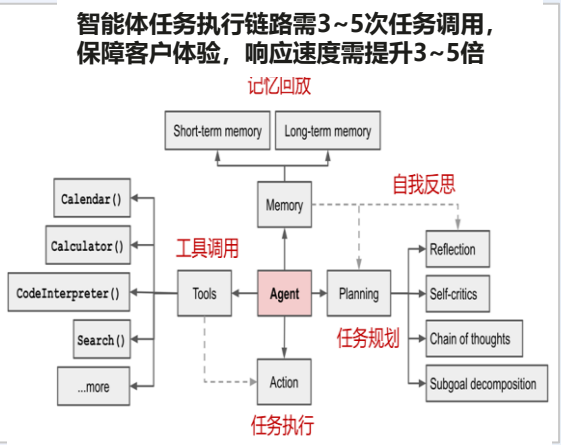
Model	Estimated Compute	Growth
GPT-2 (2019)	~4e21 FLOP	
GPT-3 (2020)	~3e23 FLOP	+ ~2 OOMs
GPT-4 (2023)	8e24 to 4e25 FLOP	+ ~1.5–2 OOMs

从 GPT-2 到 GPT-4,训练算力增加了3000-10000倍

3、规模组网：随着AI大模型的训练任务由百亿稠密走向万亿稀疏，摸高十万、百万亿稀疏，需要的算力从P级增长到10E级，对规模组网的诉求也日益增长。



2、低时延：智能语音对话的实时交互、 慢“思考”超长Reasoning过程的低延迟等待要求，AI Agent应用的机-机交互场景的多任务快速响应，极致用户体验正驱动推理业务向极低时延下的高吞吐性能目标演进。



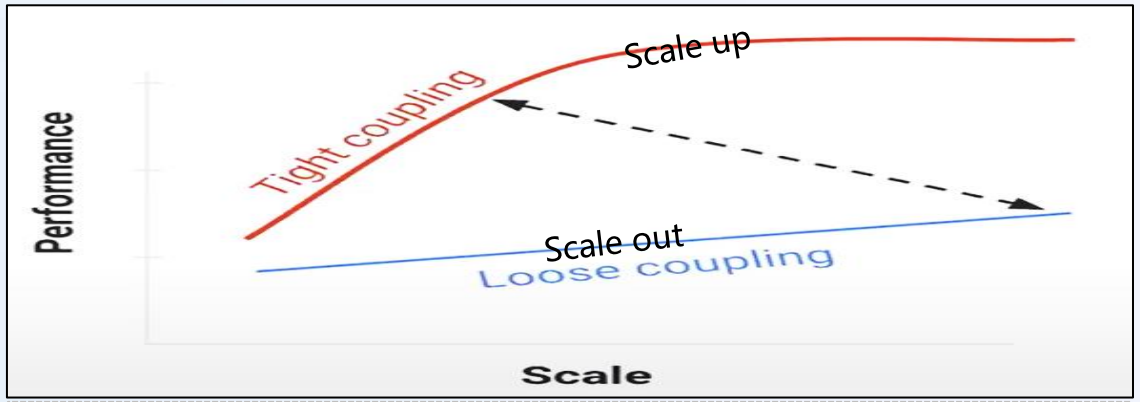
4、高可靠：由于AI训练时需要对所有计算单元的计算结果进行规约等操作，单卡故障会影响整个整个训练集群，随着模型参数量的增长，AI集群的规模的扩大，集群可靠性MTBF由天级降低为小时级，减小集群停机时间，提升运行稳定性是AI大模型训推集群普遍存在的挑战。

模型	集群大小	训练总时长	MTBF
OPT-175B	1K	60days	19.2 hours
Llama3.1-405B	16K	54days	2.78 hours

From：OPT: Open Pre-trained Transformer Language Models， https://arxiv.org/abs/2205.01068
From：The Llama 3 Herd of Models， https://arxiv.org/abs/2407.21783

高速互联的多样性对等算力构成超节点，是硬件发展的趋势

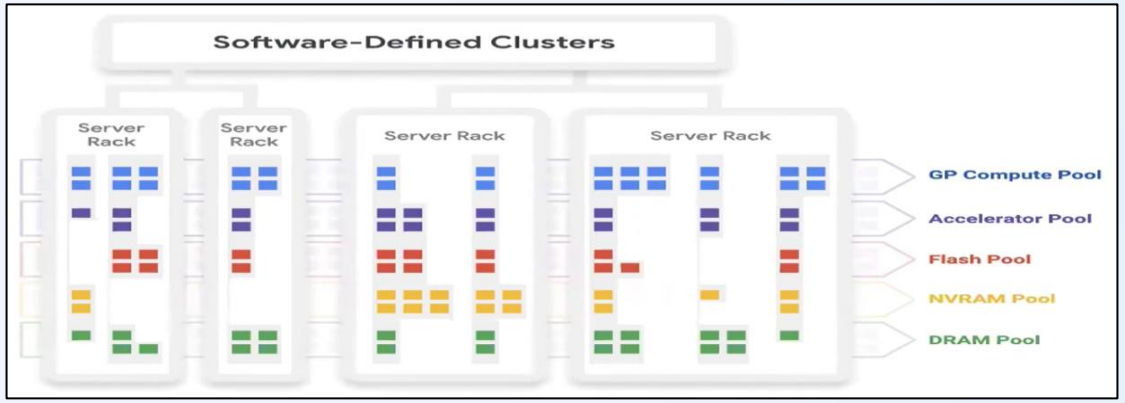
Scale out -> Scale up



- 传统松耦合思路导致器件更大规模和更低的性能获得
- 高带宽、低时延通讯和远程内存语义形成紧耦合分布式系统，在一定规模下（1K Server，200K core）具备更高性能

from:《Coming of Age in the Fifth Epoch of Distributed Computing》

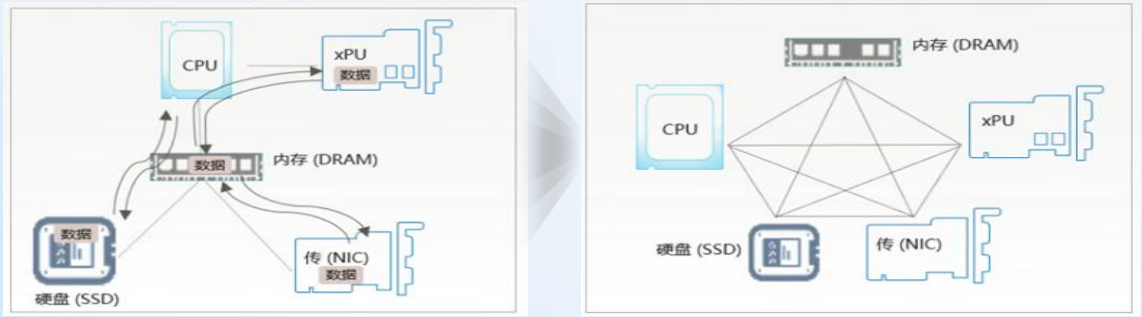
以服务器为基础的硬件系统 -> 集群+资源池化的硬件系统



- 传统分布式以服务器节点为基础，带来负载不均衡、资源碎片化、算力与IO不协调带来的冗余浪费，是影响分布式系统的关键
- 单纯工艺引进带来的红利，没法被应用感知，需要打破构成算力基础4大件之间的语义及空间隔离

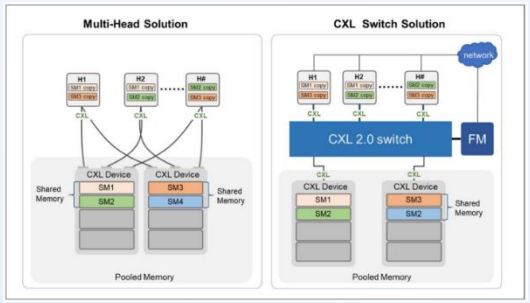
from:《Coming of Age in the Fifth Epoch of Distributed Computing》

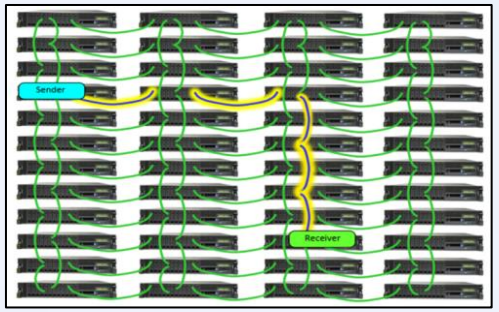
CPU为中心的异构计算 -> 多样性算力对等计算架构



- CPU作为主控节点，承担任务调度、数据搬运和加速器协调，存在性能瓶颈，且一旦故障将导致整系统瘫痪
- 不同计算单元以对等方式协同工作，可以实现最优算力匹配，降低数据搬运开销，且去中心化，避免单点故障

基于总线网络的跨节点通信-> 基于内存语义的消息通信





基于内存共享的消息通信

- Scale up域内跨芯片和节点通信时，传统的基于总线/网络的通信开销大，灵活性低，在时延敏感性业务上表现不友好
- 基于内存共享的消息通讯代替网络通讯，实现极低通讯时延

from:《TxCocket: an innovative solution for efficient cross-node data transmission enabled by CXL-based shared memory》

业界高速互联总线百家争鸣，Scale-Up超节点已是技术热点

互联网厂商开始自研AI芯片和超节点

互联网厂商	AI芯片	Scale up协议	超节点节奏
Google	博通/MTK	ICI（私有）	已商用
AWS	Marvell /Alchip	NeuronLinkV3	已商用
腾讯	沐曦/壁仞	ETH-X	未商用
阿里	平头哥	ALS（UALink）	128卡超节点已发布
字节	自研	ETHLink（SUE）	未商用

设备商利用自身优势，推出不同解决方案，积极应对

设备商	Scale up协议	协议/芯片开放模式	超节点开放模式
英伟达	NVLink	标准封闭 授权IP或Chiplet	1、提供NVL超节点 2、允许三方GPU/CPU二选一接入NVL系统
AMD	UAlink	标准开放 Marvell/Synopsys提供IP	1、提供超节点 2、交换机由第三方提供
博通	SUE	标准开放	1、不提供超节点， 2、提供定制ASIC和交换芯片
华为	UB	标准开放	提供灵衢超节点，已商用发布

总线变化，超节点池化和对等架构，需要系统基础软件重构发挥算力优势

超节点基础软件的挑战

那么代价是什么

- 爆炸半径变大
- 连接方式复杂度高
- 静默错误影响提升

1 可靠性降低

- 异构算力
- 异质内存
- 跨节点访存和通讯带宽共享，时延互相干扰

2 理论与实践的差距

- 基础软件面向超节点和传统服务器集群分别构建
- 业务超出单一范式

3 已有业务迁移复杂

基础软件面临多方面挑战

复杂应用：“单一负载”走向“多样负载异构融合”

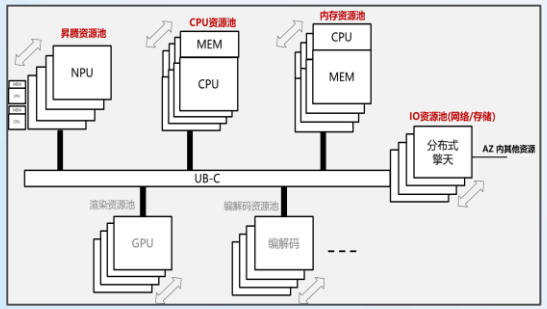
单计算范式为主，烟囱式构建

多计算范式融合的新形态应用

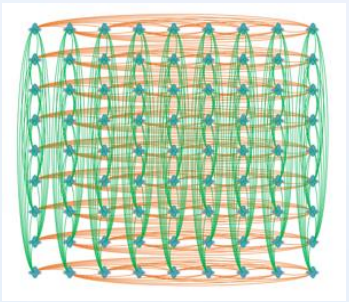
- 大模型应用：Agent、RAG、Compound AI
- 大模型训推：RLXF、PD分离、O1
- 数智融合：大数据+AI
- 函数微服务融合
- …

复杂硬件：多样异构、大规模高速互联

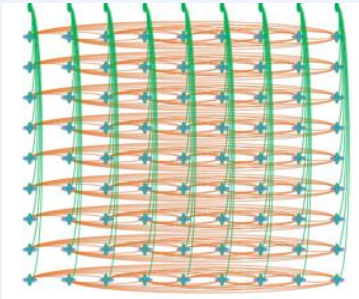
异构算力对等互联



2D Mesh-Mesh



2D Star-Mesh

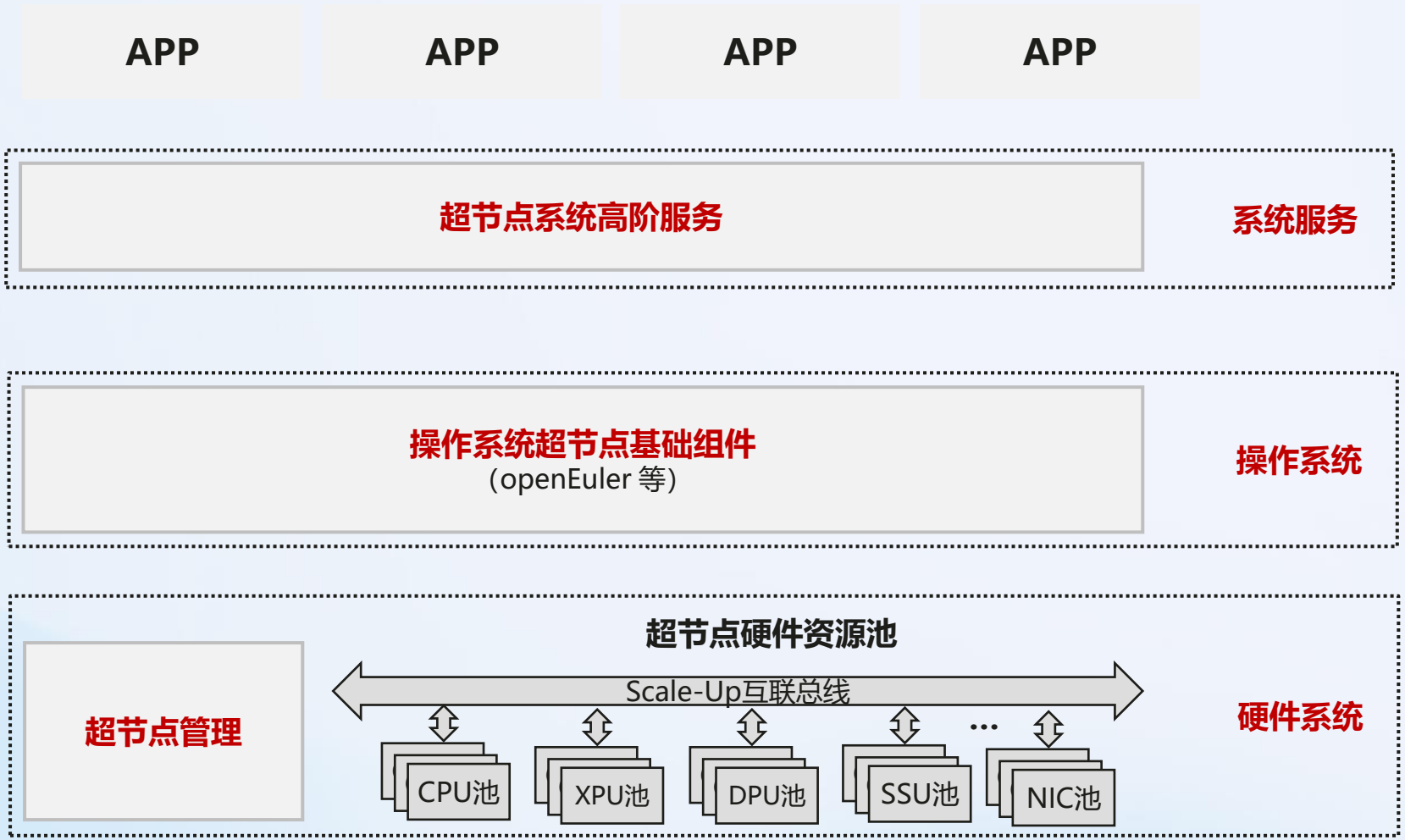


核心挑战

- **如何精准匹配“复杂应用的多样算力需求”和“复杂超节点的多样算力供给”？**
  - 应用高性能，依赖“计算/内存/带宽/时延”的调度，匹配负载需要
  - 池化高资源利用率，依赖“应用弹性扩缩容、统一调度，实现硬件资源高效时空复用”
- **如何保障超节点应用的高可用?**
  - 故障半径变大：规模大，硬件复杂，故障半径从单节点到超节点，需要软件层面实现高容错、高可用
- **如何简化超节点应用开发运维?**
  - 如何降低异构多样硬件使用门槛方便应用接入
  - 如何简化分布式应用开发部署运维
- **如何使业务“零”修改，平滑迁移到超节点?**
  - 业务面的网络、存储以及内存分配，池化后如何保持接口兼容并发挥性能优势
  - 复杂的管理面软件，需要抽象形成公共软件，避免管理面烟囱式生长

超节点基础软件

超节点系统软件架构



①高阶服务层：在业务框架层针对超节点架构调整优化，面向业务提供新的抽象和高阶API，**充分释放架构优势**

②操作系统层：操作系统超节点组件，实现操作系统的基础架构使能，**支撑超节点生态易用**

③硬件系统层：超节点管理，实现互连配置及设备资源管理，是实现异构对等和资源池化的基础。

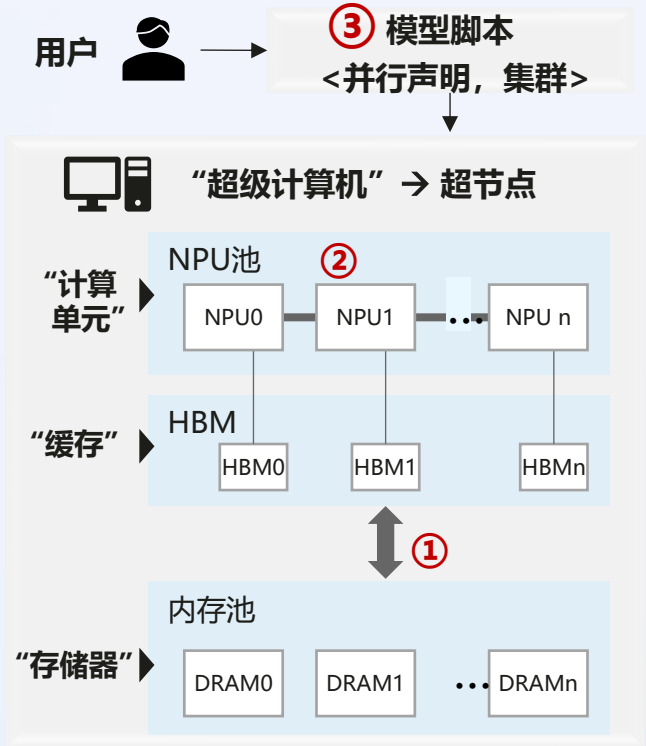
高阶服务：超节点亲和的 AI 框架设计思路

把超节点看成是一台“超级计算机”，而不仅仅是互联更快的集群

硬件变化点

- 对等架构
- 资源池化
- 网络拓扑分层多样

超节点亲和架构



① AI模型：计算与状态耦合→分离

Statefull→Stateless，
HBM作为缓存、池化内存作为存储

② 并行：SPMD→MPMD

齐步走变成异步走，集合通信走向单边通信

③编程：命令式→声明式

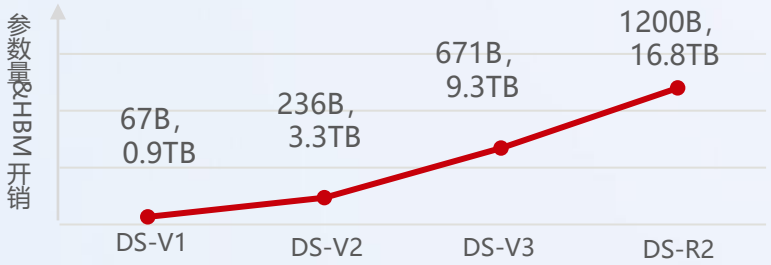
算法与并行策略解耦

计算/状态管理耦合 → 分离
HBM变为缓存，内存池为存储，提供更好的性价比和弹性

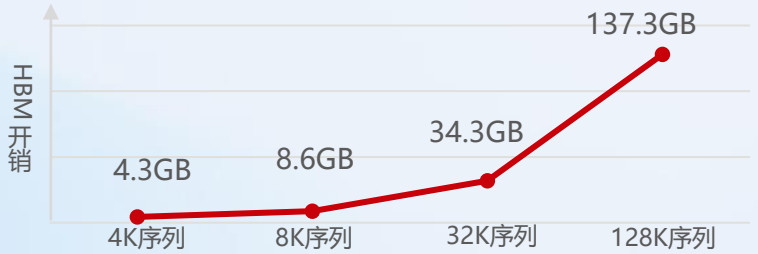
关键挑战

- **模型权重和序列长度快速增长，HBM容量成为瓶颈**

1. 1200B的MoE稀疏模型，训练时权重和状态HBM开销达到16.8TB



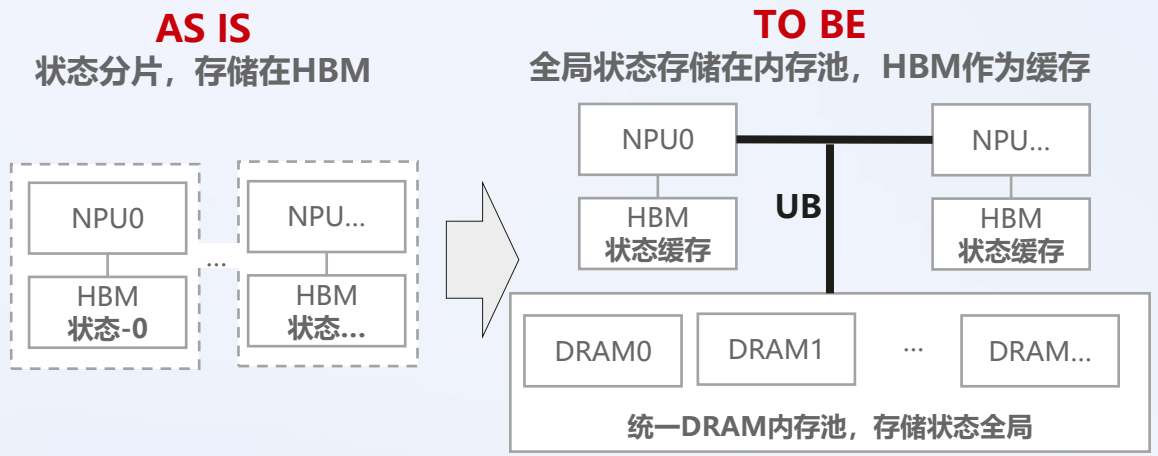
2. 推理时，KV Cache的HBM开销大小随序列长度线性增长



测试条件：DeepSeekV3，671B；BF16，BS=64，DP4EP4TP4

解决思路

- **思路：**计算与状态分离：基于昇腾内存池化架构，将HBM作为状态缓存，权重、KVCache等状态全局存储在统一的DRAM内存池中
- **挑战：**远端存储带来的时延问题
- **关键技术：**计算图全局编排，通过预取隐藏传输时延



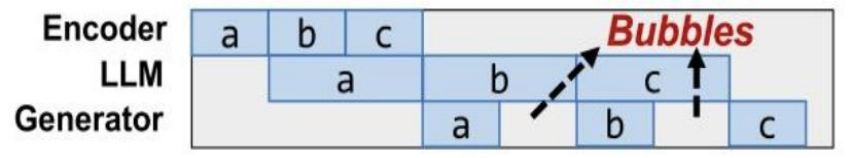
- **预期效果：**更好的性价比(相同的HBM放下更大的模型)；更好的可靠性和弹性，秒级故障恢复和伸缩；更简单的并行策略

并行–SPMD → MPMD
卡间并行与卡内并行结合，发挥超节点对等架构优势

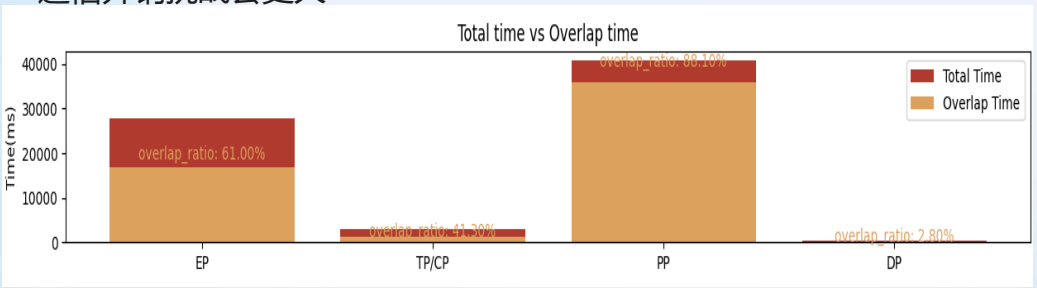
关键挑战

- **多模态、MoE、强化学习等场景，模型结构多样化，当前SPMD模式（每张卡同步执行相同任务），易出现负载不均、资源空闲**

- 多模态的不同子模型负载不同，SPMD模式叠加流水并行易出现流水线空泡

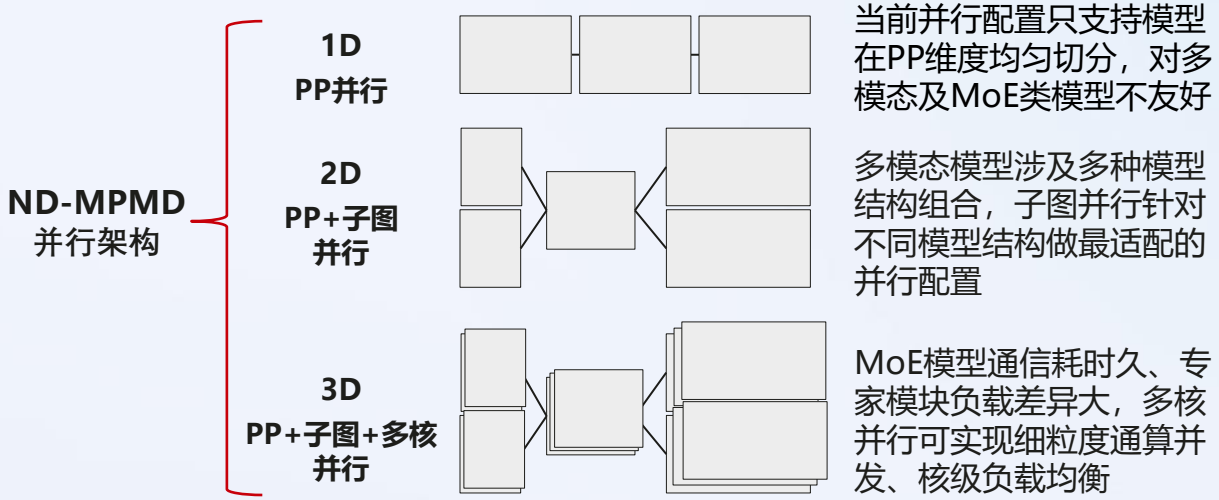


- MoE类模型通信执行耗时长，当前通信与计算掩盖比例不足
  DeepSeekV3 EP通信占比17%，掩盖比例61%，理想掩盖比例90%，未掩盖通信执行期间Cube核算力浪费；未来随着专家数的增加，EP通信开销挑战会更大



解决思路

- **思路**：细粒度的并行+细粒度的通信
- **关键技术**：ND-MPMD+基于内存语义的单边通信
  - **ND-MPMD**：基于对等算力池进行设备分组配置，实现卡间、卡内多核间多任务并发，提升集群算力利用率



  - **内存语义的单边通信**：细粒度单边通信消除非必要同步，内存语义减少host-device交互，减少断流
- **预期结果**：集群利用率提升15%

编程–命令式并行 → 声明式并行
面向超节点多样分层网络拓扑，实现新算法快速适配

关键挑战

- **新算法新硬件适配周期长**（1~2周）

1. 大模型算法在快速发展，需要新的并行策略优化组合

模型和算法	并行策略
稠密Transformer	DP、PP、TP、SP
稀疏MoE	DP、PP、TP、SP、EP
Diffusion	DP、FSDP
长序列	SP、CP
强化学习	MPMD
…	…

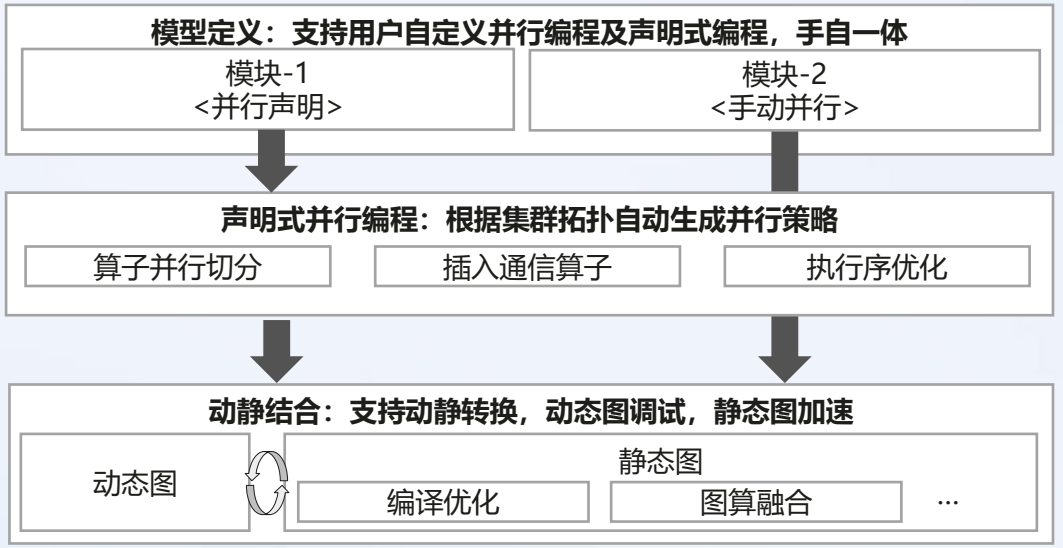
2. 集群架构演进，并行策略需持续配套

集群架构	并行策略
单机8 DIE	TP8，其他依靠PP切分
单机16 DIE	高维TP(TP16)，减少PP
8k节点超平面互联	拓扑感知高维TP(TP16)，减少PP

解决思路

关键技术：

- 算法和并行策略解耦，用户仅需声明并行需求，框架根据用户声明和硬件集群拓扑，生成昇腾亲和的并行策略
- 支持动静转换，开发态选择动态图调试，生产态使用JIT加速



预期效果：算法快速迁移，灵活组合；并行策略调优周期天级~小时级

函数分布式编程：单机编程体验，为开发者屏蔽分布式、硬件细节

方式1：原生函数编程

```
public JsonObject process(String request, Context
context) {
   // 省略业务逻辑若干行...
   // 库存管理函数调用另一出库函数“outbound”
    Function func = new Function(context, “outbound”);
   // 隐藏分布式调度、注册发现、弹性、LB、RPC通信
    ObjectRef<JsonObject> ref = func.invoke(request);
  JsonObject resp = ref.get(); // 获取调用的结果

  return resp;
}
```

方式2：少量修改自动分布式并行

（无状态）

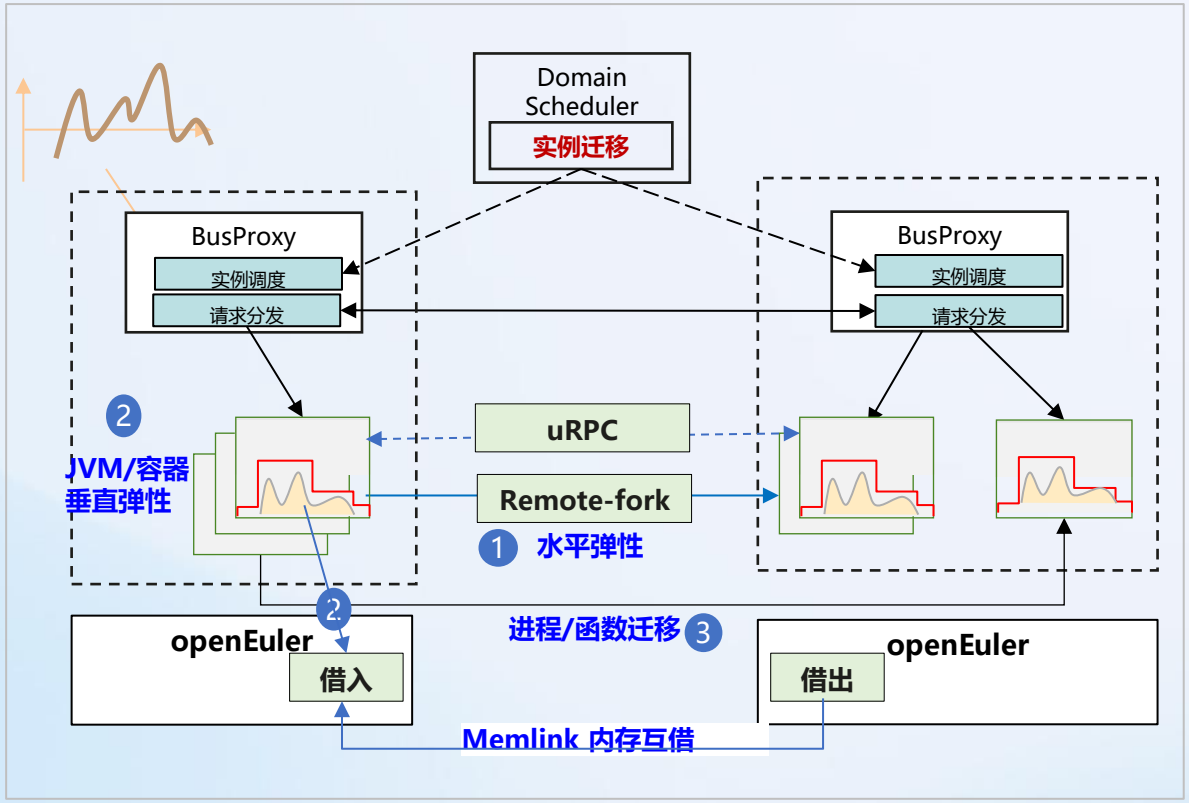
```
@yuanrong.invoke
def mc_simulation(spot):
    # 省略若干行单次蒙特卡洛模拟逻辑
    return ...

if __name__ == "__main__":
    n = 100
    s = ...

     # invoke快速返回异步future
     # for循环隐藏n次分布式并行函数调度执行
     futures = [mc_simulation.invoke(s) for i
in range(0, n)]
     # 同步阻塞获取n个并行处理结果
     outputs = yuanrong.get(futures)
```

（有状态）

```
@yr.instance
class Bill():
    def __init__(self, customerName)
        self.customerName = customerName
        self.totalMoney = 0

    def add(self, money):
        self.totalMoney = self.totalMoney + mone
y
        return self.totalMoney

    def get(self):
        return self.totalMoney

# 调用"记账"应用
instance = Bill.invoke("Jimmy")
instance.add.invoke(1234)
ret = instance.get.invoke();
# 打印出1234
print(yr.get(ret))
```

函数系统：构建灵衢亲和的实例弹性和数据共享技术，实现极致性能和高资源利用率

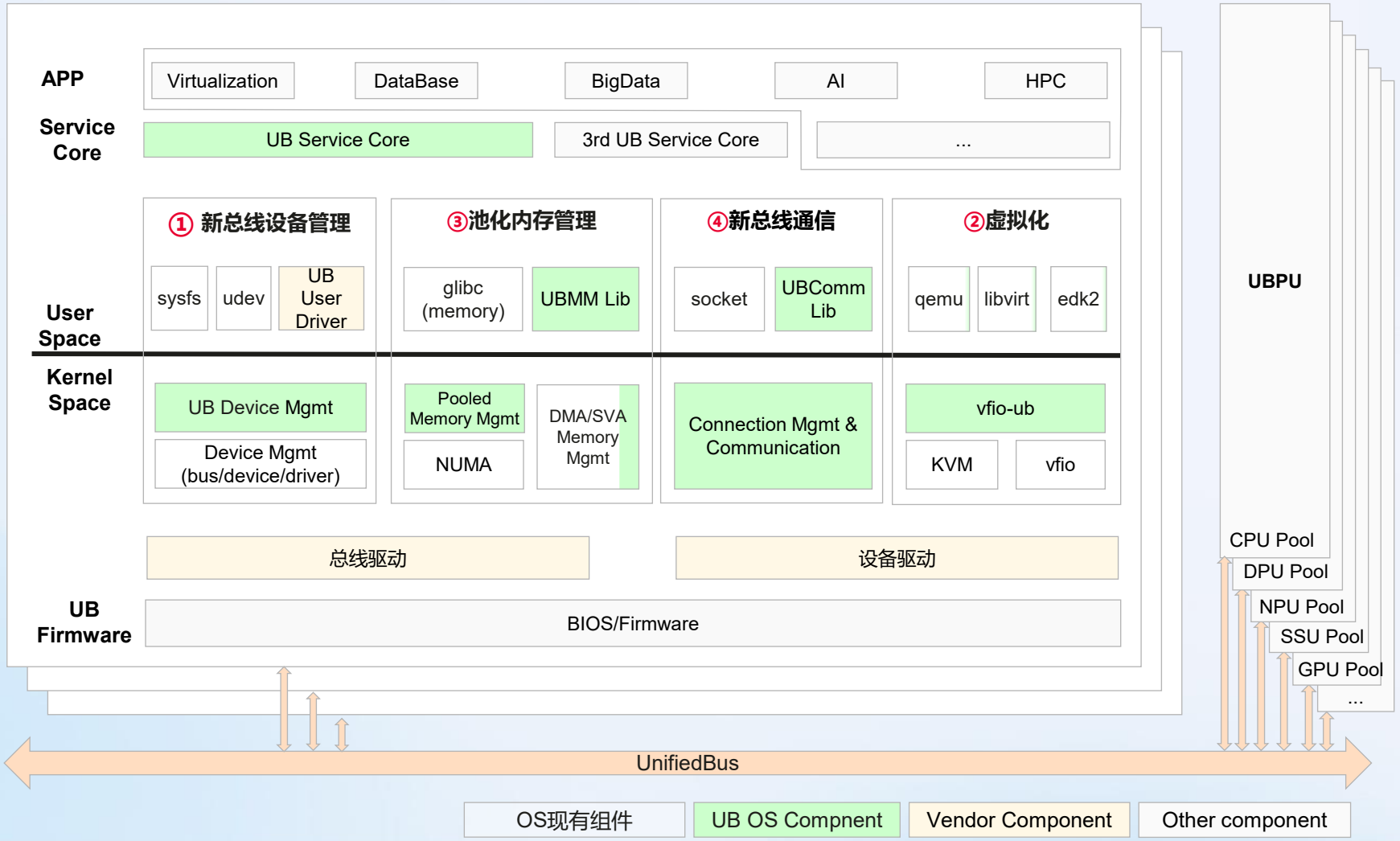
- **实现应用的高效分布式运行**：当前实例弹性、迁移等基于传统网络构建，时间达到秒级，无法实现超节点亲和的分布式算力的极致弹性和业务无感
- **实现超节点内实例的高密部署**：单实例基于最大资源预留配置，无法按需分配资源，难以实现超节点内的高密部署，同时不影响业务运行。



关键技术

- **超节点快速实例复制**：程序运行中可动态创建新函数实例，调度器根据集群实际资源使用视图选取合适节点，基于**灵衢高带宽**和 **Remote-fork**等能力，函数实例（百M镜像大小）水平扩容速度**从3秒提升至300ms**
- **内存垂直弹性**：按实例实际运行情况动态调整实例内存资源配额（如容器/JVM大小），在保证业务正常运行前提下，释放多余资源支持节点上调度更多实例。当**节点内存资源不足时通过灵衢Memorylink实现跨节点内存互借**，支持实例内存垂直扩容，避免超出单节点物理内存导致OOM
- **有状态实例自动迁移**：监测各节点资源使用情况，当节点资源不足时，可将部分有状态实例的请求保持住，**通过灵衢网络快速将状态跨节点迁移并保持实例ID不变**，完成后恢复请求分发，实现业务无感

操作系统：异构硬件统一抽象解耦、统一内存地址空间，支持资源全局微秒级调度、计算资源动态组合扩展、设备间高性能通信



操作系统灵衢组件：在OS在原有内存管理、通信、设备管理和虚拟化框架上扩展支持灵衢，扩展的4个功能分别是：

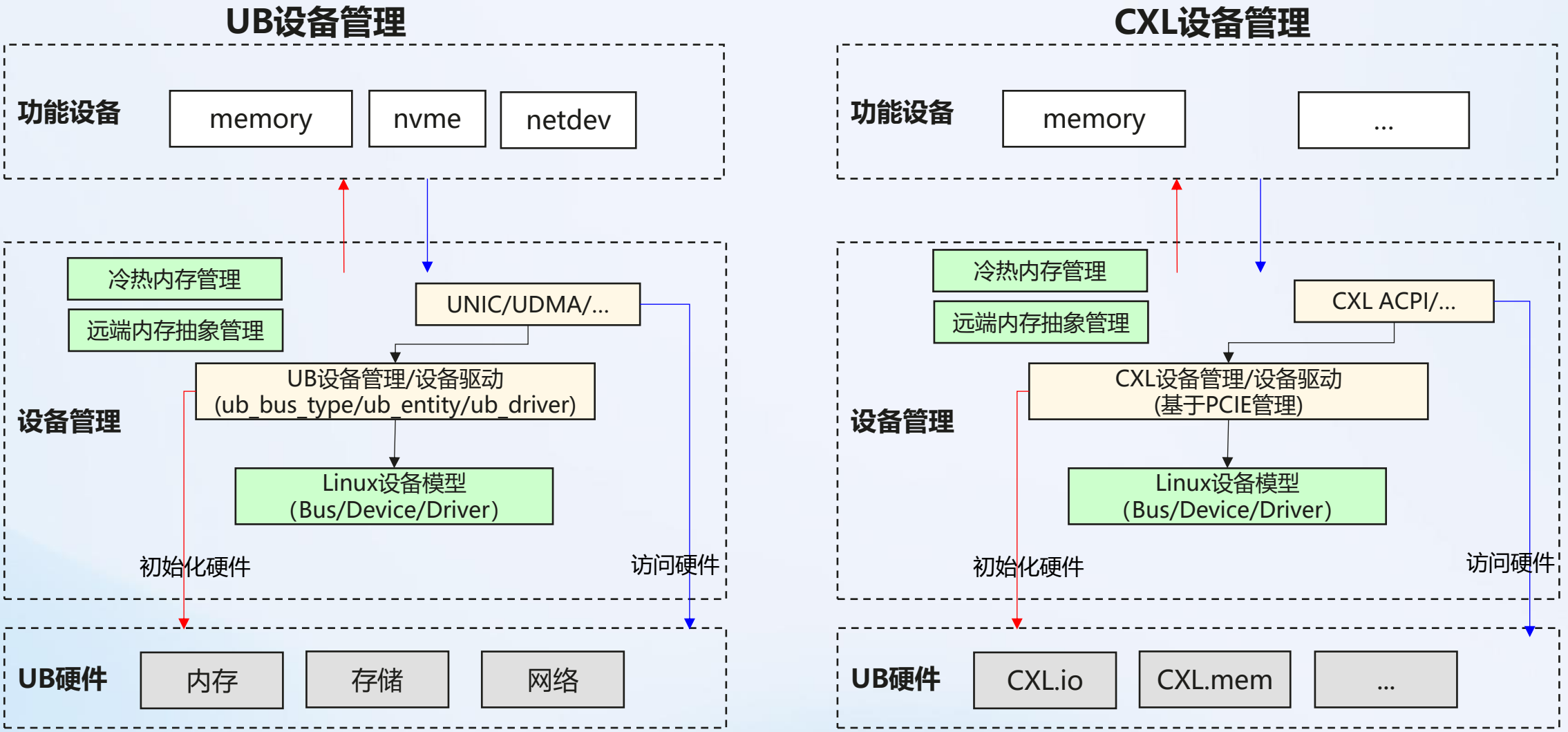
① **新总线设备管理**：基于bus/device/driver设备管理模型，扩展提供UB总线、UB设备管理能力，实现计算节点内UB设备热插拔、配置。

② **虚拟化**：基于KVM、vfio框架，扩展提供UB设备直通虚机能力。

③ **池化内存管理**：基于OS已有NUMA、DMA/SVA内存管理框架，扩展提供UB总线域内内存语义访问能力，实现跨计算节点跨设备内存借用、共享。

④ **通信**：提供异步通信能力，实现跨计算节点、跨设备通信和远程调用。

设备管理：多种总线和设备，需要软件统一抽象和管理，并兼容现有Linux生态

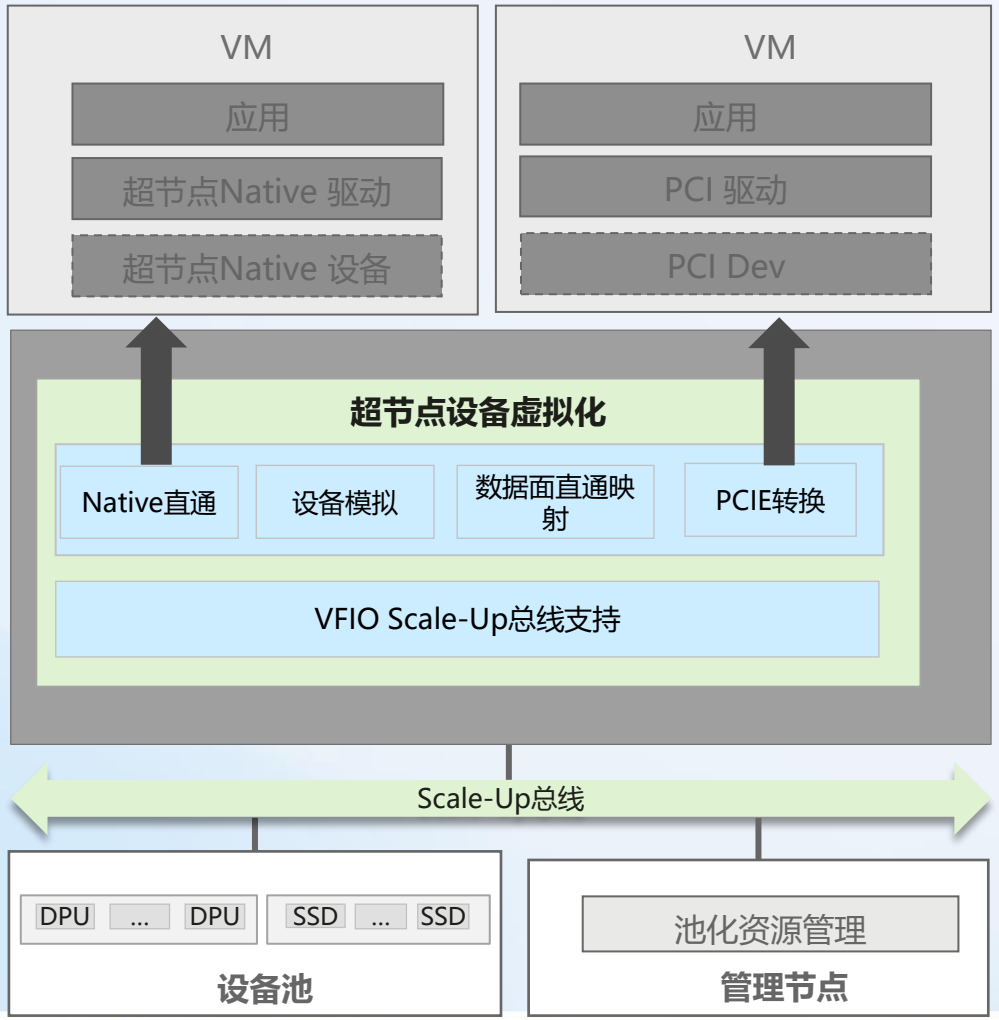


多总线的设备、远端内存等，需要有统一抽象和管理，避免多烟囱式发展

公共组件

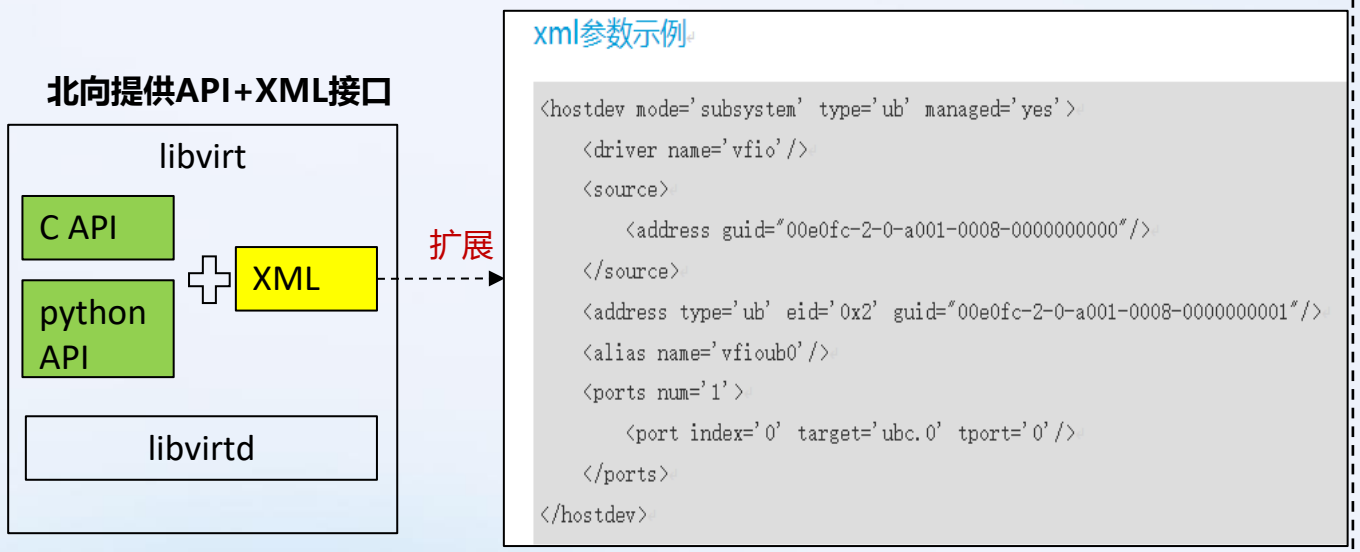
总线相关实现

虚拟化：转成PCIE设备直通虚机，虚机OS二进制兼容，无感获取高带宽低时延收益

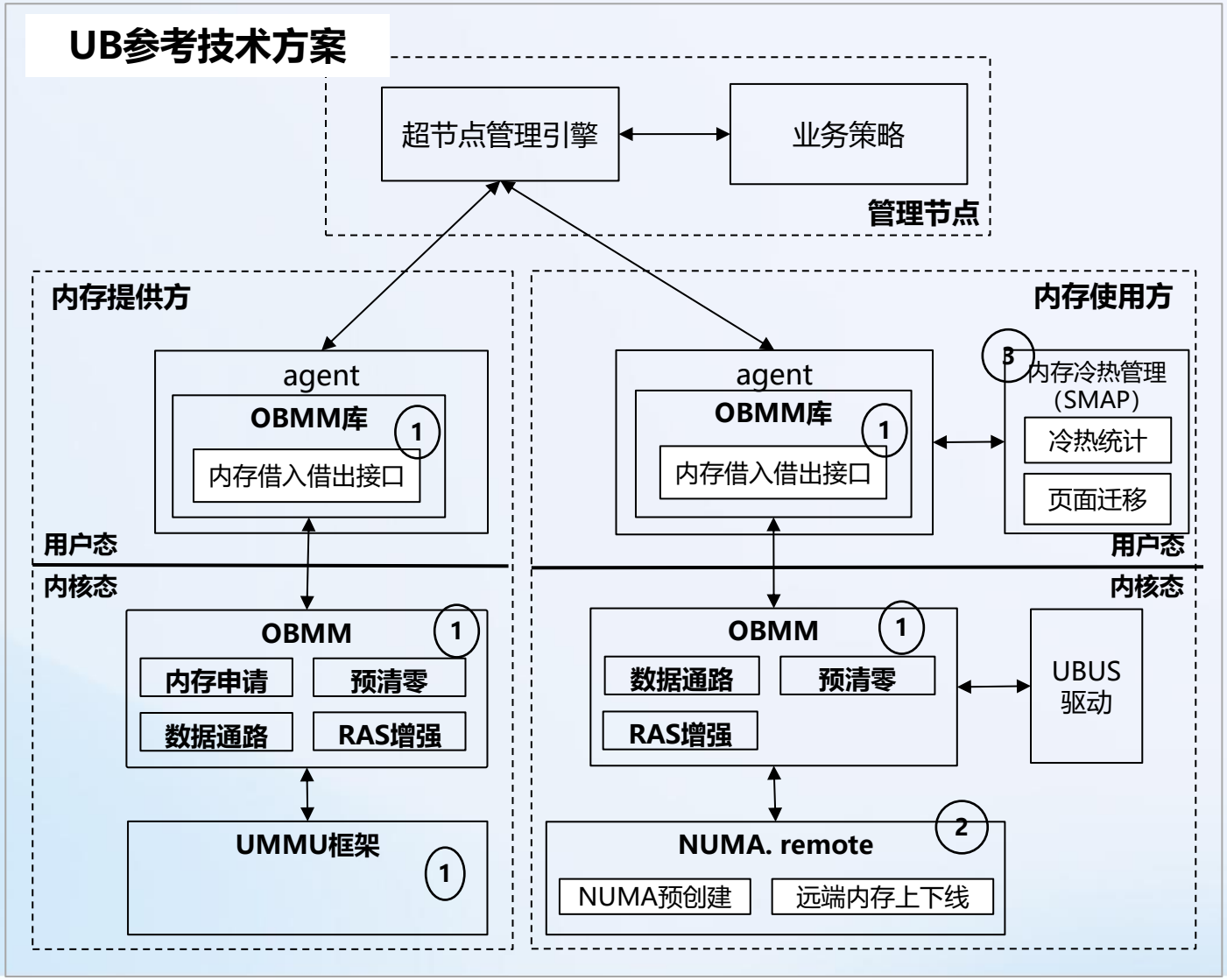


1. 虚机支持Scale-Up总线模拟，实现数据面超节点设备的直通，虚机使用远端设备池按需弹性扩展；
2. 超节点设备转成PCIE设备直通虚机，Guest OS二进制兼容不修改，并获取性能收益

UB直通案例：新增xml接口描述UB设备



内存管理：实现内存池化访问和共享，超节点内内存流动提升利用率



① **OBMM**：支撑内存快速拆借和超节点内存共享。

② **NUMA.remote**： 远端内存公共抽象管理，通过numa节点上下线。

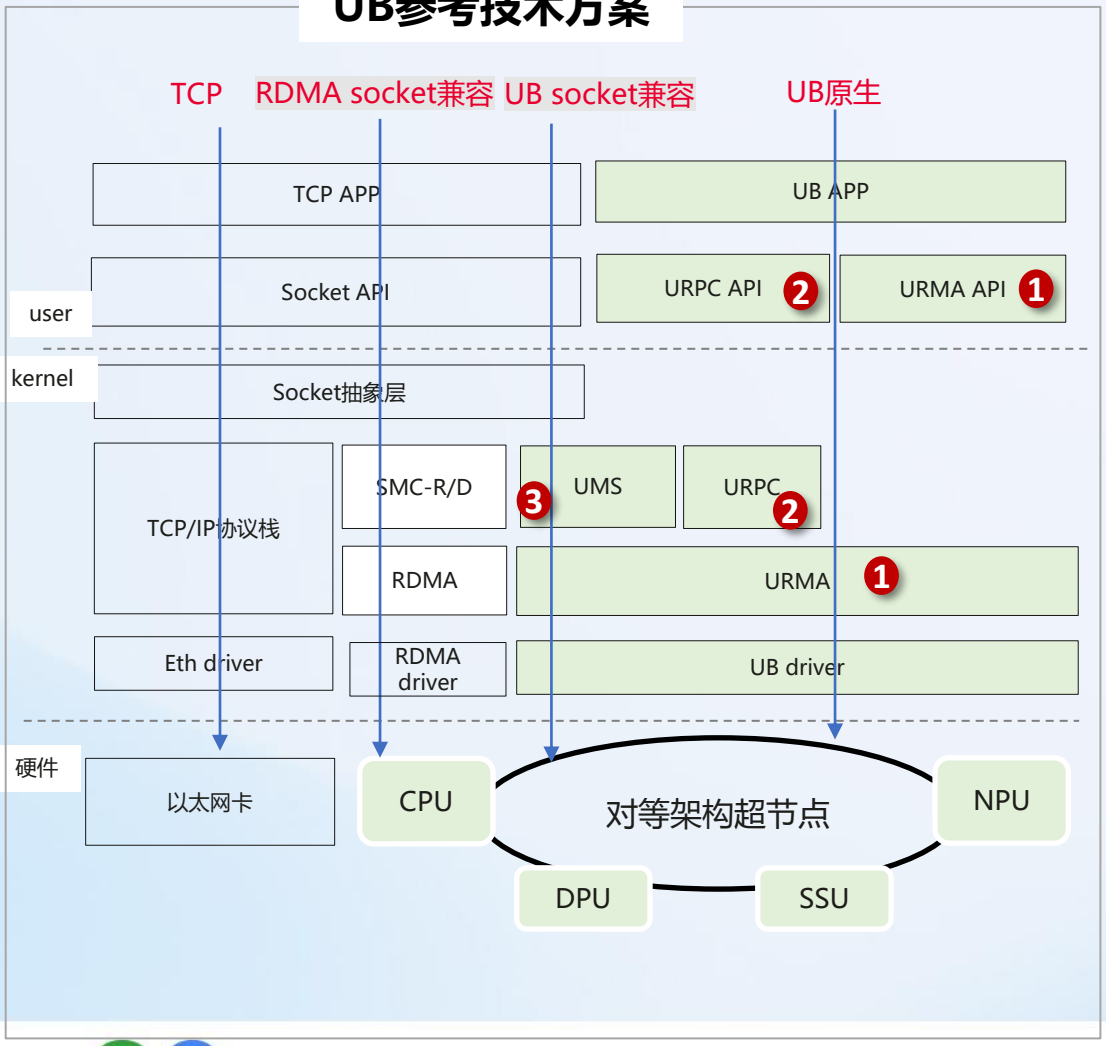
③ **内存冷热管理**：通过内存冷热管理，冷页放置在远端内存

内存拆借能力：单节点可支持借用3T+内存，提升内存利用率。

内存共享能力：支持超节点内基于内存语义的共享，降低多机之间内存拷贝，提高性能。

超节点通信：提供socket兼容、Scale-Up网络原生支持等多种高性能方案

UB参考技术方案



关键技术

1 Socket兼容方式

方式一：应用进程替换so，应用零代码修改

```
export LD_PRELOAD=/usr/lib/libubsocket-preload.so
exec $base_dir/kafka-run-class.sh $EXTRA_ARGS kafka.Kafka "$@"
```

方式二：应用修改一行代码

```
sockfd = socket(AF_INET, SOCK_STREAM, 0);
```

↓

```
sockfd = socket(AF_UB, SOCK_STREAM, 0);
```

2 URPC远程过程调用接口：

- client发起远程函数调用：

urpc_func_call(...)

- Server返回调用结果：

urpc_func_return(...)

3 URMA异步编程接口：

- 报文发送接口：

urma_post_jetty_send_wr(...)

- 下发接收任务接口：

urma_post_jetty_recv_wr(...)

- 获取发送和接收结果接口：

urma_poll_jfc(...)

业务价值

1、内存语义提供1us级低时延通信；

2、异步通信语义实现TB级超大带宽通信；

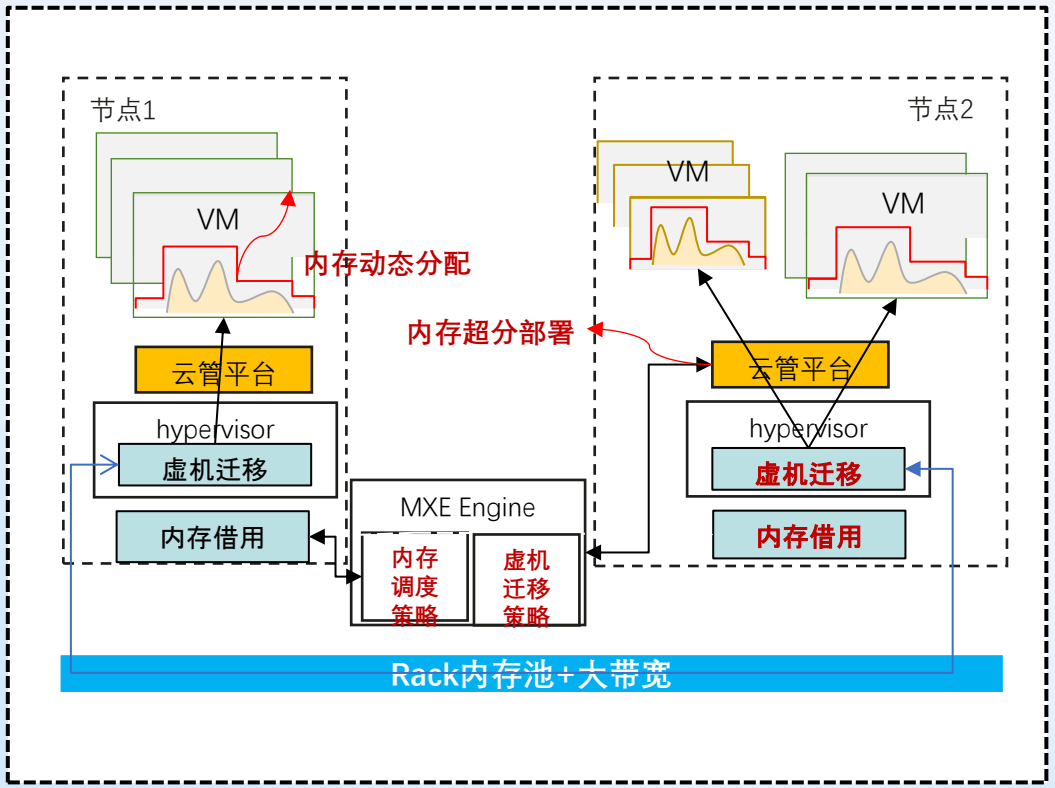
3、应用可基于兼容接口零修改快速上线，也可以基于UB原生接口实现极致性能；

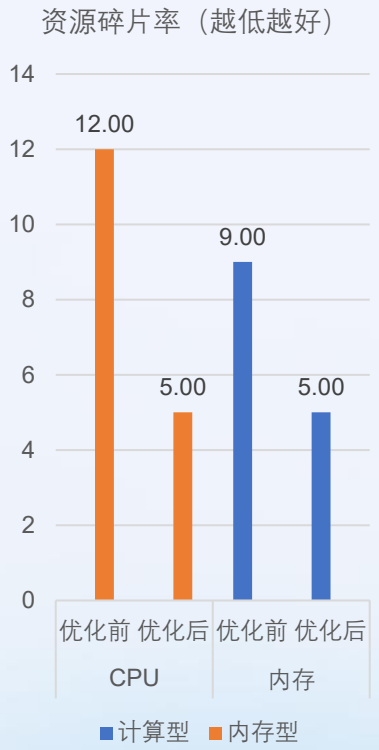
一些案例分享

虚拟化：内存借用和虚机热迁移逃生通道，内存超分25%，利用率提升20%+

- 发挥超节点低时延、语义和池化特性，IaaS层达成内存流式利用，提升基础资源利用率，内存超分25%场景，性能下降<5%；
- 基于超节点低时延内存语义+资源动态调度，支持超大规格虚机（1K+vCPU/10T+内存规格），满足云上大机诉求。

基于Rack内存池和大带宽，构建动态资源调度能力，实现高可靠的内存超分方案



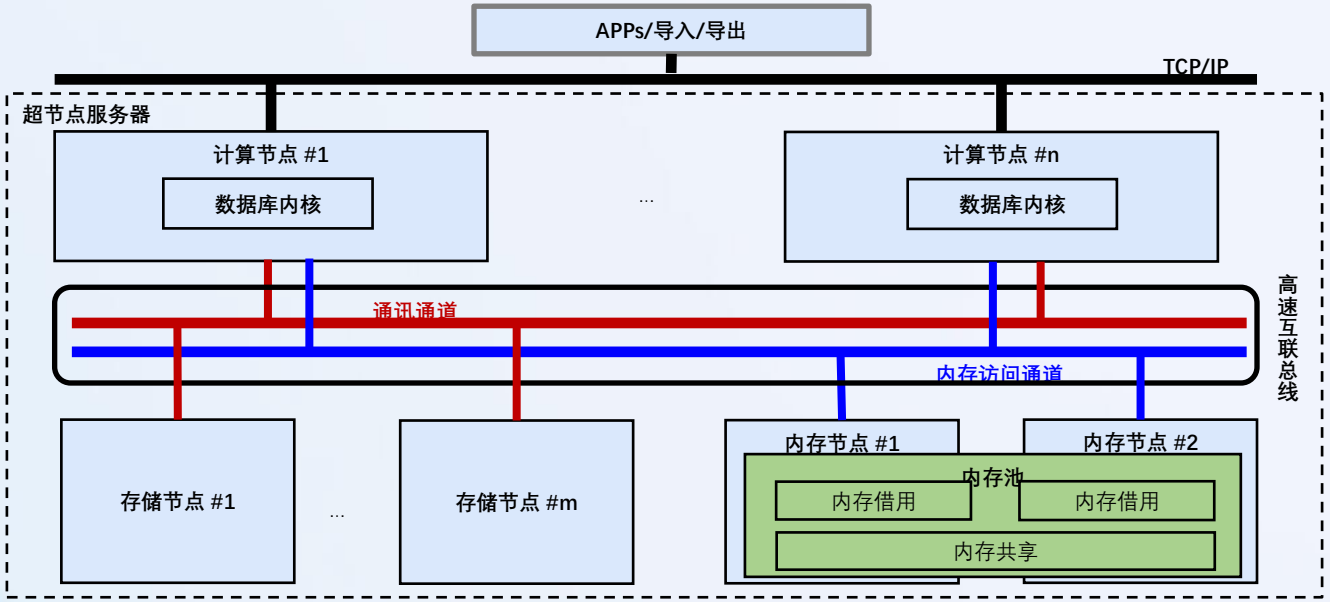


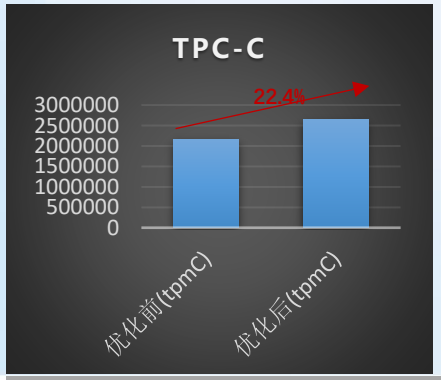
业务价值

功能	关键能力指标
极速批量迁移、升级	➢ 虚机热迁移效率：典型集群200台：月->周/天 ➢ 无损扩容：TB级大带宽传输，虚机热迁移中断<50ms，业务无感
超大规格虚机，使能“云上大机”	➢ 支持超大vCPU规格典型规格800C/1920C ➢ 支持超大规格内存典型规格3T/6T
极速热迁移，提升售卖率	➢ 降计算资源碎片，提升售卖率vCPU碎片率12%->5% 内存碎片率9%->5%

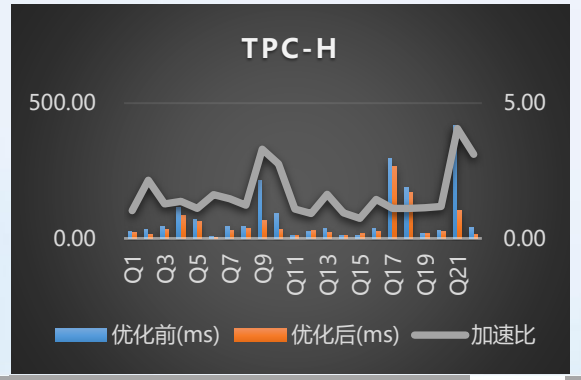
数据库：为数据库构筑低时延通讯和池化内存底座，TP提升20%，AP提升50%，RTO < 6s

- 超节点内通讯：加速数据库集群模块间通讯，及分布式系统全局资源分配
- 内存借用/内存共享：将ms级网络/存储IO操作转换成ns级内存语义操作，加速SQL算子执行、数据访问、及故障恢复

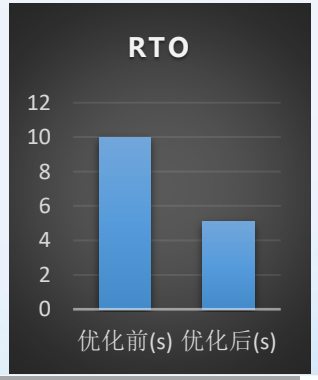




◆ 2节点TP, TPC-C，215.9WtpmC ->264.3WtpmC，提升22.4%



◆ 2节点AP, TPC-H，Q22总时间 1806.2s ->1091.9s，加速比1.65



◆ 90WtpmC，RTO 5.1s

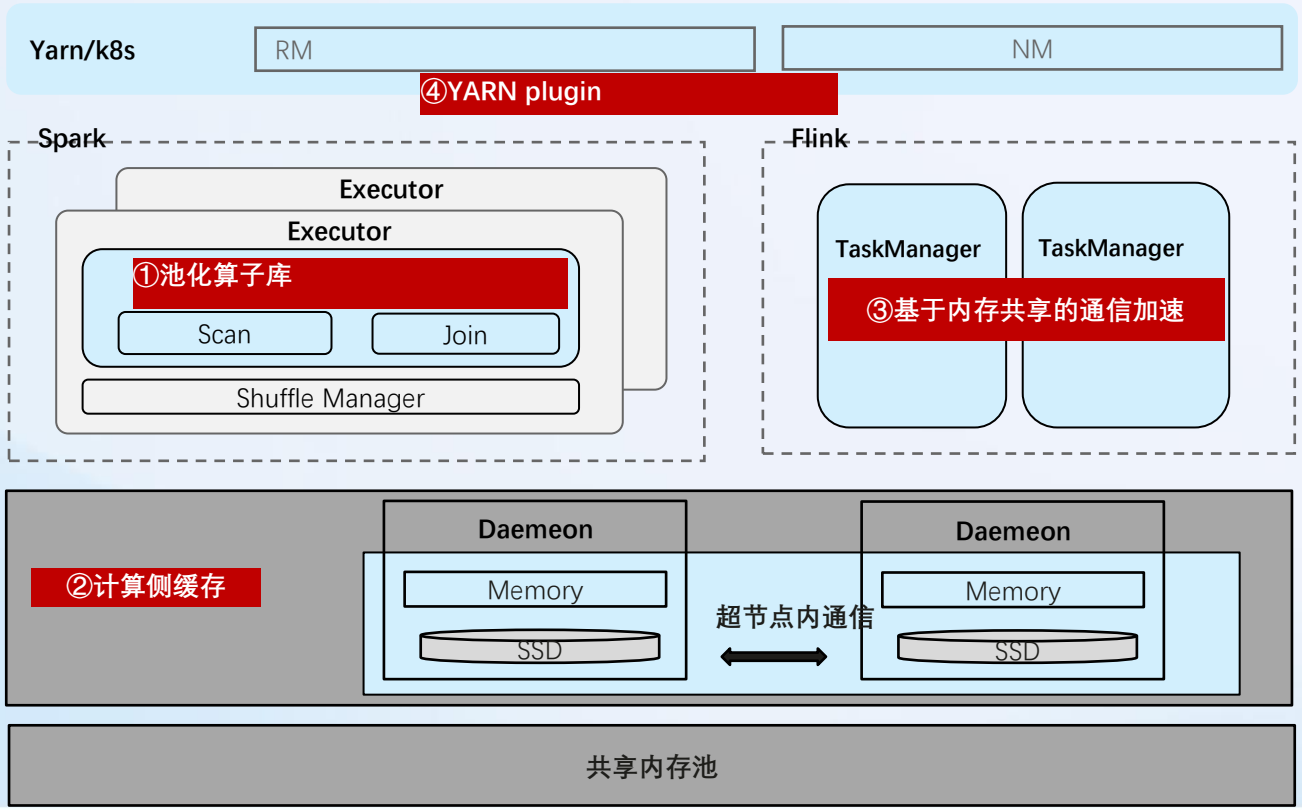
业务价值

模块	竞争力手段
OLTP	➢ 集群模块之间采用超节点内低时延/高带宽通讯 ➢ 采用超节点内通讯原子操作提升分布式系统**全局资源分配**性能，提高线性扩展比 ➢ **多级缓存池管理，动态扩缩。**热数据存入本地内存，温数据存入远端内存池，基于**访问热度动态迁移。**
OLAP	➢ 基于负载的**弹性内存扩展**，加速SQL算子(排序、聚合、JOIN类）处理，降低数据溢盘 ➢ **解耦内存层，共享数据**，加速复杂SQL算子shuffling处理，降低数据重分布
RTO	➢ 缓存脏页，**加速故障恢复**，降低RTO ➢ 集群异常主动通知，数据库**异常检测时间**降为毫秒级

大数据：基于内存语义和高性能通讯，批数据分析时长缩短20%，实时数据处理性能提升25%

- Spark：基于内存借用和内存共享的池化算子库提升算子效率；基于超节点内通信的shuffle加速和计算侧缓存提升数据通信效率。
- Flink：基于内存共享的通信加速，减少内存拷贝和免序列化；基于超节点内通信的计算侧缓存，加速checkpoint读写速度。

池化内存与高性能通信，削峰填谷，减少数据落盘，提升性能与利用率



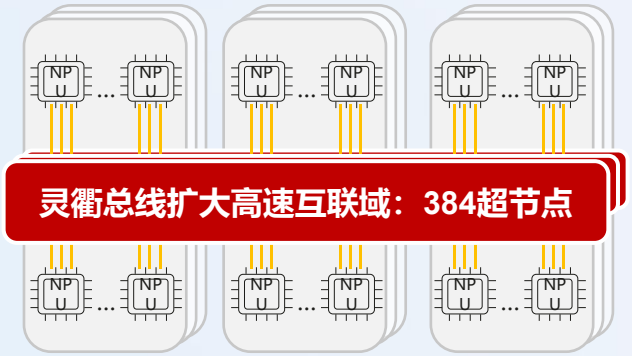
业务价值

功能	关键能力指标
高性能	◘ 算子性能：内存借用减少数据落盘，提升算子性能 ◘ 高效数据传输：内存共享实现UB数据交换，减少序列化和降低传输时延 ◘ 高时效性：flink性能提升，助力搜推场景模型更新从小时级到分钟级
高可靠	◘ 快速恢复：基于计算侧缓存的checkpoint，实现任务快速恢复 ◘ 削峰：基于内存借用，减小任务峰值导致的OOM概率

大模型训练：超节点以架构创新，实现有效算力对裸算力超线性增长25.6%

- 基于TB级大带宽、低时延内存直访，实现超节点架构创新，384卡间高速互联带宽无收敛，显著扩大高速互联域
- D2D跨级带宽提升15倍，D2D一跳转发时延降70%，支撑训练性能3.14倍提升

超节点架构



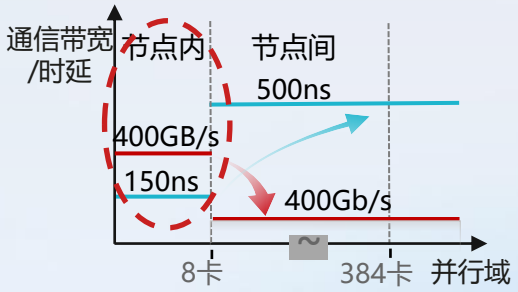
并行模式	通信不可掩盖性程度
数据并行 DP	低，可大部分掩盖
张量并行 TP	高，通常不可掩盖
流水并行 PP	中，可部分掩盖
序列并行 CP	高，通常不可掩盖
专家并行 EP	高，通常不可掩盖

高速互联总线扩大高速互联域，跨节点通信性能显著提升

服务器集群 未掩盖通信37%

DeepSeek V3预训练 计算通信占比
并行：PP8/DP64/EP64

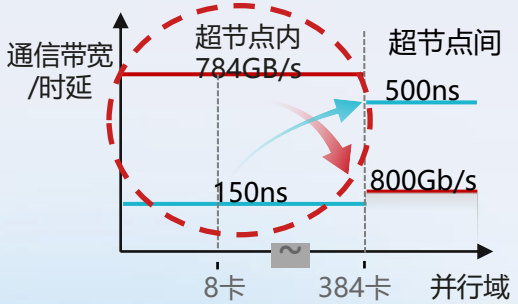
跨机(8卡)后通信带宽劣化87%



超节点集群 未掩盖通信16%

DeepSeek V3训练 计算通信占比
并行：PP8/DP64/EP64

跨机(8-384卡) 通信带宽不劣化



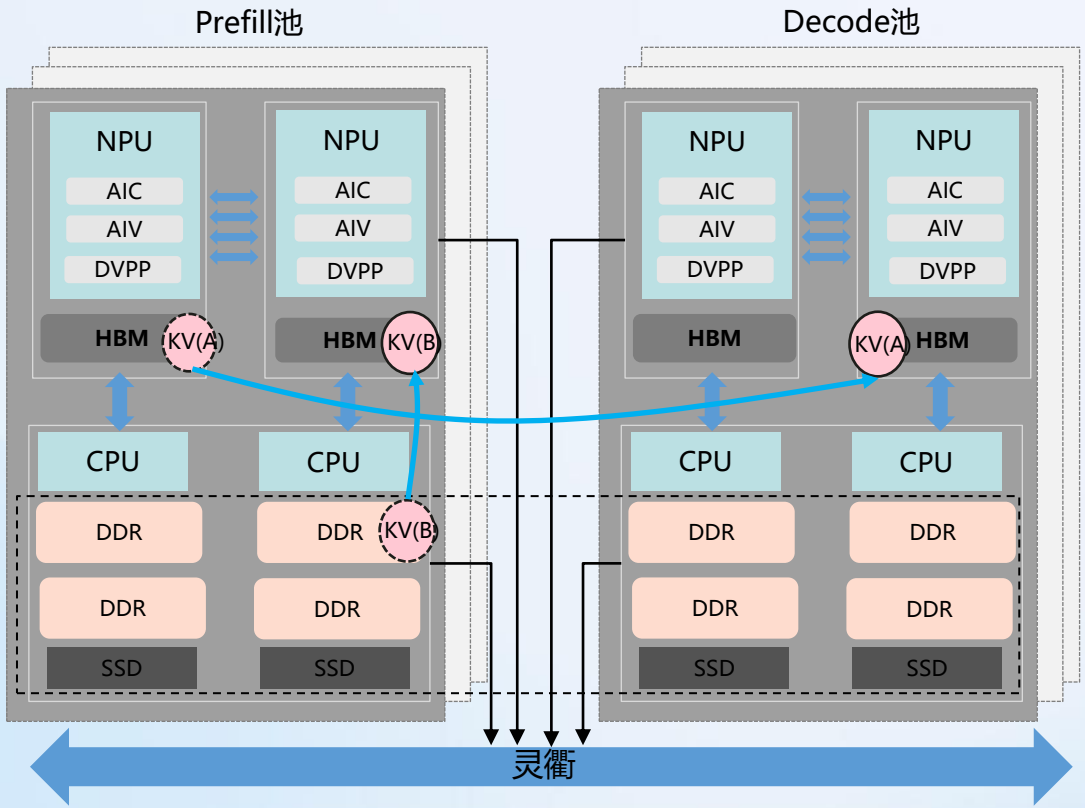
业务价值

功能	关键能力指标
集合通信性能	➢ D2D带宽(跨机双向) 400Gbps → 784GBps，提升**15倍↑** ➢ D2D 一跳转发时延500ns → 150ns，**降70%↓**
训练性能	➢ 2.5倍裸算力，**3.14倍训练性能**，超线性增长**25.6%**

大模型推理: 基于对等互连架构，实现多层级KV池数据直通，推理性能提升10%+

- 基于灵衢总线的全对等架构构建多层级KV池，提升以查代算的KV缓存命中率，首Token时延TTFT降低40%+。
- 基于内存语义的同步和异步访存实现数据的低时延交换，跨实例KV直通传输免拷贝，数据传输时延降低30%+。

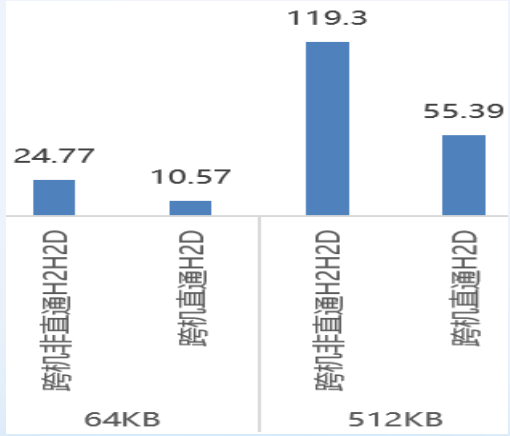
多层级KV Cache池扩展缓存空间，提升以查代算的KV缓存命中率，降低TTFT



基于内存池提升KV Cache复用性能收益（LLAMA3 70B，多轮对话数据集）

性能维度	优化
推理吞吐(tokens/s)	23.8%
首Token时延（ms）	45.5%

直通和非直通传输时延比较（us）

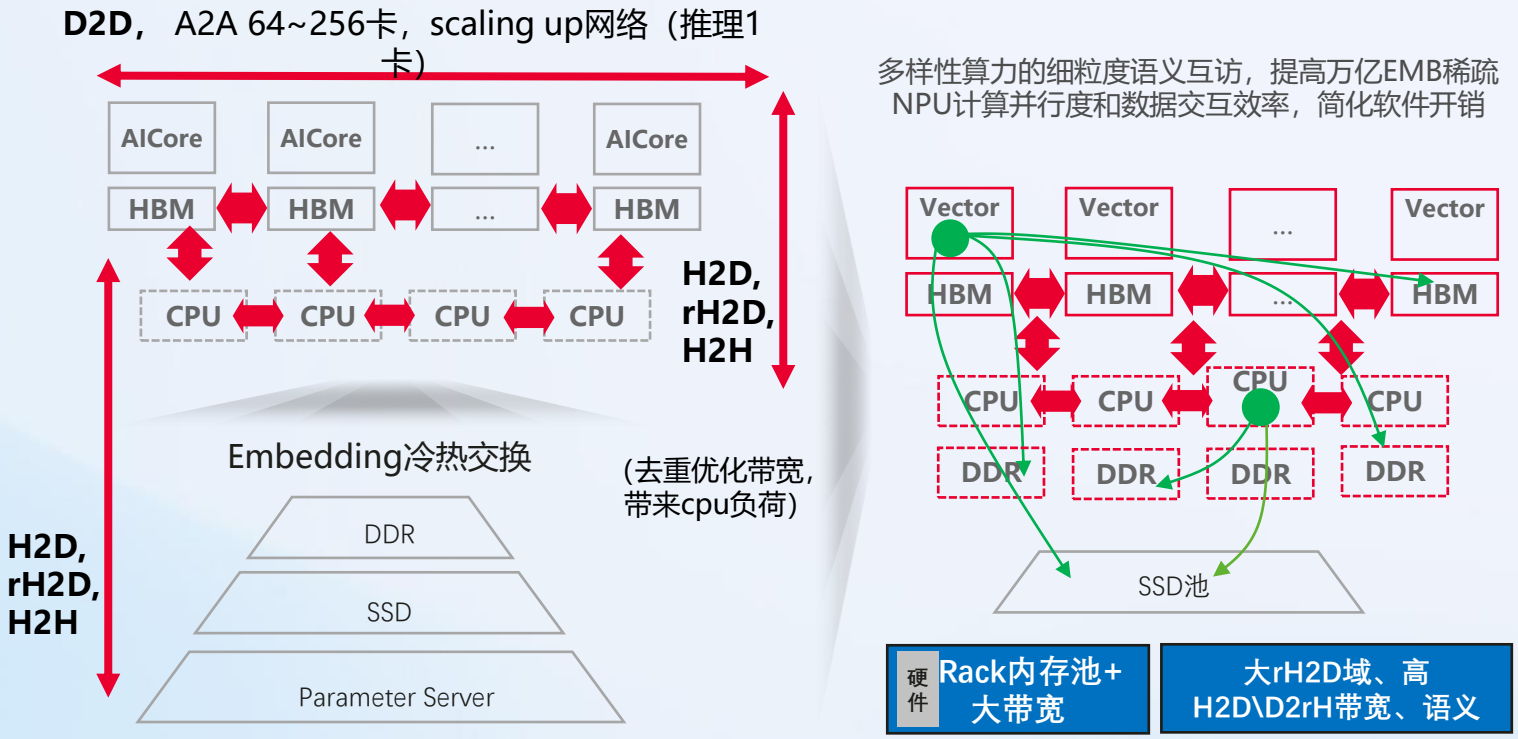


业务价值

功能	关键能力指标
多层级KV Cache池化系统	➢ 实例选择调度：感知缓存局部性，匹配KV Cache复用比例高的实例服务请求； ➢ 池化共享：纵向多层级介质扩容缓存空间，横向D-D/H-D直通加速传输，KV加载和卸载数据的性能提升10%+；

推荐：发挥H2D、rH2D高带宽、DDR池化以及低时延语义能力，提升万亿稀疏推荐模型性能30%↑， 对比PS架构大幅提升易用性

- **发挥B低时延、语义、高带宽特性，优化百卡上万亿稀疏EMB协同计算效率，优化万亿EMB训练A2A计算线性度；**
- **发挥低时延、语义和池化特性，优化推荐特征处理、万亿EMB查询缓存效率，提升推理全链路延迟性能，优化延迟约束下推理吞吐性能；**



业务价值

功能	关键能力指标
训练吞吐性能	➢ 训练吞吐：小集群提升20%↑，百卡集群场景提升40%↑；
推理延迟约束吞吐性能	➢ 特征缓存效率提升30%，特征计算和EMB交换效率提升30%，优化推理集群MFU提升20%；
运维易用性	➢ 简化编程模型，降低训推集群系统的运维成本

超节点的基础软件： openEuler实践与开源项目

	开源项目	描述
系统服务	openYuanrong	面向超节点的函数运行环境
	sysSentry	超节点系统故障巡检/故障隔离服务；配合系统级CR，异构任务通过迁移避免可用
	openEuler Copilot	超节点系统亚健康状态的检测和定位
核心子系统	FalconFS	超节点高性能分布式存储池
	XMFS	跨节点共享内存文件系统
	MemLink	超节点内存的弹性和高效复用
	xSched	CPU与XPU间负载均衡
	xMig	虚拟机热迁移迁移，内存和设备池化共享免迁移
	GMEM	基于 GMEM 的驱动程序可以无缝共享虚拟地址空间，完成 CPU 和设备之间的协同
	gVirt	XPU一虚多 与 多虚多
池化底座	SMC-UB	基于共享内存语义通信
	OBMM	内存池化基础能力，支持超节点内存借用
	UBVirt	对池化IO设备进行AA聚合、AP主备倒换，实现高带宽和高可靠

未来展望

极致性能发挥

- 应用高性能，依赖“计算/内存/带宽/时延”的调度，匹配负载需要
- 池化高资源利用率，依赖“应用弹性扩缩容、统一调度，实现硬件资源高效时空复用”

共建共享

- 超节点的硬件实现方式当前百花齐放，但是软件生态不应该割裂。共性问题在开源项目中解决。
- 复杂的管理面软件，需要抽象形成公共软件，避免管理面烟囱式生长

构建围绕超节点基础软件的开源社区

平滑过渡

- 存量业务容易迁移到新的超节点硬件
- 如何简化超节点应用开发运维?
- 如何降低异构多样硬件使用门槛方便应用接入

未来创新引领

- 超节点不是集群，需要新的技术视角来支持架构和实现的演进
- 如何保障超节点应用的高可用?

InfoQ 极客传媒

北京

QCon 全球软件开发大会

会议时间：4月10-12日

- 大模型赋能 AIOps
- 越挫越勇的大前端
- 云上业务架构演进
- 多模态大模型及应用
- 海外 AI 应用创新实践

北京

AiCon 全球人工智能开发与应用大会

会议时间：6月27-28日

- 端侧智能
- AI Agent
- 多模态大模型
- 金融+大模型

上海

QCon 全球软件开发大会

会议时间：10月23-25日

- AI Agent
- 大模型训练推理
- 端侧 AI
- 搜推深度融合
- 数智金融

4月 — 5月 — 6月 — 8月 — 10月 — 12月

上海

AiCon 全球人工智能开发与应用大会

会议时间：5月23-24日

- 通用大模型
- AI Agent
- 垂直领域应用
- 模型可解释性

深圳

AiCon 全球人工智能开发与应用大会

会议时间：8月22-23日

- 模型效率与部署
- 多模态大模型
- 大模型安全
- 智能硬件

北京

AiCon 全球人工智能开发与应用大会

会议时间：12月19-20日

- 通用大模型
- 智能硬件
- LMOps
- 具身智能

极客邦科技 2025 年会议规划

促进软件开发及相关领域知识与创新的传播

参会咨询

查看会议

QCon
全球软件开发大会

InfoQ 极客传媒

THANKS

超节点 重新定义 数据中心硬件
异构融合 重新定义 数据中心基础软件
AI 重新定义 数据中心业务