English Version →

产品与技术概览

OpenSAI 产品与技术概览

面向边缘异构设备的分布式 DAG 计算卸载与资源-应用双向自适应平台。

快速上手指南 | 软件使用手册 | 智能体接入指南 | 常见问题与排障

1. 为什么边缘协同需要 OpenSAI

在车路协同感知、多路视频分析、工业现场检测与分布式边缘数据处理等场景中，数据往往分散产生于车辆、路侧单元、摄像头终端与边缘服务器之间。单台终端算力有限，而将全部原始数据回传中心云又会面临高时延与带宽瓶颈。要让附近多台 CPU / GPU 设备像同一台机器那样协同完成一项任务，开发者通常面临四重工程阻碍：

- **数据分散与隐私本地性约束**：传感器数据驻留在不同终端，计算步骤需要就近读取源数据，并自动在设备间传递中间特征或分片结果。
- **多阶段依赖与硬编码耦合**：一个完整应用往往包含“采集预处理 → 多分支特征提取 → 跨设备汇聚 → 融合推理”，手工编写跨机通信与编排逻辑极难复用。
- **异构算力与动态网络波动**：集群中既有轻量 CPU 节点也有边缘 GPU 服务器，无线链路带宽随环境起伏，协作节点还可能随时加入或离线。
- **资源调度与应用调参彼此割裂**：传统调度器假定应用参数固定不变，而应用层调参又不了解集群实时负载——负载紧张时容易超时，资源充裕时又无法充分释放模型精度。

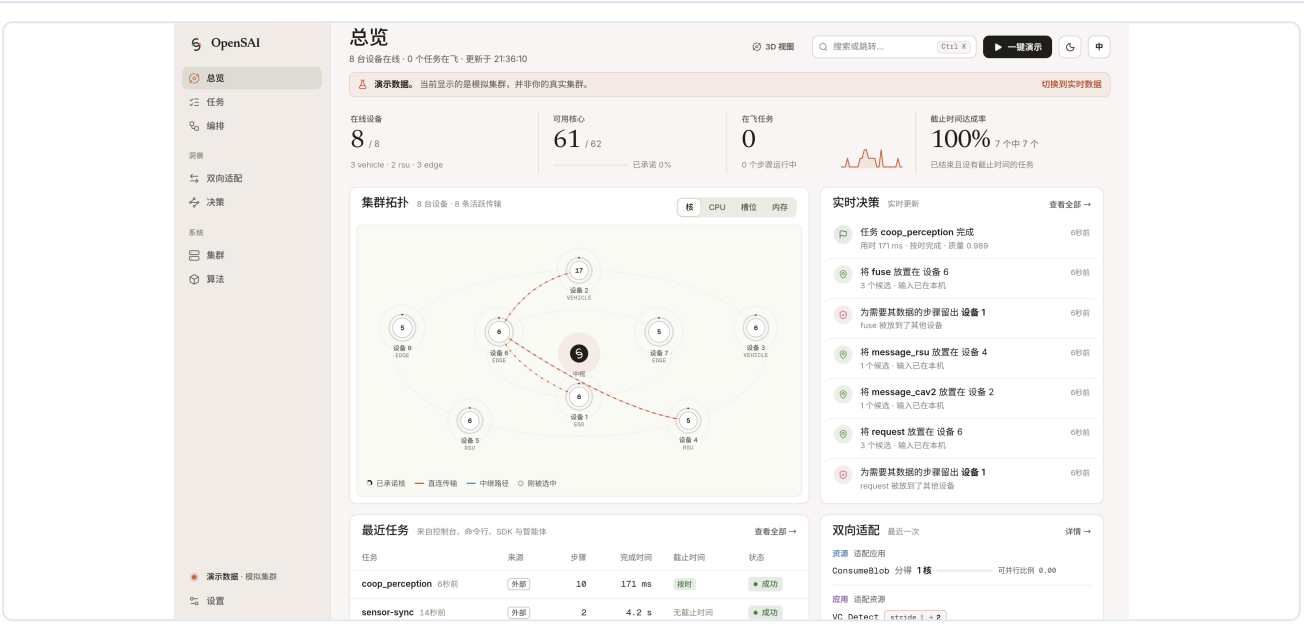
OpenSAI 位于业务应用与边缘异构资源之间：你只需将 Python 应用声明为带依赖关系的步骤图（DAG）与应用清单（`app.yaml`），平台即可统一接管设备发现、数据就近放置、四维资源分配、跨设备 QUIC 传输与运行期参数自适应，并通过双语 Web 控制台、CLI、Python SDK 与 MCP 智能体接口提供一致的操作体验。



OpenSAI 三层系统架构与双向适配闭环

2. 核心技术能力

核心维度	OpenSAI 提供的机制	给开发者与使用者带来的价值
声明式 DAG 卸载与容错	自动解析步骤依赖，无依赖分支并行派发；支持 `data` 数据本地性绑定与 `optional` 可选协作分支	业务代码只关注单步逻辑；协作节点暂时离线时自动跳过可选分支，保障主流程连续可用
资源与应用的双向适配	**资源适配应用**：联合规划执行设备、CPU 核数、GPU 与带宽配额；**应用适配资源**：按时间或质量目标自动调节分辨率、采样率、特征维数、上传比例等“旋钮”	通过 `-k` 锁定关键业务约束，将其余弹性参数交给平台权衡，兼顾响应时效、业务质量与资源效率
QUIC 直传与中继多链路	同设备步骤直接本地读取；跨设备中间产物经 QUIC（`UDP 9999`）点对点传输，按需开启 `DELTA` 统一中继多链路（`OPENSAI_MULTIPATH=1`）	无需修改内核或部署 MPTCP/MPQUIC，即可在直连受限时借用集群空闲中继路径加速传输
四维资源控制与闭环画像	进程池执行环境施加 CPU 核亲和、内存预算、出口带宽限速与产物存储回收；每次实测耗时持续回流更新性能画像	无需繁重容器编排即可在 Windows / Linux / macOS 上隔离运行，调度预测随运行不断校准
全链路可观测与智能体协同	内置中英双语 Web 控制台、命令行 `opensai`、`OpenSAIClient` SDK 及 `opensai_mcp` + 使用 Skill	既可在可视化界面直观审查每一步的放置、资源与调参依据，也可授权 AI 智能体用自然语言操控集群



OpenSAI 双语 Web 控制台总览视图

3. 典型应用场景与内置示例

发布包内置 5 个由浅入深的可运行应用示例，无需额外硬件即可在单台电脑跑通验证，随后无缝扩展至多机边缘集群：

内置示例	代表场景	重点演示的平台能力	运行准备
`number_stats`	分阶段数值采样统计	应用包最小完整结构、跨步骤产物传递、采样率旋钮	仅需 Python 标准库（零额外依赖）
`cluster_analytics`	多分片并行聚类分析	双路并行分片计算、全局簇中心合并、`dims` / `iters` 目标驱动适配	安装 `numpy`；自动生成本地合成数据
`video_counting`	多路视频流目标计数	双分支视觉流水线、分辨率与抽帧间隔联合调节	安装 `numpy`；内置合成视频帧生成器
`coop_perception`	多车与路侧轻量协同感知	自车/协作车/路侧三源点云本地性、通信比例 `comm` 控制、逐帧回放	安装 `numpy`；一键生成本地点云场景
`how2comm_perception`	路口三维激光雷达协同检测	真实 How2comm 模型边缘 GPU 推理、协作方动态缺席容错、按链路调节上传比例	边缘节点配置 PyTorch / GPU、How2comm 源码权重与 V2XSet 数据

4. 三步开启首次体验与文档导航

只需一台安装 **Python 3.10+** 的电脑即可启动完整本地集群（中枢 + 工作设备 + Web 控制台），无需 GPU、Docker 或 Node.js。克隆仓库（或解压发行版 `OpenSAI_Release.zip`）后在项目根目录执行：

```bash
git clone https://github.com/INSAI-institute/OpenSAI.git
cd OpenSAI
bash scripts/setup.sh
bash scripts/run.sh --with-web
source .venv/bin/activate
opensai dag submit -f examples/dags/pi_pipeline.json
```

Windows PowerShell 对应命令及完整结果解读请打开 快速上手指南。浏览器访问 `http://127.0.0.1:5000` 即可进入控制台（支持右上角 **EN / 中** 语言切换、**纸 / 墨** 浅深色主题切换与 **一键演示** 提交）。

按你的目标选择阅读路径

- **首次安装与 10 分钟实操闭环**：阅读 快速上手指南，完成首个 DAG、自定义应用模板与目标驱动调参对比。
- **多机集群部署、应用开发与 API 速查**：查阅 软件使用手册，涵盖端口规划、清单规范、QUIC 多链路、SDK 与 CLI。
- **连接 Claude / Cursor / OpenClaw 等智能体**：先看 智能体接入指南 完成 MCP 配置，再参考 Agent Skill 规范说明。
- **故障排查、开源合规与学术引用**：遇到异常查阅 常见问题、排障与维护；分发与合规查看 开源许可与第三方说明；参与共建或在论文中引用请参考 贡献与学术引用说明。

机构、公开仓库与开源许可

OpenSAI 由 **复旦大学（Fudan University）** 与 **Institute on Networking Systems of AI (INSAI)** 联合研发并发布，官方代码仓库位于 github.com/INSAI-institute/OpenSAI（完整发行包与演示视频见 Releases），采用 MIT License 开源许可（详见 English Overview）。