SC26 Chicago, IL hpc unites.

pdsw

Agents in Data Infrastructure: From Telemetry to Optimization

Dong Dai

University of Delaware

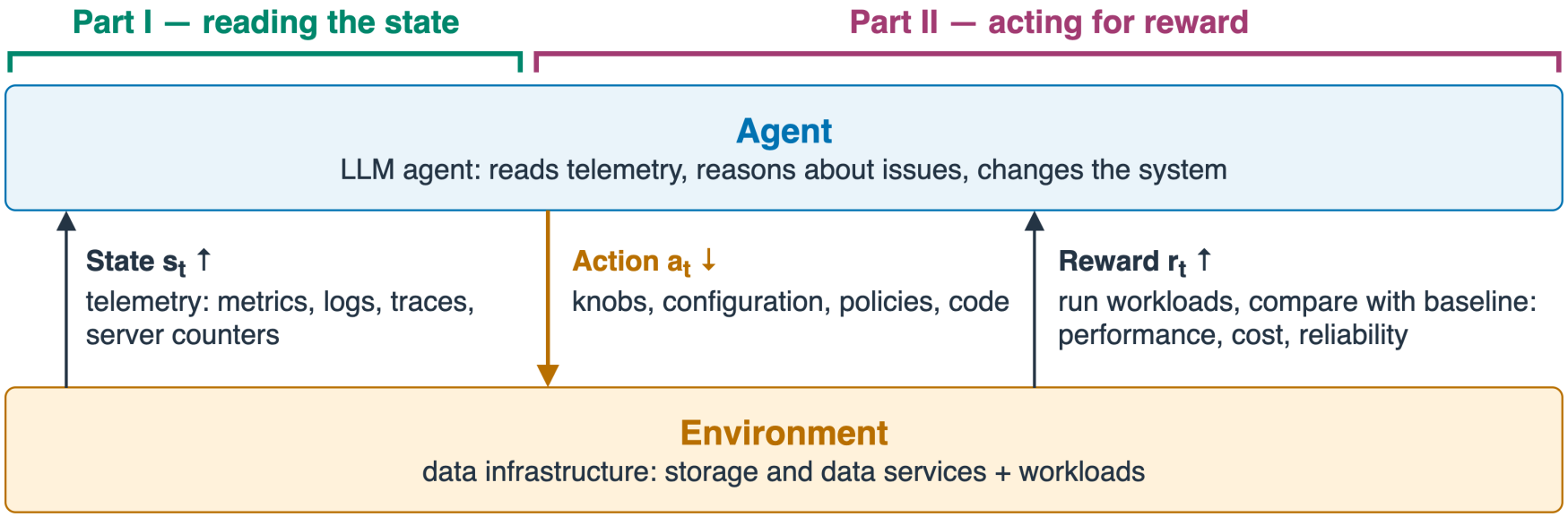
PDSW'26 talk · Mon Nov 16, 2026

University of Delaware

DIRLAB

DIRLAB

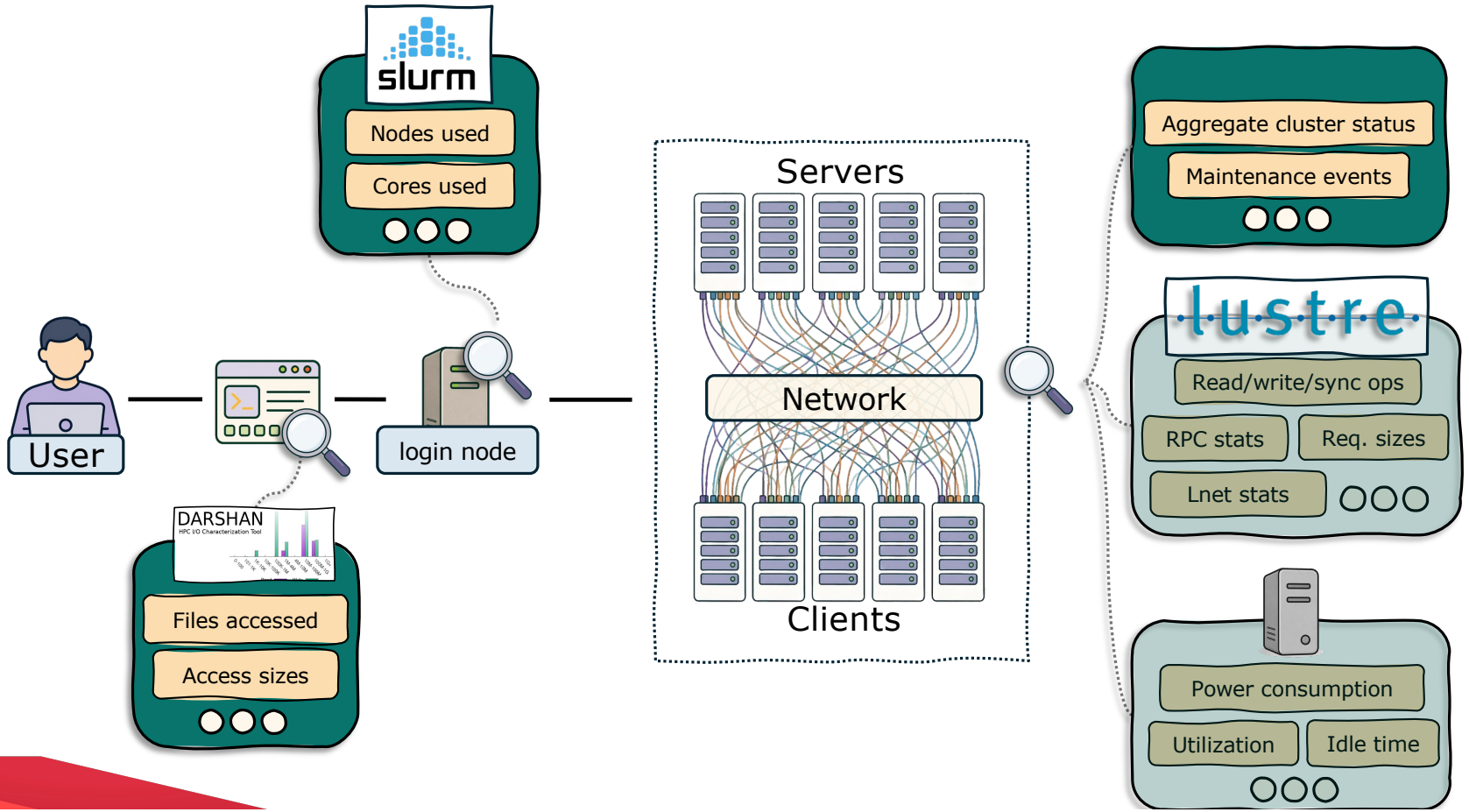
Agents Interact with Data Infrastructure



Part 1: Read the state - I/O Telemetry Analysis/Understanding

DIRLAB

HPC Systems Generate Rich I/O Telemetry



DIRLAB

	2011–22	2023–24	2025	2026	
Tracing & non-LLM Analysis	**Darshan** TOS'11 **UMAMI** PDSW'17 **DXT Explorer** PDSW'21 **Drishti** PDSW'22	**AIIO** HPDC'23 **IOMax** PDSW'23 **DFTracer** SC'24 **ExaQuery** ICPE-C'24	**WisIO** ICS'25 **XIO** SSDBM'25	**DataCrumbs** ProTools@SC'26 **GDSight** PDSW'26 **STARCH** PDSW'26	
HPC / storage LLMs + agents		**ChatGPT PFS logs** HPDC'23 (Poster) **ION** HotStorage'24	**IOAgent** IPDPS'25	**EPIC** ISC'26 **KORAL** IPDPS'26 **SMARTTalk** OSDI'26 **Metis** HPDC'26	**Symbolic Separation** preprint'26 **IOSage** preprint'26 **CIODI-Bench** SC'26
Database / cloud LLMs + agents		*D-Bot* PVLDB'24 *RCACopilot* EuroSys'24 *RCAgent* CIKM'24	*DBAIOps* preprint'25	*Cloud-OpsBench* preprint'26;	

DIRLAB

LLM Agents on Telemetry Analysis: Our Work

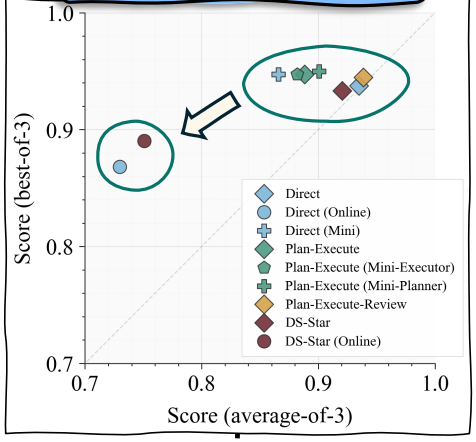
Problem type	Typical questions	Related work
Telemetry preparation	Which reports describe the same event? What should be filtered or labeled?	**Ongoing work** Expert-guided event reduction
Interactive telemetry analytics	What share of node time has traces? Is MPI-IO more common in larger jobs?	**CIODI** (SC'26) **EPIC** — operational analytics
Job / device diagnosis	Which access patterns hurt a job's I/O? What explains abnormal SSD behavior?	**IOAgent** (IPDPS'25) — Darshan job traces **KORAL** — SSD reasoning
Facility-wide incident discovery	When and where did I/O degrade? Do slow jobs share a system problem?	**Ongoing work** - Job reports + system trends

DIRLAB

LLMs + Agent on Telemetry Analysis: Works but Has Issues

SC26 Paper: Understanding Reliability Bottlenecks in LLM Agents for HPC I/O Telemetry Analysis, ***Tue Nov 17, 11:15 am, W175abc, Paper Talk (main results)***

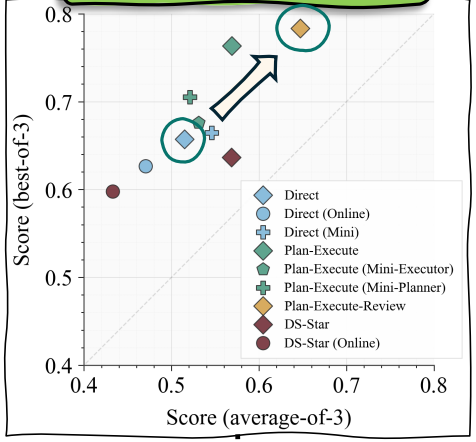
Data Extraction



Data processing hinges primarily on accurate understanding

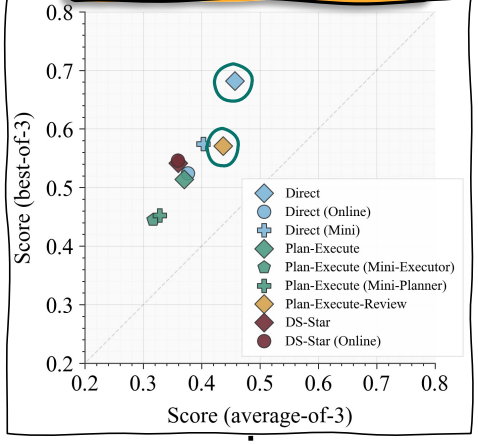
Dedicated schema discovery effort does not make up the gap

Data Analysis



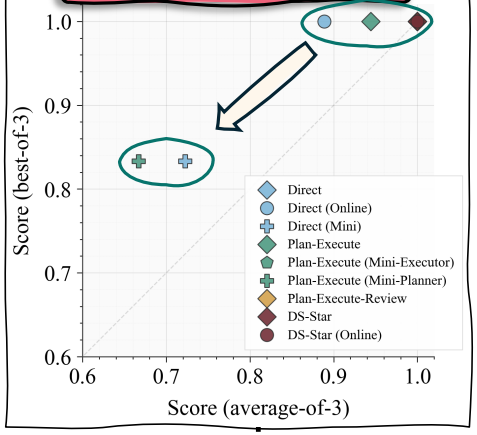
Orchestration supporting review helps when answers are enumerable

Insight Generation



Orchestration does not noticeably benefit open-ended tasks

Unsolvable

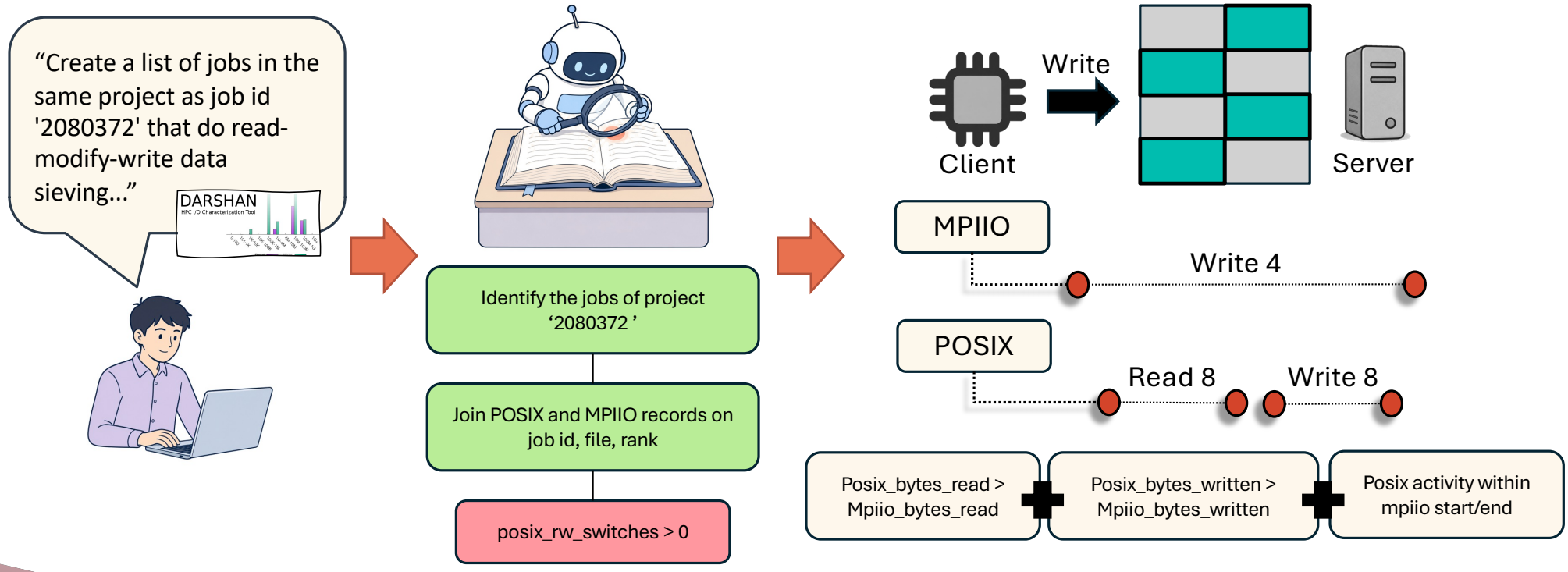


Weaker models less reliably abstain

DIRLAB

Issue 1: Missing context the agent does not know it lacks

Examples



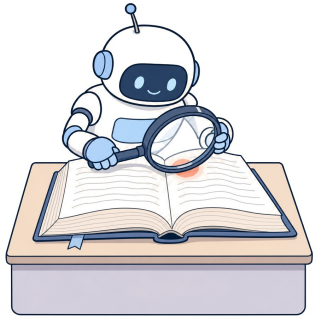
DIRLAB

Issue 1: Missing context the agent does not know it lacks

Examples

"Label each event Minor / Major / Critical — refer to the Intel manual." (lazy prompt)





Scans the manual → anchors on the **most salient** bit: **OVER** — set on **1.67M of 1.93M rows**.

'confidently' commits a wrong rule

`Major ⟵ UC==0 & PCC==0 & VAL==1 & OVER==1`

✓ **runs clean**
✓ **passes every check**
✓ **output looks plausible**
→ did not record trace

"Defensible — but wrong. Our convention uses **UC / PCC / VAL** only."



DIRLAB

Issue 1: Missing context the agent does not know it lacks

What we did

- **Curated semantics**: CIODI extraction 73.0% → 93.5%. We have to know the issue beforehand
- **What if we do not know failures**? 97.8–100% in all 20 settings, ≤ 4 questions per run; replaying stored answers: 16 → 4 questions; 150k–502k tokens per run vs 3k–64k.
- **Did not work**:
  - prompting for curiosity (asks always or never);
  - an optional ask-the-expert tool;
  - online clarification ("substantial unnecessary escalations").

When should experts get involved?

A agent's choice

B alternative

C alternative

same saved input; compare row counts and label distributions

agree → continue

differ → ask the expert, with the executed evidence

```
HITL PAUSE #1 -- Phase 3 - Exact-Dedup
Option A: Keep the agent's own validated result.
  -> rows: 307931 -> 307931  (removed 0, retained 100.0%)
Option B: Remove only records that are exact duplicates
  -> rows: 307931 -> 307929  (removed 2, retained 100.0%)
> T  Remove exact duplicates on the event identity: rows
  with the same ERROR_CODE, EVENT_EPOCH_TS and
  FAILED_COMPONENT are one hardware error ...
  -> rows: 307931 -> 128710 (removed 179221, retained 41.8%)
> Y
```

What's next

- Knowing when to ask.
- Machine-readable semantics shipped with the telemetry.
- Reusing expert answers across runs and sites.

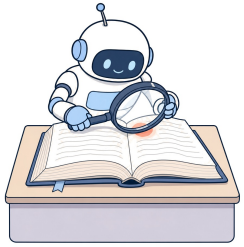
DIRLAB

Issue 2: Tunnel Vision, once committed, the agent defends its choice

Examples

"Identify the features of the top performing jobs using MPIIO which explain their high performance"



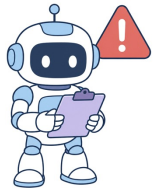


Review MPIIO access sizes, patterns characteristics. Find general characteristics: large sequential access

`MPIIO: files in '/lus/eagle'`

`POSIX: files in '/local/scratch'`

"Ignore Posix as files don't match by name to MPIIO"



POSIX activity in '/local/scratch' aligns exactly on time with MPIIO activity

MPIIO Performance is inline with '/local/scratch' memory access

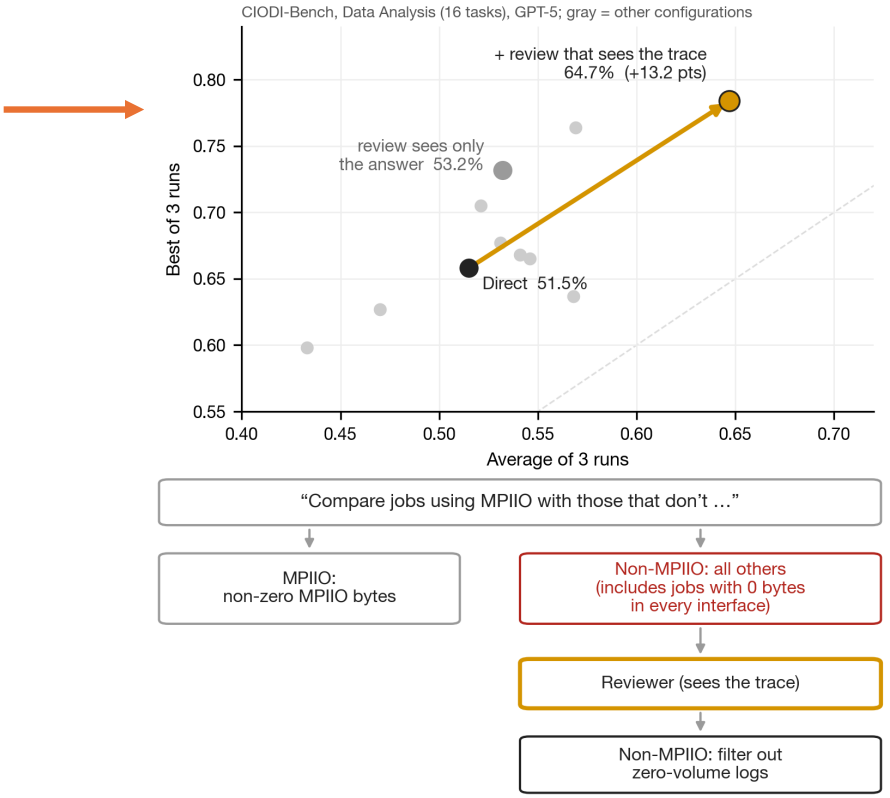
Job most likely does local staging

DIRLAB

Issue 2: Tunnel Vision, once committed, the agent defends its choice

What we did

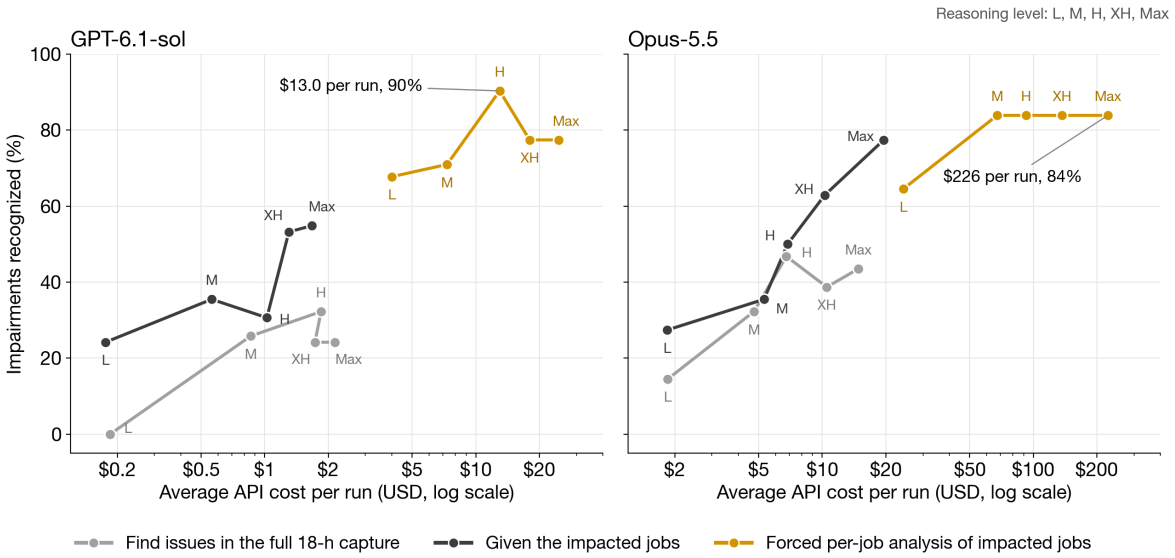
- **Process-aware review:** COIDI analysis: 51.5% → 64.7%.
- **Diverse Population**? AlphaEvolve stores many candidate programs in a database and keeps them diverse with MAP-Elites and island-based selection.
- **LLM Proposes but other methods explores**: GPTuner (LLM guilds how to do Bayesian Optimization)
- **Did not work**:
  - A reviewer that reviews only the final answer. 53.2% on Data Analysis, against 51.5% with no review.
  - More orchestration, A heavy multi-agent pipeline



DIRLAB

Issue 3: Cost, good answers to hard questions still cost too much

Examples



Settings:

1. **Given various telemetry data within a small time window of an HPC system.**
2. **Build an agent to query needed information and identify slowness and its root causes.**

Observations:

1. **Frontier models can do it very well.** When you do not care about latency and cost.

All costs are estimation (we use subscription or we will bankrupt)

Part 2: Acting for Reward - Optimization

DIRLAB

	2013–22	2024	2025	2026
Search / ML / RL	**H5Tuner** SC'13 **CAPES** SC'17 **Magpie** IC2E'22 **VAE-guided BO** CLUSTER'22	**TunIO** IPDPS'24	**Dial** CCGrid'25	**COMPASS** preprint'26 General HPC; not LLM
LLM-guided Storage / I/O tuning		**ELMo-Tune** HotStorage'24	**STELLAR** SC'25 **ELMo-Tune-V2** preprint'25 **StorageXTuner** preprint'25	**Rudder** ICS'26 GNN prefetch control
DB / systems Language-guided tuning		*GPTuner* *PVLDB'24*	*λ-Tune* *SIGMOD'25*	*SysInsight* *PVLDB'26* *IDSTune* *preprint'26* *SysPilot* *preprint'26*
Broader systems Code synthesis			*PolicySmith* *HotNets'25* *AlphaEvolve* *preprint'25*	

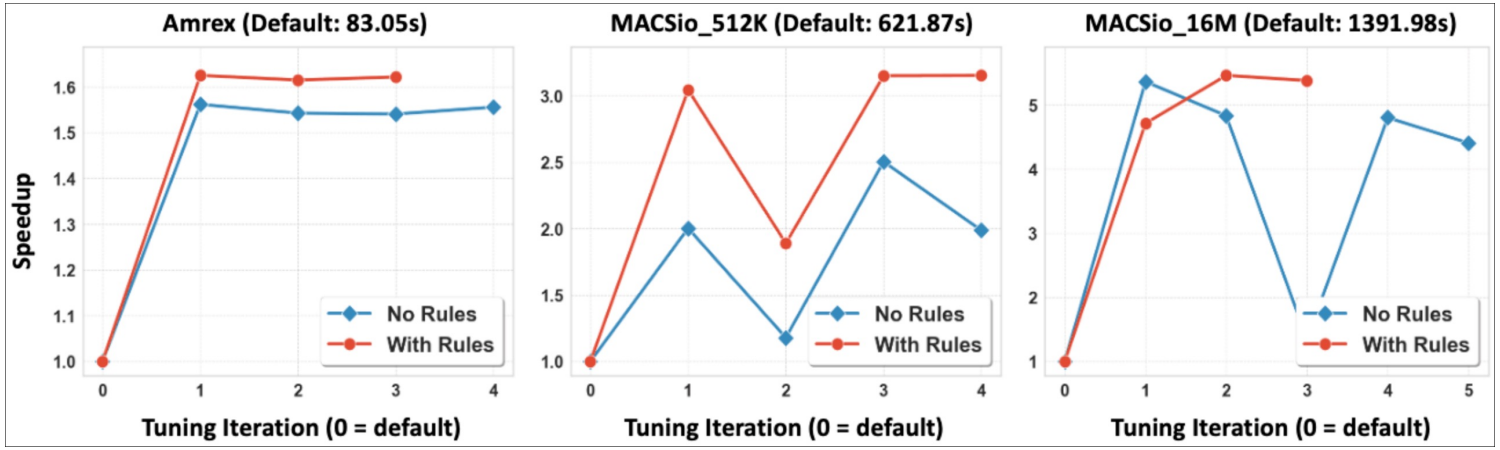
DIRLAB

Research Question 1: What should the agent change — a knob, a composition of components, or the code?

Current practice

Knobs: H5Tuner → STELLAR, StorageXTuner.
Code: ADRS, AlphaEvolve, PolicySmith.
Composition: Mochi makes it possible, but no agent chooses the composition yet (Dorier, CLUSTER'22, tunes HEPnOS parameters with the composition fixed).

STELLAR (13 Lustre parameters) and DAOS



DIRLAB

Research Question 1: What should the agent change — a knob, a composition of components, or the code?

Problems

- Knobs cap the gain.
- Spaces are large: Lustre 2.12.5 has 159 user parameters; Ceph Nautilus, 1,536.
- Code needs a reliable verifier, generalizes poorly (PolicySmith is best on 14–64% of traces), and can game the score.

Where to go

- Climb gradually: knob → composition → code.
- Build verifiers and safety checks first.
- Read code to set knobs before rewriting code.
- Composition is an open opportunity

DIRLAB

Research Question 2: Benchmarks & Applications. What are we optimizing for?

Current practice

Benchmarks: IOR, IO500, mdtest;
A few real applications: Real-application tuning cost is dominated by runtime.
Middle ground: I/O kernels and trace replay — Behzad et al., PDSW'14; Skel, 2011; TunIO, IPDPS'24; FBench, 2026 (LAMMPS replay 101.8 s vs 566.5 s).

Problems

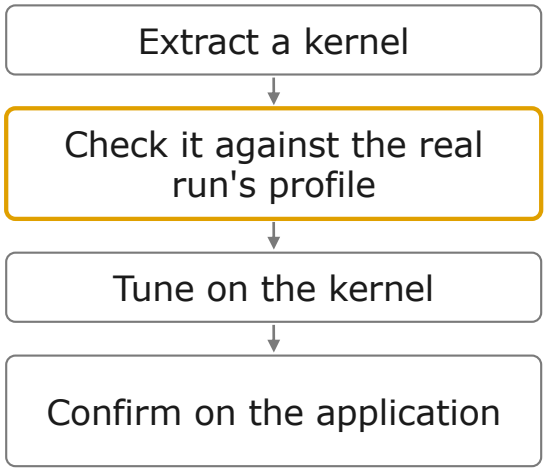
- **Tuned parameters** depend on application, platform, problem size and concurrency (Behzad et al.).
- A benchmark–application metric **mismatch** has been observed (Liem et al., SSDBM'26).

Ours

Open question: how much trace is enough to rebuild a faithful kernel? Darshan is too little; full DXT or Recorder traces are too much. No results yet.

Polaris: of 20,956 job pairs with near-identical Darshan summaries, 758 still differ by > 5×. Each trial must be repeated.

Where to go



Research Question 3: Tunnel Vision of the LLM Optimizer

Current practice

The LLM proposes, the system runs it, the LLM reflects and proposes again (STELLAR, ELMo-Tune);

StorageXTuner adds tree search.

Problems

- LLMs "do not robustly engage in exploration without substantial interventions" (Krishnamurthy et al., NeurIPS'24).
- Greedy search plateaus (AIDE).
- Our observation: a good first trial is refined; a bad one stays stuck.
- No storage-tuning paper measures this yet.

Where to go

- Populations and novelty (AlphaEvolve; ShinkaEvolve: weighted parent sampling beats hill climbing).
- Archives of past candidates.
- LLM + Bayesian optimization (GPTuner).
- RL-style exploration schedules (CAPES).

Summary

- Agents can already analyze telemetry, but only for questions whose meaning someone has written down

Agents will be part of the data-infrastructure stack. Our community's knowledge (the meaning of telemetry, trustworthy rewards, and experts in the loop) determines how well they work.

- For optimization, the open problem is a reward signal we can trust and afford.

DIRLAB

Q & A

Tue Nov 17, 11:15 am, W175abc

Understanding Reliability Bottlenecks in LLM Agents for HPC I/O Telemetry Analysis

Chris Egersdoerfer — CIODI-Bench full results

Wed Nov 18, 12:15–1:15 pm, W183b

BoF: Operational Data Analytics: Agentic AI for HPC Facilities

daidong.github.io

I am around all week — stop me and talk.