MIDWEST RCD 2026 · BREAKOUT 1

Hello Computer

HPC in the Agentic Era

Geoffrey Lentner

Principal AI Scientist · Rosen Center for Advanced Computing · Purdue University

STILL FIGURING THIS OUT

WIP: Work in Progress

TODAY'S TALK · WALKING ALL THREE LEGS

RCAC’s posture on agentic AI

Proactive engagement, not prohibition.

/etc/agents.d

System-wide configs

Cluster context every agent inherits before turn one.

purpose-built

MCP servers

`rcac-mcp` · `globus-mcp`

Typed tools the agent actually understands.

docs.rcac

Researcher-facing documentation

What agents can and can't do, and how to verify what they say.

Working hypothesis: agentic AI is **mostly harmless** *— if you shape its context.*

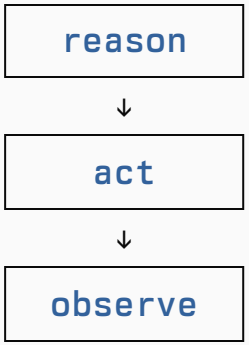
The agentic era · the CLI Renaissance

Under four years: chatbot → copilot → **agent**.

TIMELINE

2022 ● ChatGPT
2023 ● GPT-4 · Claude · the "copilot"
2024 ● MCP, the ReAct pattern at scale
2025 ● agentic IDEs (Warp, Cursor, Claude Code)
2026 ● semi-autonomous research workflows

An **agent** is a language model in a loop with **tools** and **memory**, pursuing a user goal over multiple steps.



↻ *loop back to reason*

CLI RENAISSANCE

2020

GitHub `gh`

```
$ gh pr view 42 --json \
    state,reviews,checks
```

Text in, text out, JSON on request. Every modern agent harness loves it.

2026

Google Workspace CLI

```
$ gworkspace docs export \
    --doc-id 1aB… \
    --as pdf
```

New. Vendors feel the pressure: agents *demand* first-class CLIs.

Anything the shell can do, the **agent** *can do.*

Leg 1 · `/etc/agents.d`

Cluster context every agent inherits, before turn one.

EXAMPLE · `/etc/agents.d/cluster.md`

```
# RCAC cluster context
```

- **Storage:** myquota for usage.
  $HOME is small/persistent;
  $SCRATCH is fast, 30-day purge;
  $DEPOT is group / long-term.
- **Software:** module avail,
  module load <name>.
- **Slurm:** sbatch to submit;
  every job needs --time.
- **Prohibited:** heavy work on
  login nodes — use a compute job.

Real files are larger and per-cluster. This one is redacted to fit a slide.

WHY THIS PLACE

- **Cluster scope.** `/etc/agents.d/` is to a cluster what `~/.config` is to a user — every agent on the system inherits it.
- **Operator owns the contract.** Not the researcher, not the vendor. We write what RCAC means by `$SCRATCH`, by *partition*, by *quota* — once — and every agent absorbs it.
- **Vendor-neutral, mostly.** Claude Code, Warp, Cursor, Codex — each looks for rules files in slightly different places. Standardization is *emerging*; we track what each reads and write the same content into all of them.

Before the agent runs a single command, it already knows the rules of the road.

Leg 2a · rcac-mcp

Typed Slurm + filesystem + docs tools, over the user's existing **SSH**.

TOOL SURFACE

SLURM

- sbatch
- squeue
- sacct
- scancel
- sinfo

FILESYSTEM

- run_command
- read_file
- write_file
- upload_file
- download_file

RCAC

- myquota
- storage_paths
- jobinfo

The RCAC-isms agents otherwise guess at.

DOCS

- doc_search
- doc_load

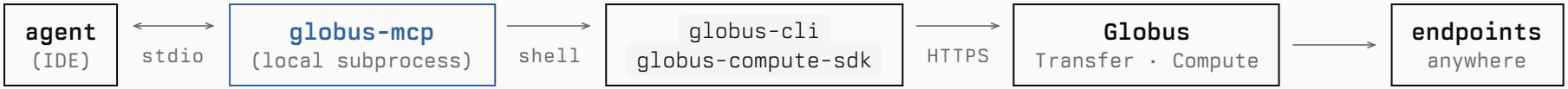
FTS5 over **400+** markdown files from RCAC-Docs. The agent grounds its advice *before* it answers.

EXECUTION MODES · `ssh` (default) · `local` (dev/test) · `delegate` (multi-user · JWT / OIDC)

If you can SSH, your **agent** *can too.*

Leg 2b · globus-mcp

The same **local-first** *shape — for Globus Transfer + Compute.*



ROADMAP · 26 / 36 TOOLS · THROUGH PHASE 5

- ✓ **1.** Core MVP — identity · endpoints · filesystem · transfer · tasks
- ✓ **2.** Extended operations — batch · delete · rename · task events
- ✓ **3.** Auth & consent — structured `ConsentRequired` → `session_consent(scopes)` → retry
- ✓ **4.** Globus Compute — submit · status · result + `uv` `worker_init` rendered in the server
- □ **5.** Bookmarks & convenience
- □ **6.** Advanced — collections · permissions · flows · timers · groups

WHAT IT BUYS YOU

▸ **Local-first.** Subprocess of the IDE over `stdio`. No hosted service, no new credentials, no new attack surface. `uvx git+…/globus-mcp` and it's running.

▸ `worker_init` **generalized.** The server renders a `uv`-based `worker_init` from the agent's `requirements` argument — hash-keyed, scratch-cached venvs on every Globus Compute endpoint, for free.

▸ **Consent recovery.** `ConsentRequired` comes back structured with the exact scopes; the agent calls `session_consent(scopes)` and retries. No shell gymnastics mid-workflow.

▸ **Beta.** Tool signatures may shift; rough edges (e.g. Python 3.14 Compute constraint) are tracked in the public roadmap.

Honest scope note: combining `rcac-mcp` & `globus-mcp` in one chat is currently rough — pick the server that owns the workflow's center of gravity.

The demo · interactive

Hey, run this n-body simulation on the cluster, bring the output back, and tell me what happened.

↳ `nbody-sim.py`

On it. I'll stage it under your scratch space — that gives us the bandwidth and quota for the snapshots. Which partition?

▸ STORAGE_PATHS() · MYQUOTA() · DOC_SEARCH("SCRATCH POLICIES") ✓ DISCOVERY

The standard `cpu` partition.

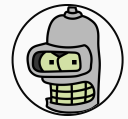
Got it. Verifying and staging.

▸ DOC_SEARCH("PARTITIONS CPU") · RUN_COMMAND("MKDIR …/NBODY-…") · UPLOAD_FILE(NBODY-SIM.PY) ✓ STAGED

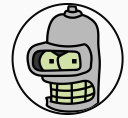
The demo · autonomous

> Submitting now. I'll watch it and ping you when results are back.

▸ WRITE_FILE(SUBMIT.SH, …) · SBATCH(SUBMIT.SH) → JOBID 41229884
▸ SQUEUE(JOBID=…) · JOBINFO(41229884) STATE: RUNNING → COMPLETED
▸ DOWNLOAD_FILE(OUT.H5) · RUN_COMMAND("PYTHON ANALYZE.PY OUT.H5") ✓ ANALYZED



> Job **41229884** ran for 11 m on `cpu`. Energy drift 3e-5; final two clusters at *r ≈ 1.4 / 0.9*. Snapshots are in `$CLUSTER_SCRATCH/nbody-2026-…/`. Want me to plot the trajectories?



One well-scoped MCP server: transfer, scheduler, filesystem, docs lookup, **verification**, *post-processing.*

Leg 3 · researcher-facing documentation

Treat agentic AI like any other tool in the HPC ecosystem.

docs.rcac.purdue.edu › ai agents

RCAC USER GUIDE

Using AI agents on RCAC clusters

01. **What is wired into the agent?**

02. **Verify before you trust**

03. **Report bad cluster advice**

No magic. No prohibition. Just the same contract we write for modules, queues, filesystems, and every other cluster tool.

WHAT'S WIRED IN

- `/etc/agents.d` baseline context
- `rcac-mcp` and `globus-mcp` tool surfaces
- the agent's view of storage, queues, and docs

HOW TO VERIFY

- `jobinfo`, `sacct`, log inspection
- sanity-check generated commands
- trust the docs, not the confident tone

HOW TO REPORT

- bad `sbatch` script? ticket it
- wrong filesystem advice? ticket it
- we want the failure modes visible

AI makes you faster at what you already **understand**, *not expert at what you don't.*

Lessons learned · PEARC'26

Long-horizon work only becomes agentic when you give it durable structure.

DECOMPOSE THE MONUMENT · FIRST PASS

1 HARNESS
manuscript repo · ACM template · build scripts

↓

2 CONCEPT
plan · notes · thesis alignment

↓

3 REFERENCES
research + align on citation scope

↓

4.1–4.13 SOURCE DIVES
independent context · fit · citation notes

↓

5 OUTLINE SCAFFOLD
section budgets · likely reference traces

↓

6.1–6.6 SECTION DRAFTS
outline-native prose with metadata + notes

↓

7 INTEGRATE → LATEX
first markdown → LaTeX assembly

HANDOFF → THEN LOOP

REVIEW → EDIT → INTEGRATE LOOP

↻ 2–3×

8a REVIEW
1+ hr recorded human critique

↓

8b TRANSCRIBE
annotated feedback map · concept IDs

↓

8c THEME
principal concepts → edit themes

↓

8d PATCH
in-place edits to outline materials

↓

8e INTEGRATE → LATEX
re-render outline → LaTeX

Repeated 2–3× to accepted manuscript; one more loop likely before camera-ready.

“Don’t cross the streams” · confinement

The risks aren't new — agents accelerate the **pace**, *not the magnitude.*

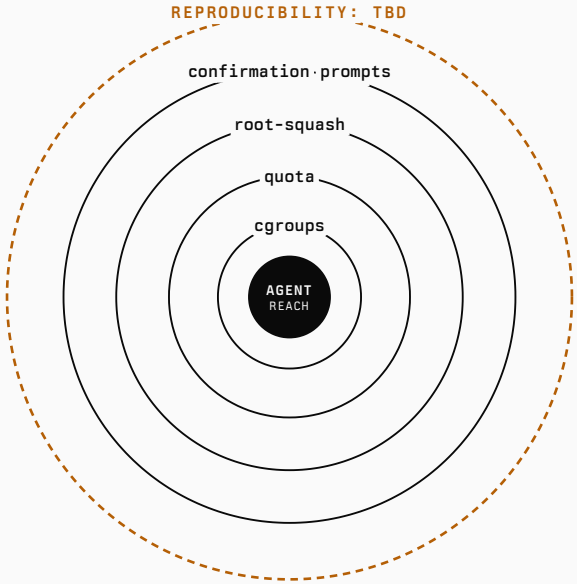
CANONICAL RISKS · BOUNDED BY…

- `rm -rf`
  an agent with shell access can wipe a project directory
  ↳ bounded by **root-squash + user-confirmation prompts**

- **allocation exhaustion**
  an agent managing jobs can burn through an allocation in hours
  ↳ bounded by **SU quotas + job-count limits + node health checks**

- **API-key leak**
  an agent reading env vars might log a credential into context
  ↳ bounded by **least-privilege creds + rotation**

BLAST RADIUS · EXISTING HARDENING



Mature centers already build confinement. The amber ring is the open question.

Questions we expect you to ask

All valid questions — whatever we don't get to is for the hallway track.

AGENT CONFINEMENT

Today agents act as the user — same UID, same SSH, same blast radius. Is there a future where the *agent* and the *human* are distinguishable identities with different access — and what does the transition look like?

MULTI-TENANT DEPLOYMENT

We operate a shared facility. Per-user MCP servers, or hosted with JWT / OIDC delegation? What's the migration path from local-first to facility-wide?

REPRODUCIBILITY

If an agent orchestrates a multi-step HPC workflow, how do I capture that for the *methods* section of a paper? What does "same result" even mean when the agent's choices are non-deterministic?

MIXING MCP SERVERS

You said `rcac-mcp` + `globus-mcp` is rough today — what does the path to clean composition look like? Tighter scoping, namespaces, an orchestrator on top?

END OF LINE

Thanks

Find me in the hallway, online, or on Slack.

rcac-mcp

purduercac/rcac-mcp

BETA · MIT

globus-mcp

purduercac/globus-mcp

BETA · MIT

Hello Computer · PEARC'26

glentner/pearc26-hello-computer

PAPER

PURDUE UNIVERSITY | Rosen Center for Advanced Computing

Geoffrey Lentner · glentner@purdue.edu

Midwest RCD CONSORTIUM