KubeCon CloudNativeCon OpenInfra SUMMIT › ASIA PyTorch CONFERENCE

China 2026

How Intsig Serves Billions of Document Scans: GPU Virtualization at Scale with HAMi

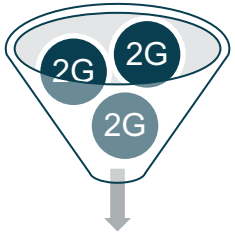
Mengxuan Li, Walter Duan

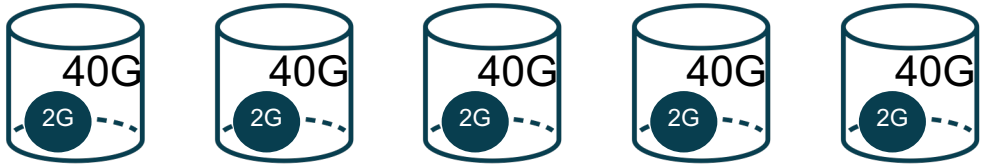
Catalog

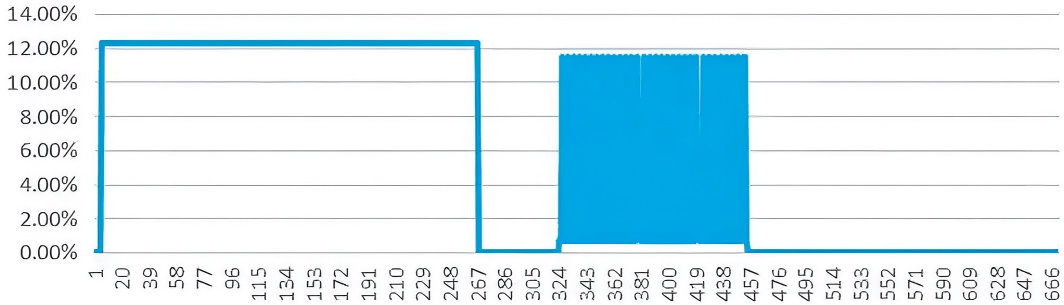
- Background
- HAMi Introduction
- Case-study

Background:
Inadequate Compute resources

Task queue status:
pending

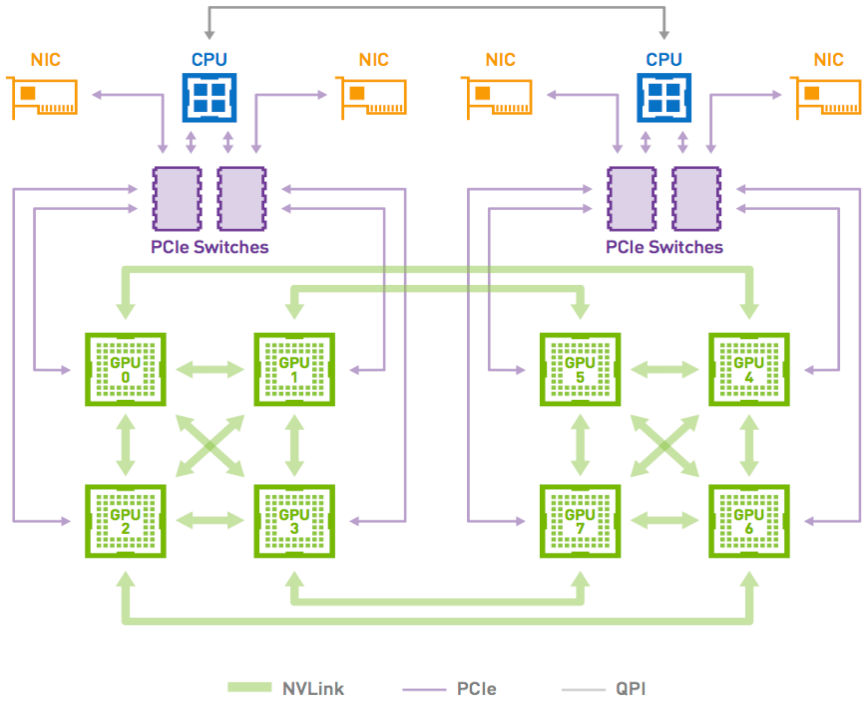






- Exclusive GPU Allocation
- Large number of small deployments

Background: Topology-aware scheduling is needed



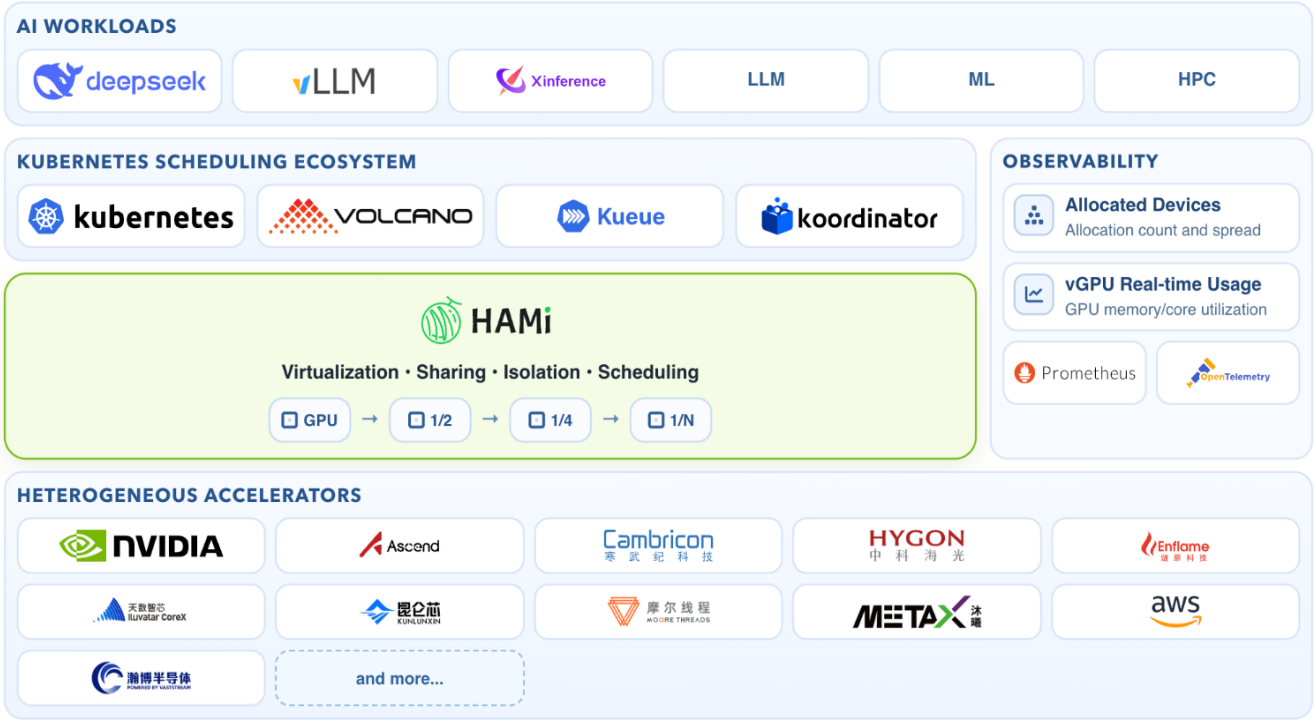
If we allocate 2 GPUs

(GPU0, GPU1) ✓
(GPU0, GPU6) ✗

NVLink (25GB/s – 1800GB/s) > Pcie (16GB/s)

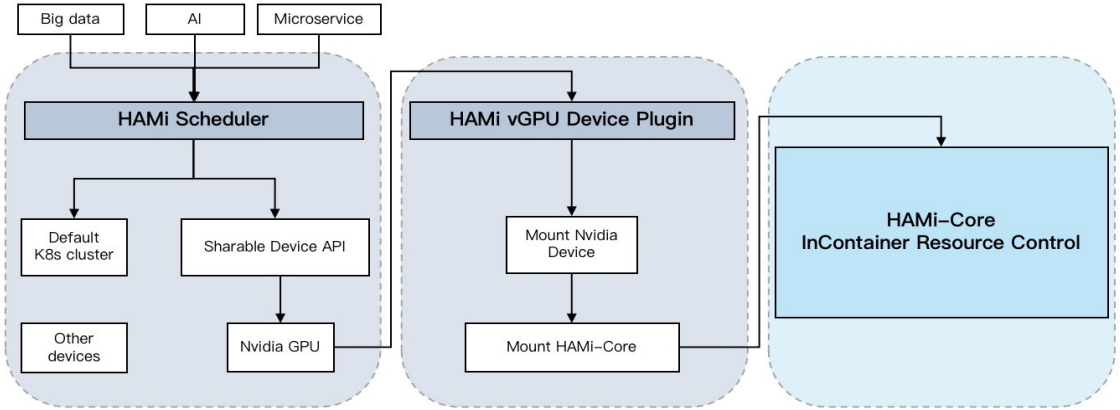
HAMi: Middleware for GPU Clusters

KubeCon CloudNativeCon OpenInfra SUMMIT ASIA China 2026 PyTorch CONFERENCE



- Device Slicing
- Scheduling Optimization
- Observability

HAMi architect



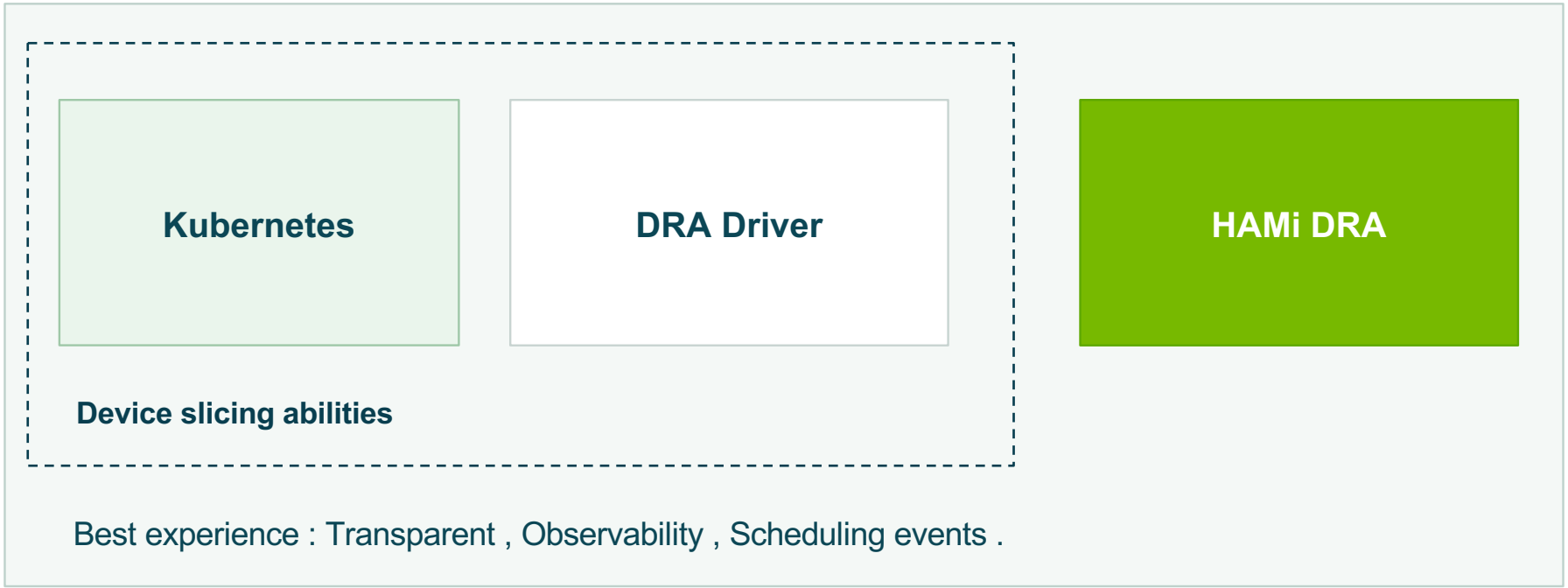
Scheduling layer:

- Pluggable
- Green
- Compatible

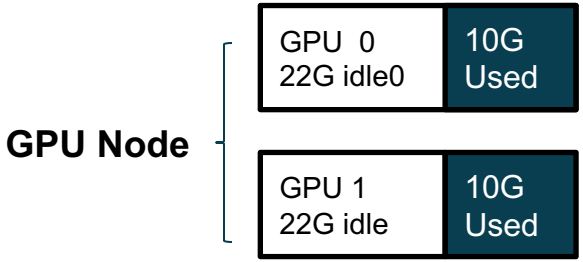
CUDA layer:

- Transparent to GPU tasks

HAMi DRA


Kubernetes
DRA Driver
HAMi DRA
Device slicing abilities
Best experience : Transparent , Observability , Scheduling events .

HAMi Usage




```
root@gpu-demo-6b6b88b75b-x9tjb:~# nvidia-smi
[HAMI-core Msg(35:140111864956736:libvgpu.c:836)]: Initializing.....
Thu Apr 18 06:22:54 2024
+-----------------------------------------------------------------------------+
| NVIDIA-SMI 535.104.12             Driver Version: 535.104.12   CUDA Version: 12.2     |
|-------------------------------+----------------------+----------------------+
| GPU  Name                 Persistence-M | Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp   Perf          Pwr:Usage/Cap |         Memory-Usage | GPU-Util  Compute M. |
|                                         |                      |               MIG M. |
|===============================+======================+======================|
|   0  NVIDIA A800 80GB PCIe          On  | 00000000:13:00.0 Off |                    0 |
| N/A   36C    P0              81W / 300W |      0MiB / 10240MiB |      0%      Default |
|                                         |                      |             Disabled |
+-------------------------------+----------------------+----------------------+
|   1  NVIDIA A800 80GB PCIe          On  | 00000000:1C:00.0 Off |                    0 |
| N/A   39C    P0              82W / 300W |      0MiB / 10240MiB |      0%      Default |
|                                         |                      |             Disabled |
+-------------------------------+----------------------+----------------------+

+-----------------------------------------------------------------------------+
| Processes:                                                                  |
|  GPU   GI   CI        PID   Type   Process name                  GPU Memory |
|        ID   ID                                                   Usage      |
|=============================================================================|
+-----------------------------------------------------------------------------+
[HAMI-core Msg(35:140111864956736:multiprocess_memory_limit.c:434)]: Calling exit handler 35
root@gpu-demo-6b6b88b75b-x9tjb:~#
```

```
apiVersion: v1
kind: Pod
metadata:
  annotations:
    hami-schedule-policy: “mutex, numa, spread”
  name: gpu-pod12
spec:
  containers:
    - name: ubuntu-container
      image: ubuntu:18.04
      command: ["bash", "-c", "sleep 86400"]
      resources:
        limits:
      nvidia.com/gpu: 2 # requesting 2 vGPUs
      nvidia.com/gpumem: 10240
      nvidia.com/gpucores: 30
```

HAMi Supported devices

NVIDIA

aws

Ascend

Cambricon 寒武纪科技

Enflame 燧原科技

HYGON 中科海光

天数智芯 Iluvatar CoreX

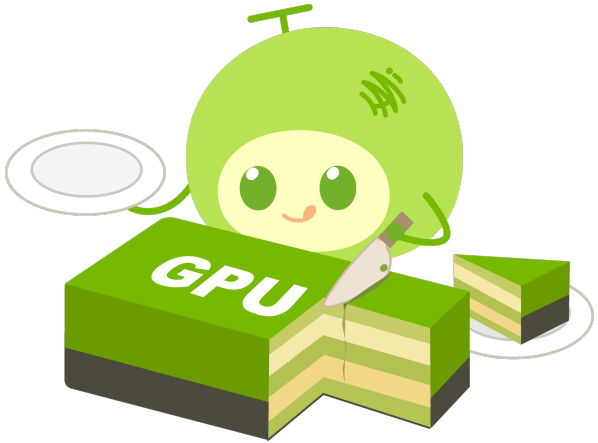
昆仑芯 KUNLUNXIN

METAX 沐曦

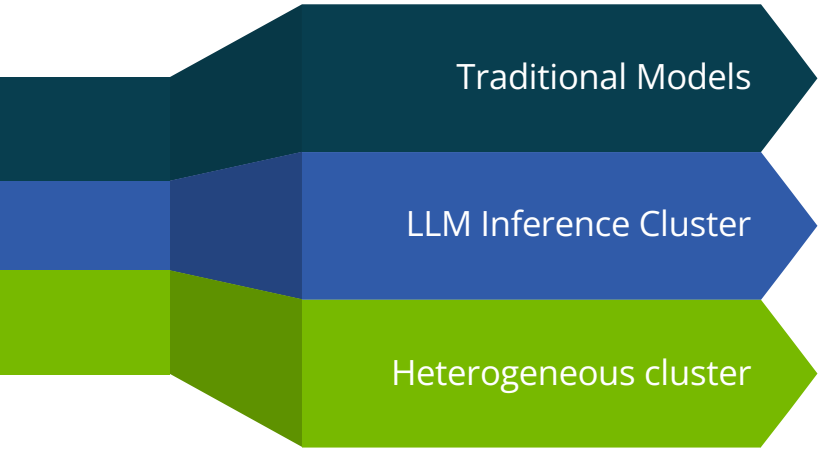
摩尔线程 MOORE THREADS

AMD

壁仞科技 BIREN TECHNOLOGY



HAMi: Use Cases

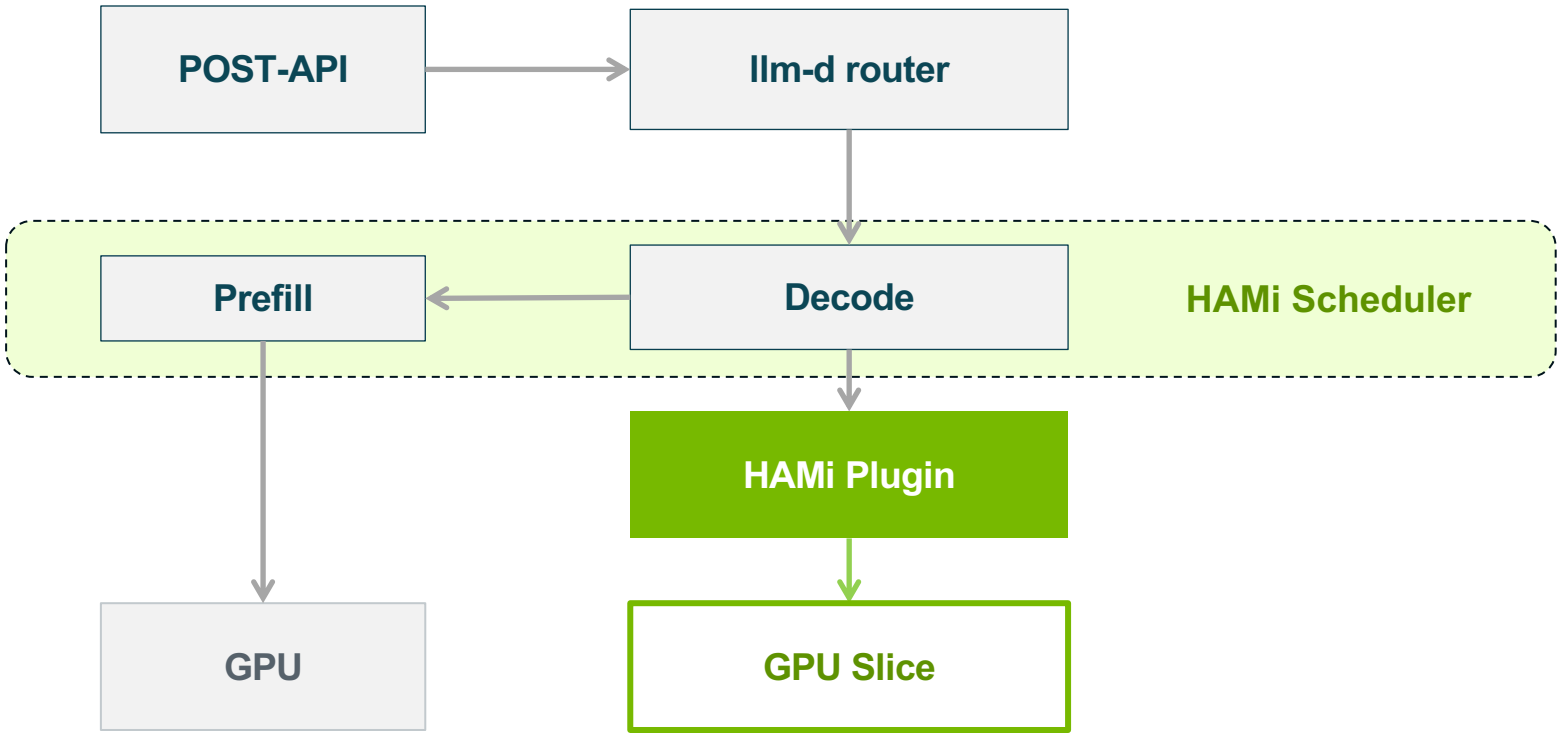


A/B Testing, Embedding, RL

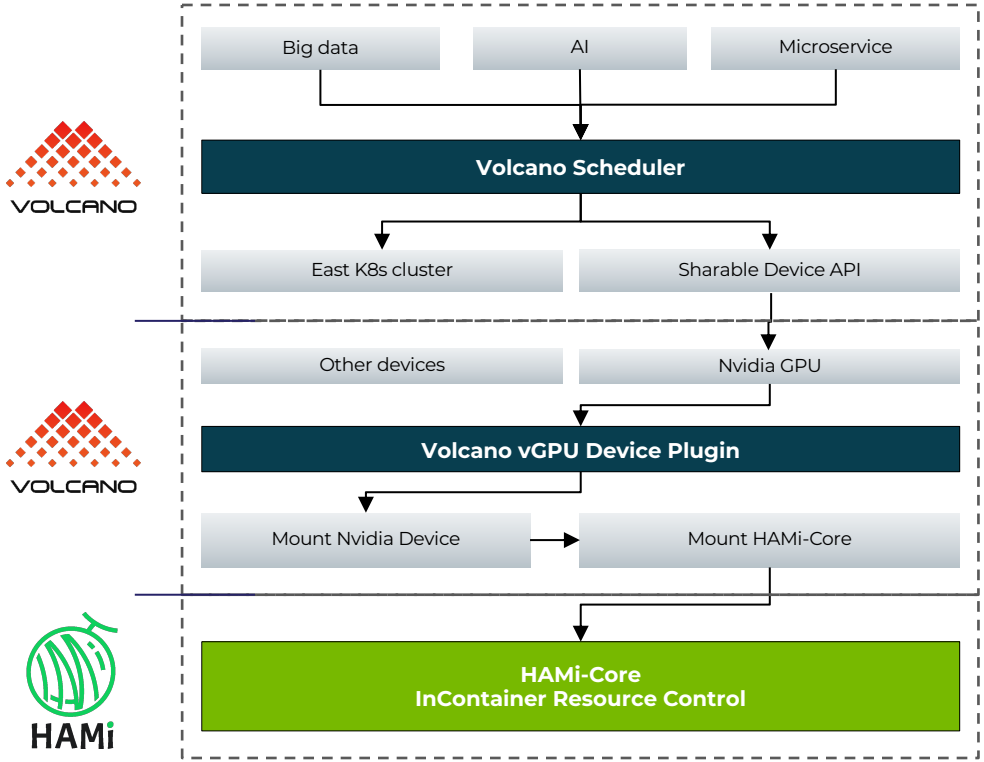
mooncake, llm-d, dynamo

Ascend 910X, MetaX+Nvidia

HAMi in LLM inference



Volcano + HAMi




```
$ cat <<EOF | kubectl apply -f -
apiVersion: v1
kind: Pod
metadata:
  name: gpu-pod12
spec:
  schedulerName: volcano
  containers:
    - name: ubuntu-container
      image: ubuntu:18.04
      command: ["bash", "-c", "sleep 86400"]
      resources:
        limits:
          volcano.sh/vgpu-number: 2 # requesting 2 vGPUs
          volcano.sh/vgpu-memory: 10240
```

```
root@gpu-demo-6b6b88b75b-x9tjb:~# nvidia-smi
[HAMI-core Msg(35:140111864956736:libvgpu.c:836)]: Initializing.....
Thu Apr 18 06:22:54 2024
+-----------------------------------------------------------------------------+
| NVIDIA-SMI 535.104.12             Driver Version: 535.104.12   CUDA Version: 12.2     |
|-------------------------------+----------------------+----------------------+
| GPU  Name                 Persistence-M | Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp   Perf          Pwr:Usage/Cap |         Memory-Usage | GPU-Util  Compute M. |
|                                         |                      |               MIG M. |
|===============================+======================+======================|
|   0  NVIDIA A800 80GB PCIe          On  | 00000000:13:00.0 Off |                    0 |
| N/A   36C    P0              81W / 300W |      0MiB / 10240MiB |      0%      Default |
|                                         |                      |             Disabled |
+-------------------------------+----------------------+----------------------+
|   1  NVIDIA A800 80GB PCIe          On  | 00000000:1C:00.0 Off |                    0 |
| N/A   39C    P0              82W / 300W |      0MiB / 10240MiB |      0%      Default |
|                                         |                      |             Disabled |
+-------------------------------+----------------------+----------------------+

+-----------------------------------------------------------------------------+
| Processes:                                                                  |
|  GPU   GI   CI        PID   Type   Process name                  GPU Memory |
|        ID   ID                                                   Usage      |
|=============================================================================|
+-----------------------------------------------------------------------------+
[HAMI-core Msg(35:140111864956736:multiprocess_memory_limit.c:434)]: Calling exit handler 35
root@gpu-demo-6b6b88b75b-x9tjb:~#
```

CaseStudy: IntSig_训推总览

统一调度 · 弹性供给 · 高效利用

万卡集群
GPU 总规模
自建 + 多云混合部署

1000+
在线推理服务
服务实例在线运行

1000+
离线训练任务
训练任务并行调度

90%+
24 小时利用率
训练侧持续水位

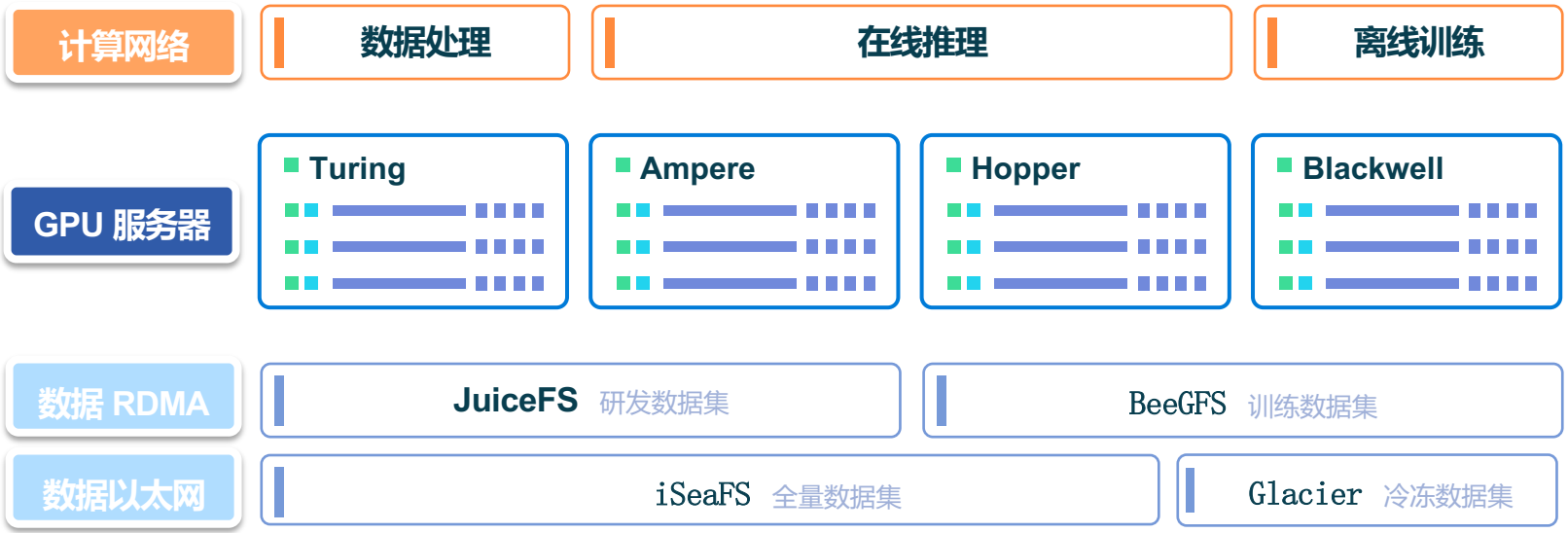
异构算力覆盖

训练	Blackwell	Hopper	Ampere	Ascend	
推理	Blackwell	Hopper	Ampere	Turing	Ascend

GPU 服务器部署在：自建数据中心 · 阿里云 · AWS · 腾讯云 · 火山引擎

CaseStudy: IntSig_混合调度

异构 GPU 服务器共池 · 计算与存储同机部署



计算任务与分布式存储在同一批 GPU 服务器上，按计算网络 / 数据 RDMA / 数据以太网分层承载流量

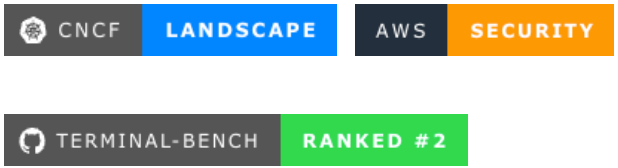
CaseStudy: IntSig_GPU服务器与K8S 资源管理

从资源切分到监控闭环的完整落地路径

Chaterm AI

通过自然语言管理EC2，RDS，K8S的开源Harness

荣誉与认证



沙利文《2025年 中国 生成式AI 最佳应用实践》

Akasha AI

本地优先的企业级开源自进化记忆系统

核心能力

1. **让碎片信息变成知识**
   将文档、会议、对话、邮件等多渠道信息统一沉淀，减少知识流失。
2. **让知识持续保持准确**
   通过 Dream Cycle 自动整理、验证和更新知识，并逐步构建知识图谱。
3. **让 Agent 掌握领域经验**
   通过 Skills 和渐进式披露，为 Agent 提供领域知识和操作经验。

https://github.com/chaterm/Chaterm，https://github.com/chaterm/Akasha

CaseStudy: IntSig_HAMi 部署方式与收益

从资源切分到监控闭环的完整落地路径

部署方式

1. **切片优先 • 整卡兼容** 小模型推理切片使用，高负载与大模型保留整卡
2. **Binpack 优先装箱** 优先填满已有 GPU，减少半卡碎片与节点分散
3. **亲和性搭配** 高负载与低负载混布，避免两个高负载竞争同一物理卡
4. **监控闭环** HAMi 分配指标接入原有 GPU 监控，统一面板看分配到使用
5. **Karpenter 弹性扩容** 改造插件兼容资源识别，海外集群 GPU 节点无感扩容

使用收益

GPU 利用率提升	+50%
综合成本降低	-30%
推理性能下降可控	<10%

通过 HAMi 解决原有GPU分片方案对操作系统和内核的限制要求，实现多云GPU统一管理

Join our community

KubeCon CloudNativeCon OpenInfra SUMMIT ASIA China 2026 PyTorch CONFERENCE

- **GitHub**
  https://github.com/Project-HAMi/HAMi
- **Website**
  http://project-hami.io/
- **Slack**
  #hami-dev in Cloud Native Computing Foundation


KubeCon
CloudNativeCon
OpenInfra
SUMMIT › ASIA
PyTorch
CONFERENCE
China 2026