

The Genesis Mission: From Supercomputers to AI- Enabled Scientific Infrastructure

Ian Foster
Argonne National Laboratory
University of Chicago

DOE Genesis Mission • September 23, 2026



Topics

- Recent AI highlights
- Genesis Mission
- Discovery Environment
- Directions

AI highlights

EXISTENCE AND SMOOTHNESS OF THE NAVIER-STOKES EQUATION

CHARLES L. FEFFERMAN

The Euler and Navier-Stokes equations describe the motion of a fluid in $\mathbb{R}^n$ ($n = 2$ or 3). These equations are to be solved for an unknown velocity vector $u(x, t) = (u_i(x, t))_{1 \leq i \leq n} \in \mathbb{R}^n$ and pressure $p(x, t) \in \mathbb{R}$, defined for position $x \in \mathbb{R}^n$ and time $t \geq 0$. We restrict attention here to incompressible fluids filling all of $\mathbb{R}^n$. The *Navier-Stokes* equations are then given by

$$(1) \quad \frac{\partial}{\partial t} u_i + \sum_{j=1}^n u_j \frac{\partial u_i}{\partial x_j} = \nu \Delta u_i - \frac{\partial p}{\partial x_i} + f_i(x, t) \quad (x \in \mathbb{R}^n, t \geq 0),$$

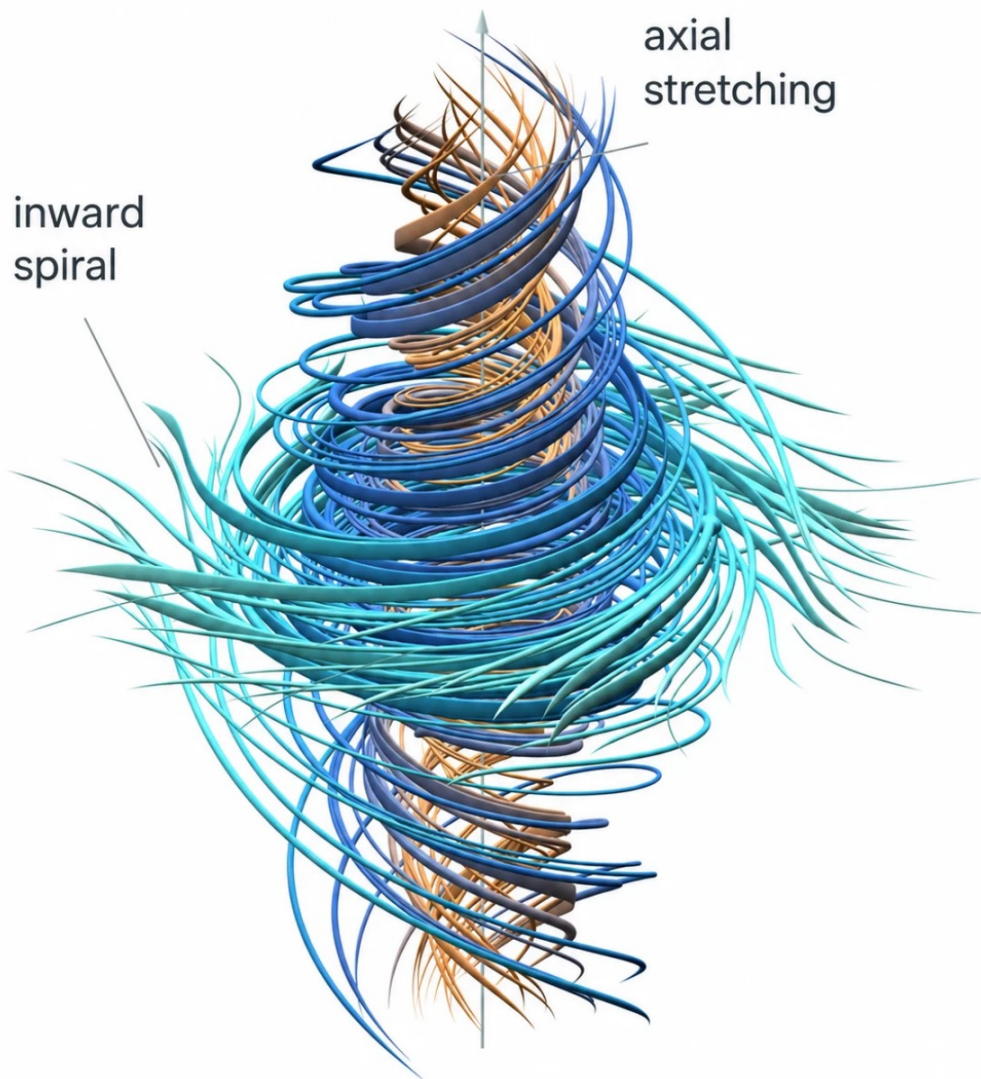
$$(2) \quad \operatorname{div} u = \sum_{i=1}^n \frac{\partial u_i}{\partial x_i} = 0 \quad (x \in \mathbb{R}^n, t \geq 0)$$

with initial conditions

$$(3) \quad u(x, 0) = u^\circ(x) \quad (x \in \mathbb{R}^n).$$

Here, $u^\circ(x)$ is a given, C^∞ divergence-free vector field on $\mathbb{R}^n$, $f_i(x, t)$ are the components of a given, externally applied force (e.g. gravity), ν is a positive coefficient

(the viscosity), and $\Delta = \sum_{i=1}^n \frac{\partial^2}{\partial x_i^2}$ is the Laplacian in the space variables. The *Euler equations* are equations (1), (2), (3) with ν set equal to zero.



In the process of resolving the Navier–Stokes problem, the agents sent 2.7 million messages and used approximately 130 billion output tokens.

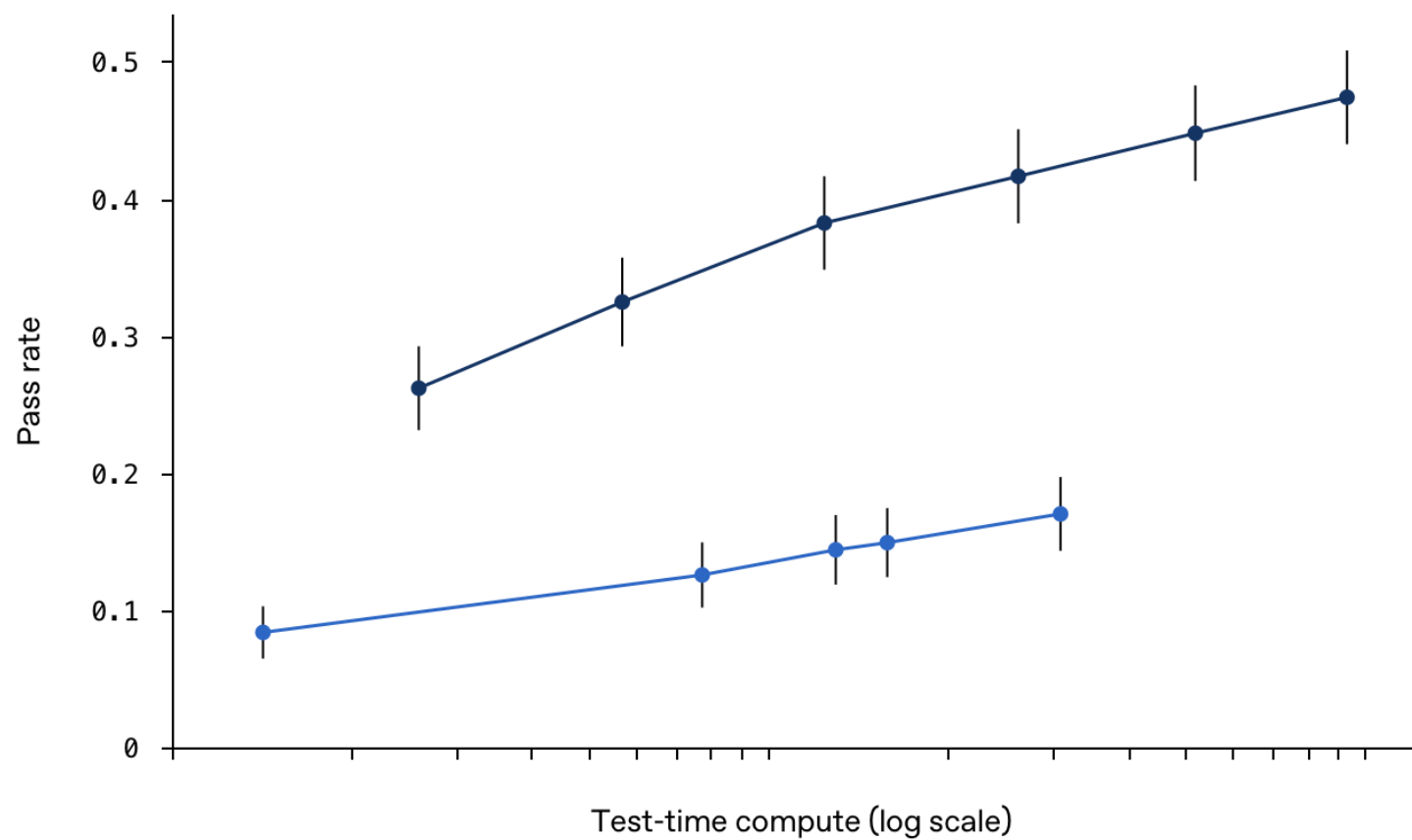
At \$50 per million output tokens

~\$6.5M of compute

Performance of GPT-6 Astra and our Internal Model on a curated set of open math problems



● GPT-6 Astra ● Internal Model



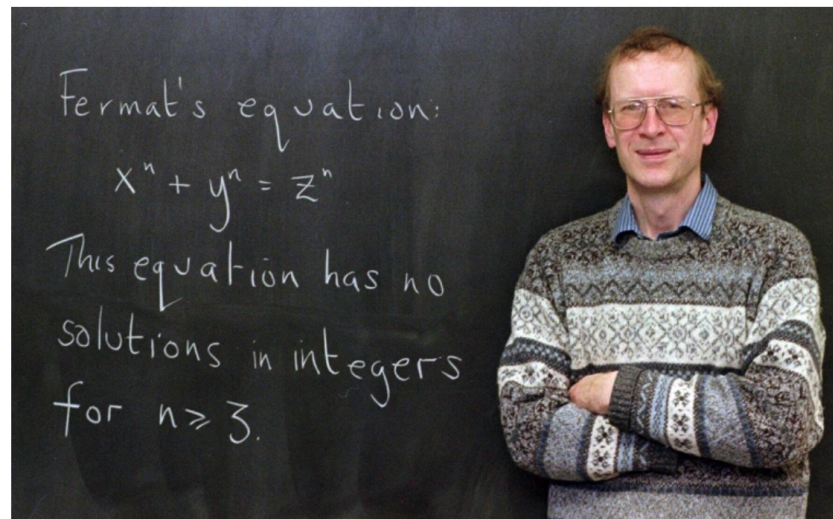
Anthropic AI 'formalizes' proof of Fermat's last theorem in just 11 days

Claude produced a 13-million-line, computer-checked proof of the famed conjecture — a major milestone in mathematics.

By [Davide Castelvecchi](#)



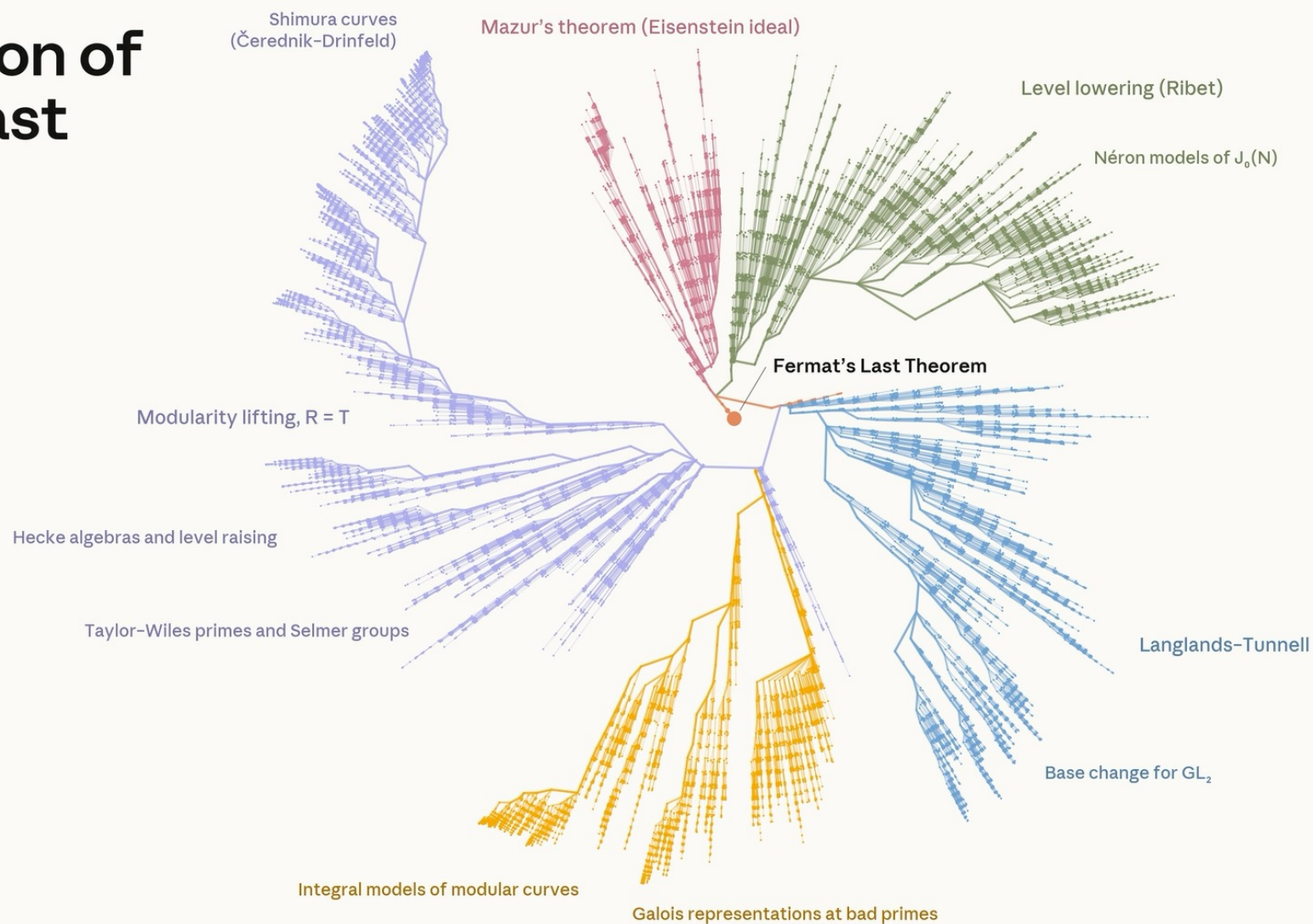
 [Save article](#)



Mathematician Andrew Wiles completed the proof to Fermat's last theorem in 1994, more than 350 years after Pierre Fermat proposed the conjecture. Credit: AP Photo/Charles Rex Arbogast/Alamy

Fermat's last theorem, one of the most celebrated mathematical results of the last half-century, has been turned into computer-verified code for the first time, using an advanced prototype of the artificial-intelligence (AI) chatbot Claude.

Formalization of Fermat's Last Theorem



OpenAI's GPT-6 Astra on ARC-AGI-3

Summary

- GPT-6 Astra scores 62.7% for \$26K on ARC-AGI-3 Semi-Private with our Standard harness, and 99.9% for \$19K with a Provider Adapter harness.
- GPT-6 Astra surpasses the human baseline in action efficiency on ARC-AGI-3. It used fewer actions than the median tested human on 96% of levels.
- A key behavior observed in GPT-6 Astra was its ability to turn unfamiliar environments into compact symbolic world models. It represented game mechanics as logical rules and developed its own domain-specific language shorthand to track state and plan actions.

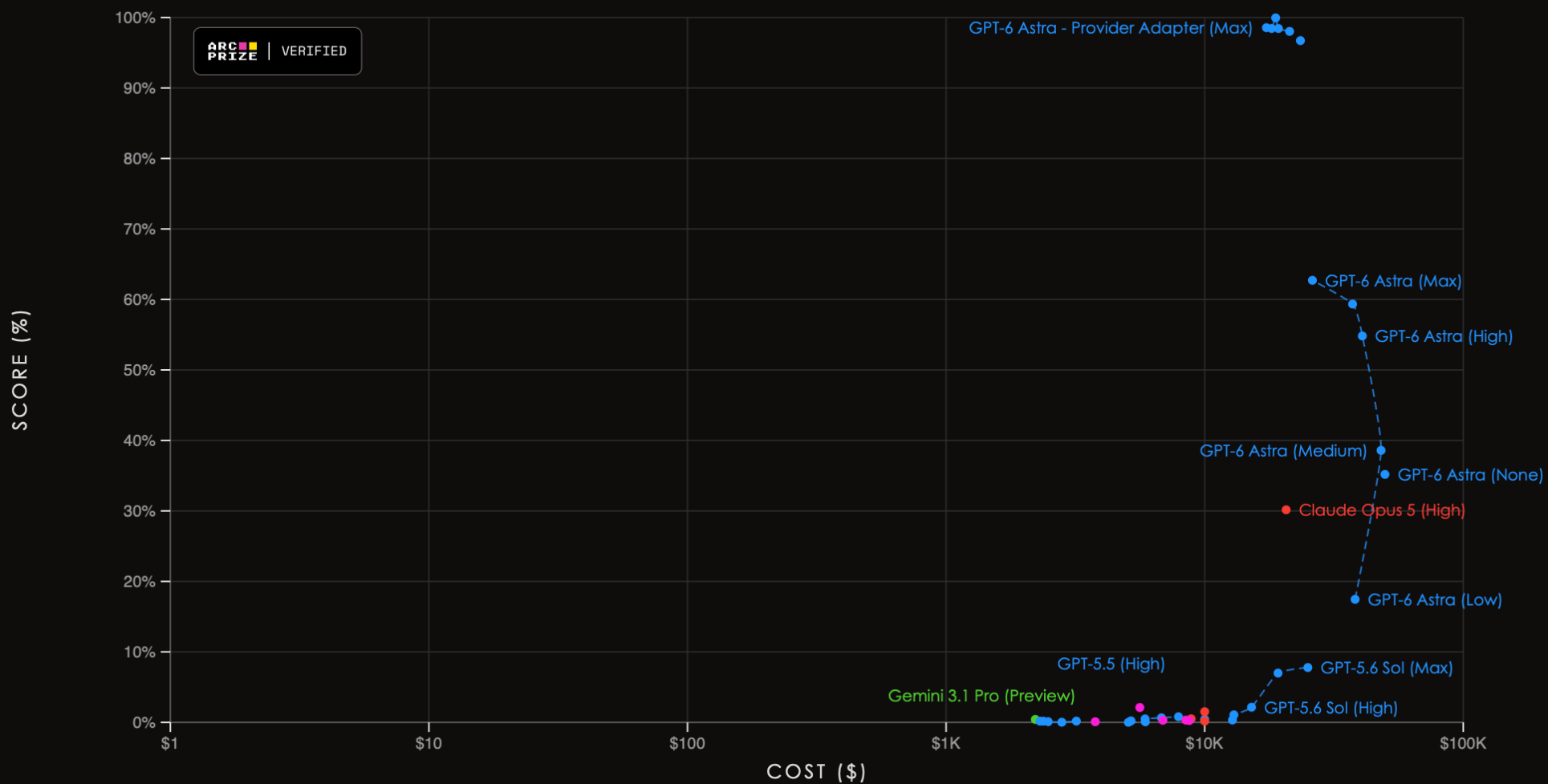
ARC-AGI-3

<https://arcprize.org/blog/astra>

ARC-AGI-3 is a benchmark for studying agentic intelligence through novel, abstract, turn-based environments. Agents must explore, infer goals, and build internal models of environments to effectively plan actions *without* explicit instructions. You can [play ARC-AGI-3 yourself](#).

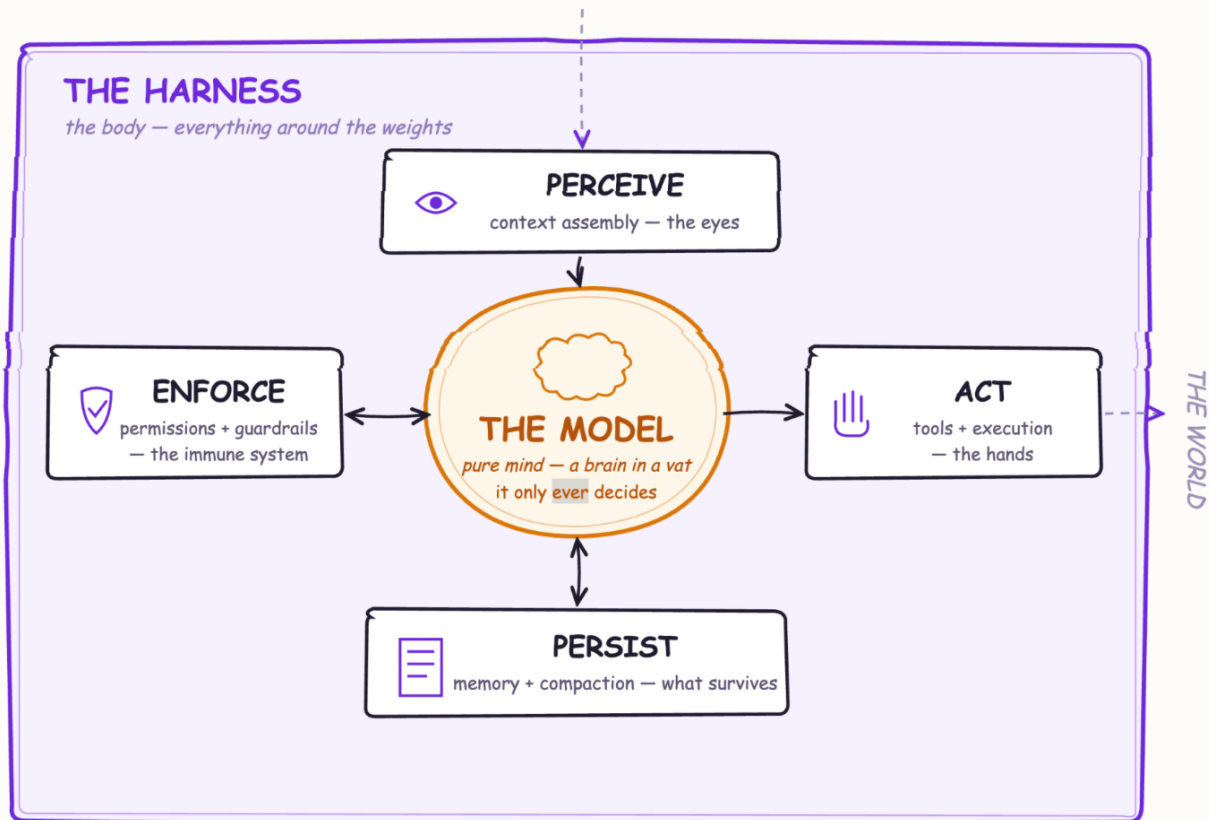


ARC-AGI-3 LEADERBOARD



The Anatomy of an Agent Harness

the harness gives the mind a body



perceive → decide → act → observe → repeat

The model only ever *decides*. Everything else is harness.

AI System

Task



Harness

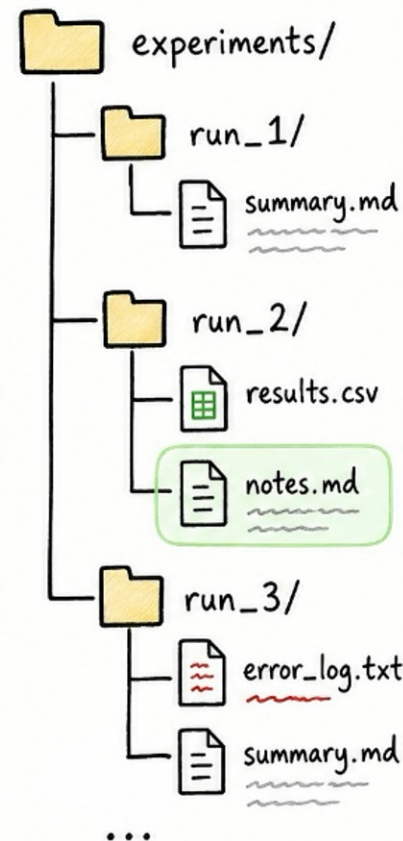


Model

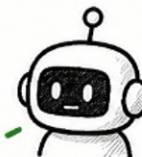
- Tool calls
- Memory
- Loops
- Evaluation
- State management

Long-horizon tasks need persistent memory

FILE SYSTEM MEMORY

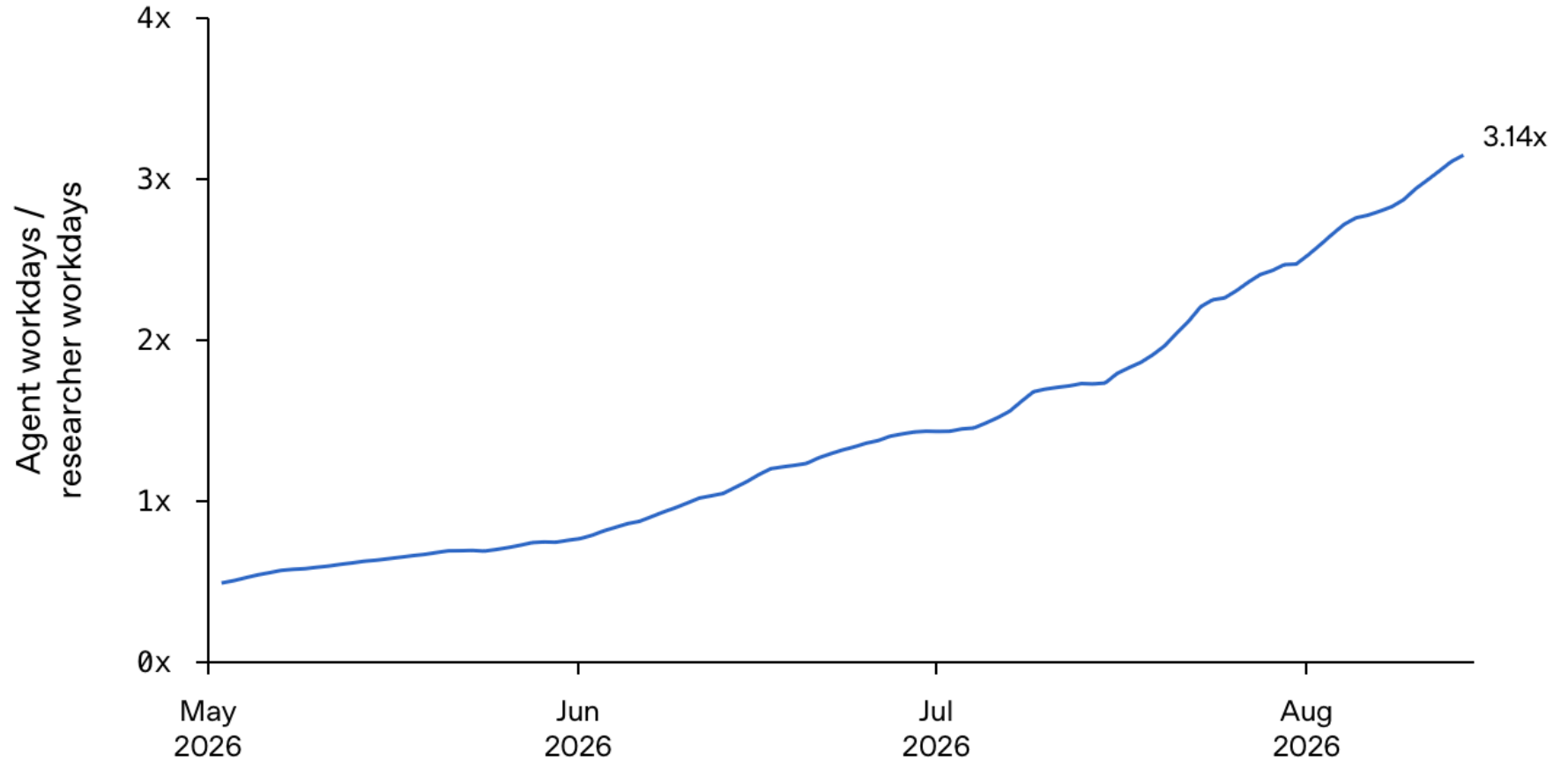


Agent reads only one small file



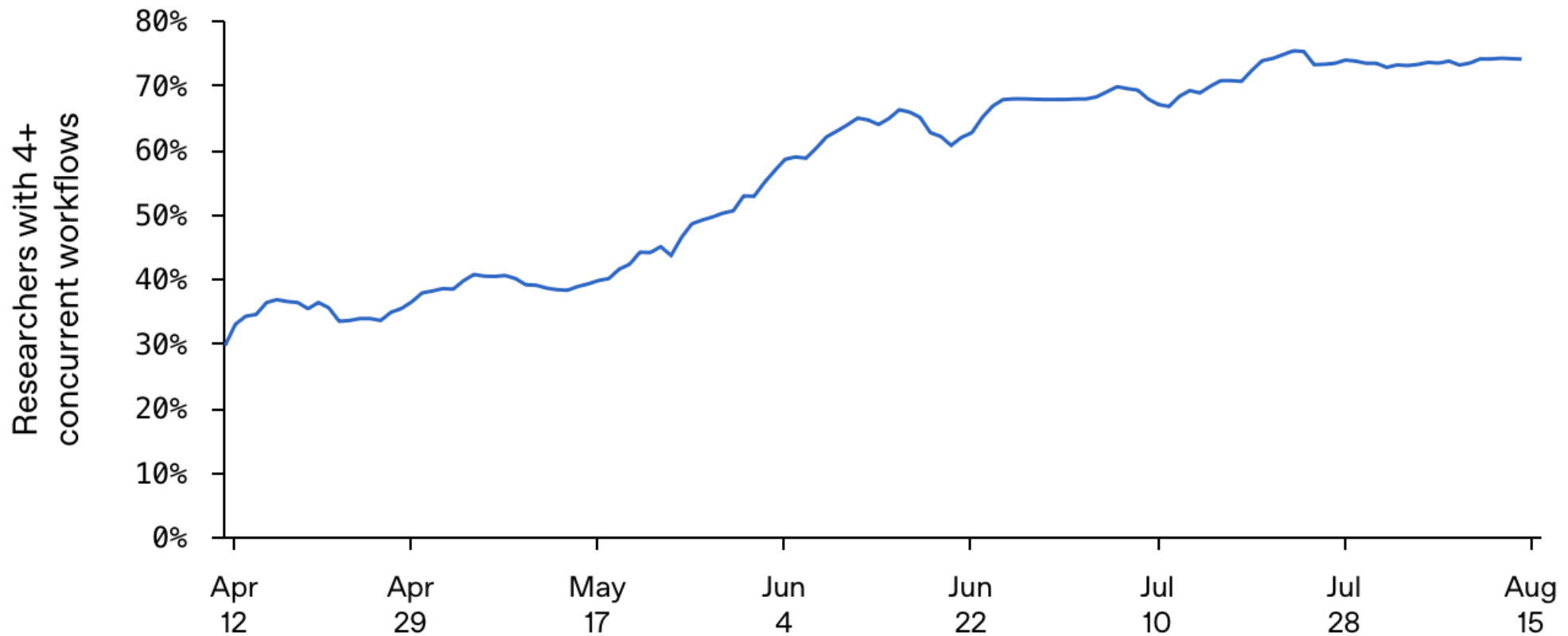
Clean context. Any step. ✓

Agentic workdays now far exceed those of human researchers



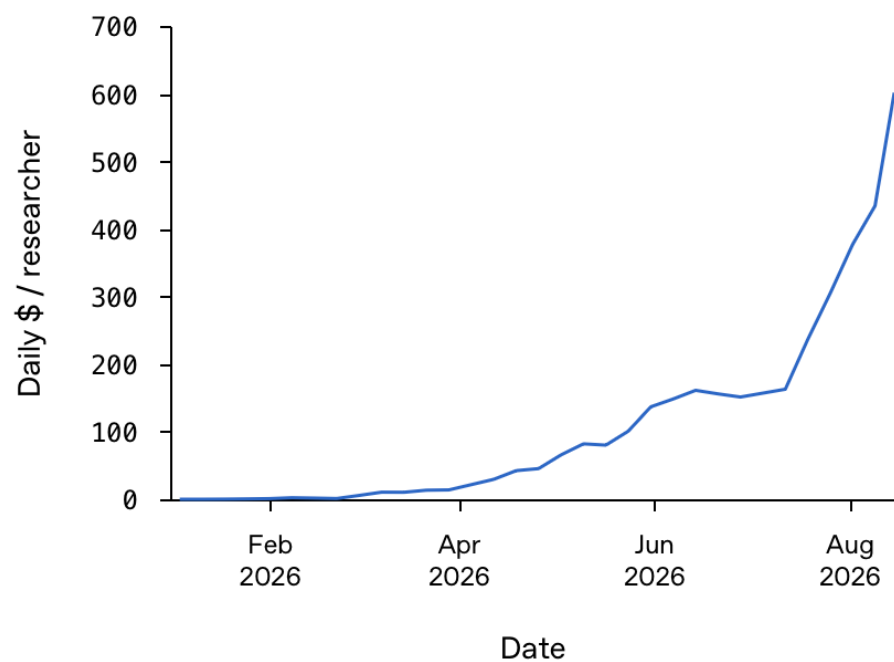
<https://openai.com/index/research-acceleration-view-inside-openai/>

Researchers are leveraging concurrent workflows more over time

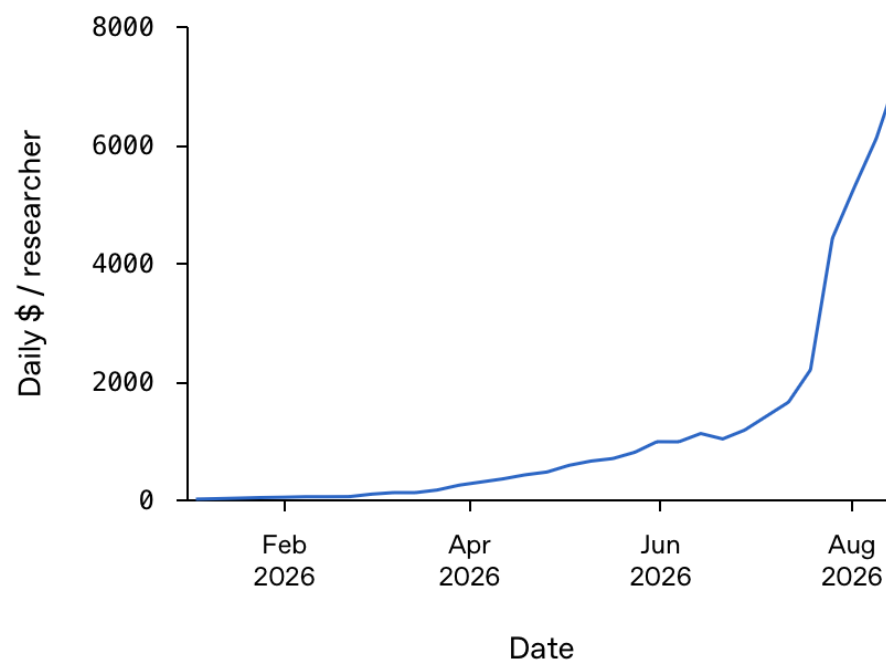


<https://openai.com/index/research-acceleration-view-inside-openai/>

Usage of internal coding agents is increasing significantly—Median researcher



Usage of internal coding agents is increasing significantly—90th percentile researcher



<https://openai.com/index/research-acceleration-view-inside-openai/>

We aim to safely build an automated AI researcher that can work under human supervision to further progress on deep learning and alignment, enabling iterative improvements. According to our measurements, we have now reached the goal, [announced](#) last fall, of having an automated research intern by September of this year.

By “research intern,” we mean a system that can carry out well-defined research tasks under human direction, including tasks that would take a skilled researcher a few days.

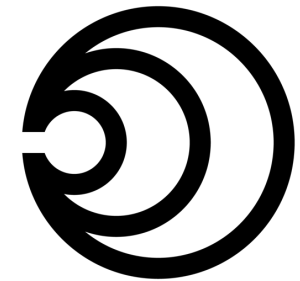
We are making strong progress toward creating an automated AI researcher by March of 2028.

Genesis Mission

U.S. Department of Energy

Genesis Mission

A national initiative to accelerate
science through artificial intelligence



Goal: double research productivity and impact within a decade

Energy dominance / Discovery science / National security

National science and technology challenges

Energy systems

Nuclear energy deployment and an electric grid that can meet growing demand.

Biotechnology and health

Biological discovery and the root causes of chronic disease.

Strategic technologies

Domestic microelectronics and U.S. capabilities in space.

Materials and manufacturing

Critical minerals, materials with predictable properties and industrial productivity.

Fundamental science

Physics across scales, quantum algorithms and particle accelerators.

Scientific infrastructure

Autonomous laboratories and improved design and operation of infrastructure.

The American Science and Security Platform

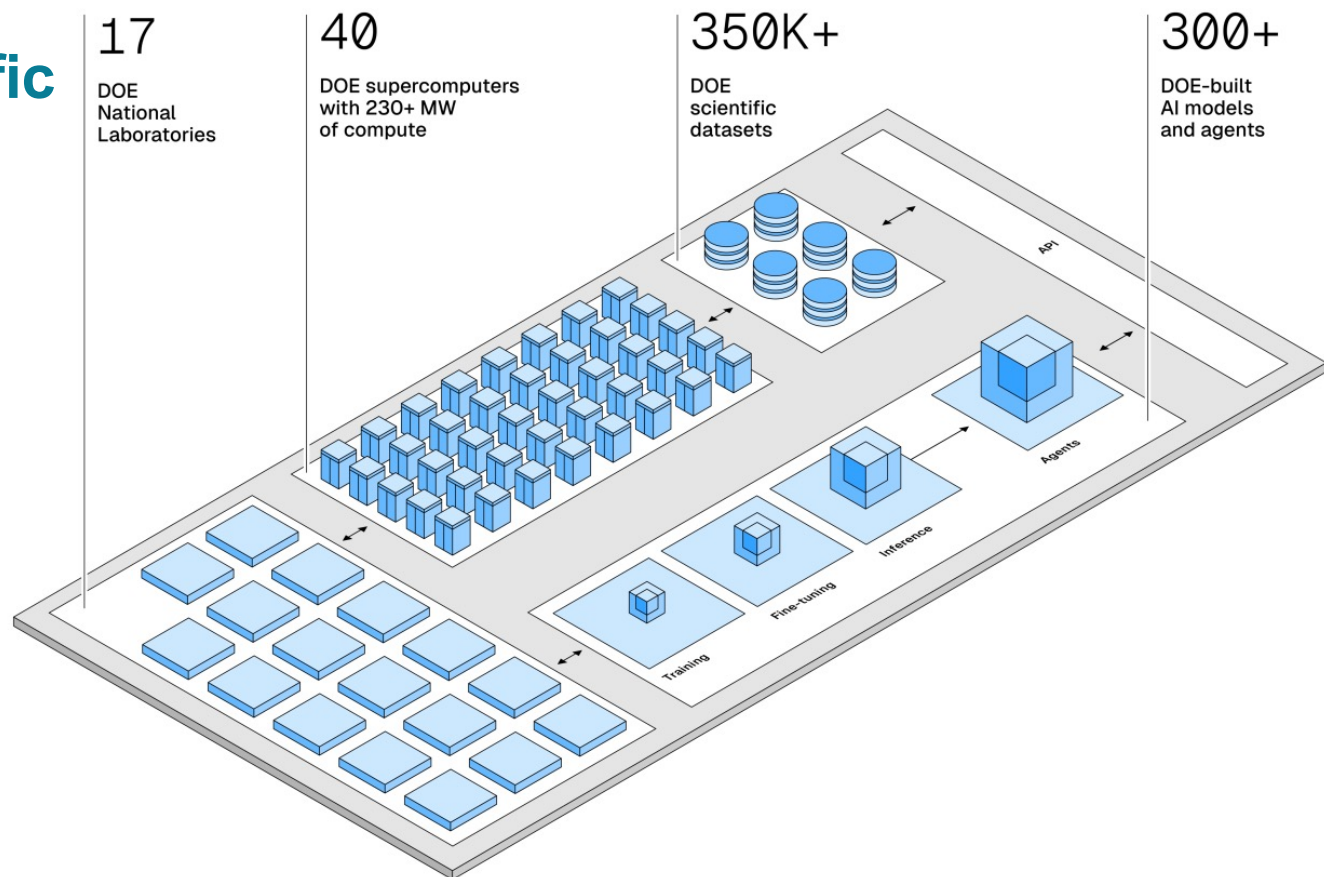
A coordinated scientific discovery system

Supercomputing and cloud

Curated scientific data

AI models and agents

Experimental and production facilities



DOE platform inventory graphic • April 2026

A national collaboration organized for delivery

DOE and NNSA teams

Data

Curated, traceable scientific resources

Infrastructure

Connected computing and facilities

Models

AI tools for hypotheses and experiments

National challenges

Research directed at national priorities

Public-private partnerships

Shared expertise and resources

Genesis Mission Consortium

17 national laboratories connect with industry, academia, nonprofits and philanthropy.

Working groups

Models and validation; data standards; computing and cloud; robotics; workforce training.

Genesis Mission industry partners

Companies named in the consortium directory or DOE's signed-MOU announcement

Accenture Federal Services	Domino Data Lab	OpenAI
AMD	Dow	Oracle†
Amazon Web Services	DrivenData	Palantir
Anthropic†	Edison Scientific	Periodic Labs
Arcee AI	Emerald AI	Prometheus
Armada†	Essential Knowledge Systems	Radical AI†
Astris AI	Google Public Sector	Reflection
Cerebras Systems	Groq†	SambaNova Systems
Cognition	HPE	Scale AI
CoreWeave	IBM	Siemens
Cornelis Networks	Intel	Undermind
Databricks	Medra	Veritone
Dataerai	Microsoft	Wiley
Dell Technologies	NVIDIA	xAI†

† Signed MOU announced December 18, 2025; not listed in the current consortium directory.

U.S.–Japan Genesis Mission partnership

June 4, 2026 • Japan becomes the mission's first international partner

\$1 billion

Planned over five years
\$500 million from each nation

Research scope

Quantum information science, fusion, biotechnology, advanced materials, particle physics and autonomous labs.

11 joint scientific teams

12 DOE national laboratories
1 DOE science user facility
12 Japanese research institutions

Shared computing and facilities

DOE supercomputers and Japan's Fugaku; reciprocal computing access for joint research on the same terms as domestic researchers.

DOE–MEXT–METI Statement of Intent • Planned investment is subject to future appropriations.

Announced progress in 2026

278

projects selected

July 22 • Teams span 342 institutions and will develop AI-enabled scientific workflows.

\$800M+

partner commitments

July 22 • Computing, models, expertise and funding committed through the GM consortium.

Open models

initiative launched

August 7 • Scientific open-weight models planned, beginning with Genesis-Science-1.

Selections require negotiation. Commitments include in-kind support. Planned models are not confirmed releases.

278 projects expose a common opportunity

278

selected projects

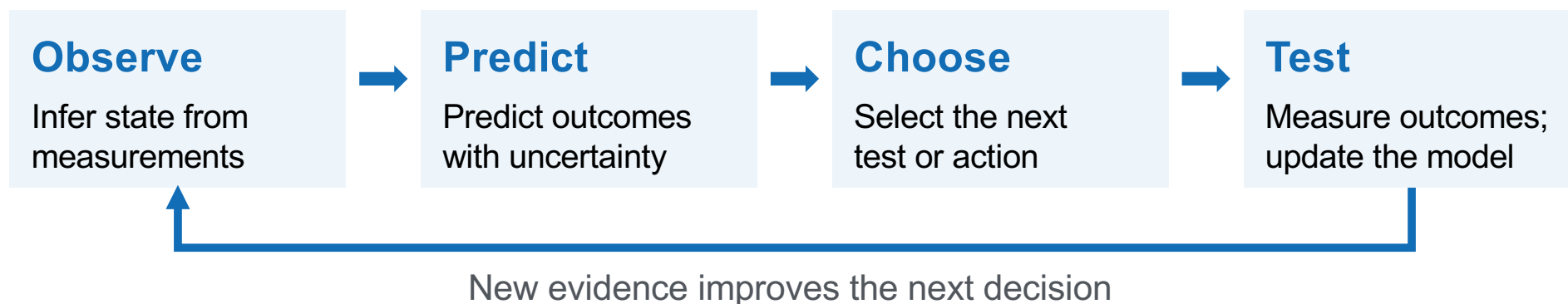
13 science themes
One opportunity to make
capabilities reusable

Materials / chemistry	29
Microelectronics	17
Quantum information	23
Fusion / plasma	20
Fission / fuel / waste	17
HEP / nuclear / cosmology	29
Accelerators / labs	12
Earth / atmosphere / water	28
Subsurface / geothermal	18
Critical minerals	20
Grid / energy / CFD	18
Trustworthy AI / HPC	27
Biology / biomanufacturing	20

THE UNIFYING IDEA

The common unit is a learning “discovery” loop

Projects repeatedly connect observation, prediction, decisions and new evidence.



105 / 278

projects close a hardware loop

114 / 278

require language-model reasoning

Source: covering-set analysis; classifications may overlap.

Materials discovery must end in a recipe

The scientific bridge is composition → structure → process → performance.

Interstitial alloys

Predict how oxygen changes strength and ductility; connect atomistic screening to melting and microscopy.

294741 • LBNL / LLNL / UC Davis

AIM-MAG

Link processing, microstructure and magnetic properties in heavy rare-earth-free Nd–Fe–B magnets.

298135 • RPI / Ames / GE Vernova

AlphaFilm

Resolve nucleation and growth bottlenecks in sputtered AlScN thin films through closed-loop synthesis.

299142 • AlphaFilm

Shared need: atomistic prediction + measurement + process history.

Projects 294741, 298135, 299142 • Proposed workflows

Co-design crosses devices and algorithms

Continuous physics and discrete design must constrain each other.

CMOS + X

Couple cryogenic electronics, superconducting circuits and thermal fields in one design workflow.

AutoQEC

Search error-correcting codes jointly with hardware connectivity, noise and circuit implementability.

PRISM-Q

Connect processor layout, fabrication, loss mechanisms and calibration in superconducting arrays.

Shared need: typed graphs, physical surrogates and exact verifiers.

CONTEXT

ModCon: Transformational AI Model Consortium

Needs come in from four directions. Broadly useful outputs go out to production through AmSC. ModCon is the R&D engine in between.

INPUTS & NEEDS

278 RFA teams
25 yrs of DOE lab work
Users & communities
Internal challenges

MODCON — R&D SPEED

Define, prototype, evaluate

- Defines the services, models and data
- X-Lab / skunkworks agility: move fast, iterate, prototype
- Deploys and evaluates prototypes with real users first
- Big challenge problems drive progress
- Infrastructure carve-out for rapid R&D

AMSC — PRODUCTION SPEED

Host, scale, sustain

- Operates production infrastructure
- Hosts artifacts and workspaces
- Runs large agentic jobs at scale
- Productionizes broadly useful outputs

A two-speed model: R&D in ModCon, production via AmSC. Neither is a blocker to the other.

Nuclear systems need coupled prediction

A fast component model has value only when the coupled system remains credible.

Fusion core–edge	Fuel recycling	Fuel qualification
REACT couples core transport and edge exhaust surrogates to study performance while protecting divertor surfaces.	An oxide-reduction digital twin joins process physics, experimental data, anomaly detection and documentation.	Bayesian calibration propagates shared mechanistic uncertainty into engineering predictions and experiment selection.
294879 • REACT	295104 • ANL-led	295732 • LANL-led

Shared need: multi-fidelity models with uncertainty that propagates.

Facilities must learn from raw signals

Preserve information that early compression can discard.

Multimodal physics	FM4NPP joins tracking and calorimeter information; validate with particle identification and jet observables.
Transferable tracking	Learn directly from raw signals across TPC, GEM and straw-tube technologies, then reconstruct tracks.
Self-learning machines	Accelerator twins infer hidden machine state and recommend informative actions under uncertainty.

Share representation methods; keep detector and machine tests distinct.

Discovery Environment

The scientific discovery cycle

Implementation synthesis based on DOE's closed-loop vision

01 Frame the question

Set the scientific objective, constraints and success criteria.

02 Assemble evidence

Retrieve relevant observations, theory and prior experiments.

03 Propose and predict

Generate hypotheses and estimate which tests will teach the most.

04 Run test

Execute approved simulations or physical experiments.

05 Validate result

Compare prediction with measurement and quantify uncertainty.

06 Update and repeat

Preserve the evidence and use it to choose the next question.

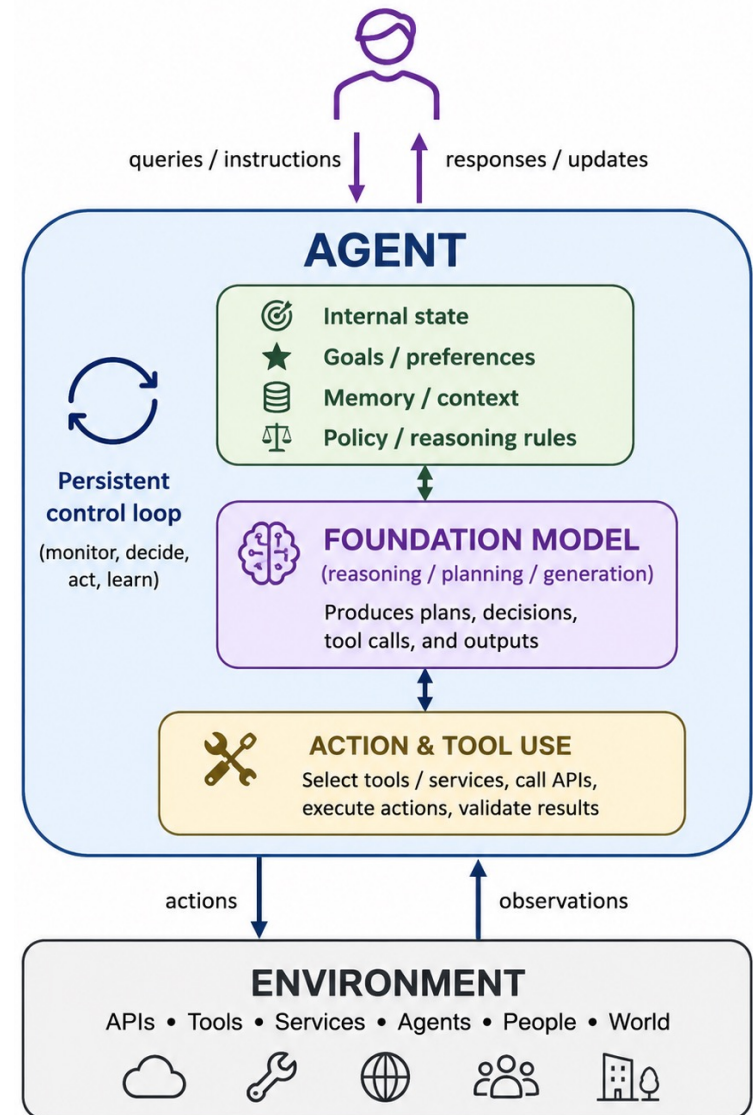
What is an “agent”?

An agent is a persistent, stateful process that acts on behalf of a user or system. An agent may:

- **Observe** inputs or events [or user query]
- **Plan** actions using a policy (rules or model)
- **Act**: Execute tools or call other agents
- **Learn**: Update state to adapt over time

A **scientific agent** operates in a **scientific ecosystem** to meet **scientific goals**

We need machinery to allow scientific agents to operate effectively in the DOE ecosystem



Common challenges across projects

- Invoking LLMs and parsing responses from arbitrary models
- Scaling LLM inference on HPC systems
- Accessing resources in a federated DOE ecosystem
- Scalable tool calling: MCP, Code, Skills
- Identity, resource rights and management
- Maintaining and managing agentic memory
- Provenance, observability, reproducibility, human oversight

Agentic systems today are not scalable, federated, & infrastructure-aware

Common needs across diverse applications

THINK

Interact with Models

- 1 Use diverse LLMs and foundation models; manage prompts, context, and structured outputs.

REMEMBER

Manage Memory & State

- 2 Maintain context, persistent memory, scientific state, artifacts, and knowledge across executions.

PLAN

Plan & Orchestrate

- 3 Decompose goals; coordinate tasks, agents, and workflows; manage dependencies, retries, and long-running work.

ACT

Act on the World

- 4 Invoke scientific tools, data services, compute resources, APIs, and physical instruments.

OPERATE

Deploy & Operate Agents

- 5 Run agents from edge to exascale; federate, scale, monitor, steer, pause, and recover.

TRUST

Govern, Verify & Track

- 6 Manage identity, authorization, policy, evaluation, provenance, lineage, and auditability.

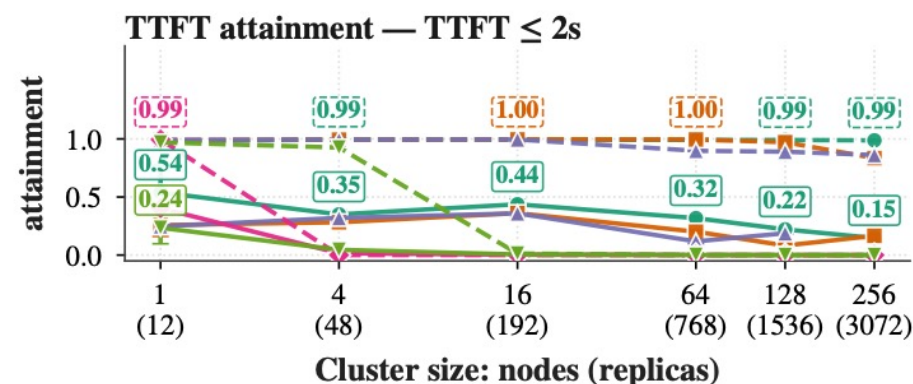
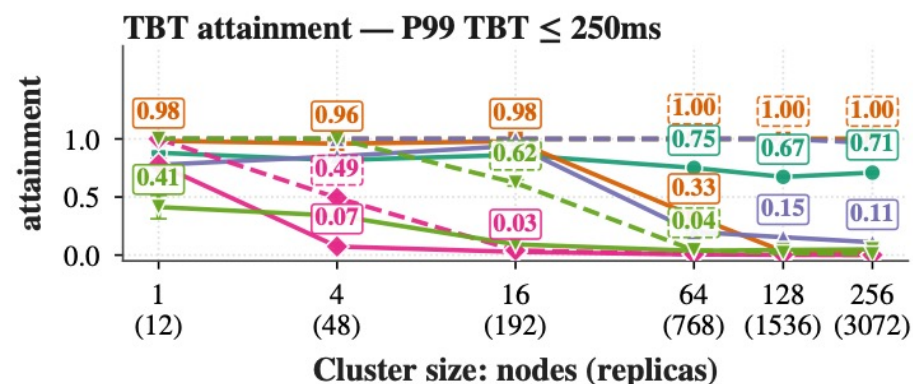
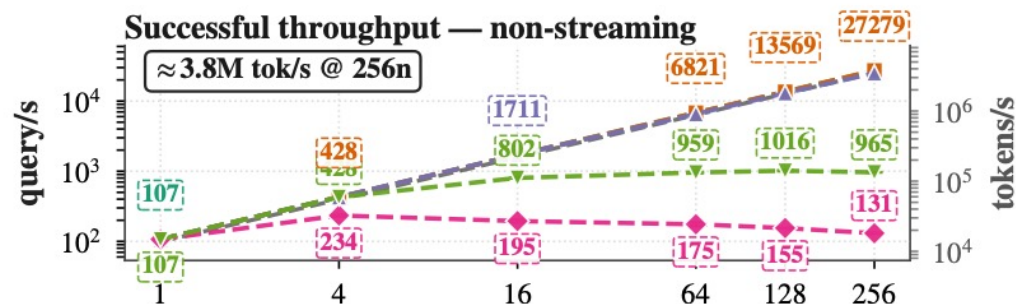
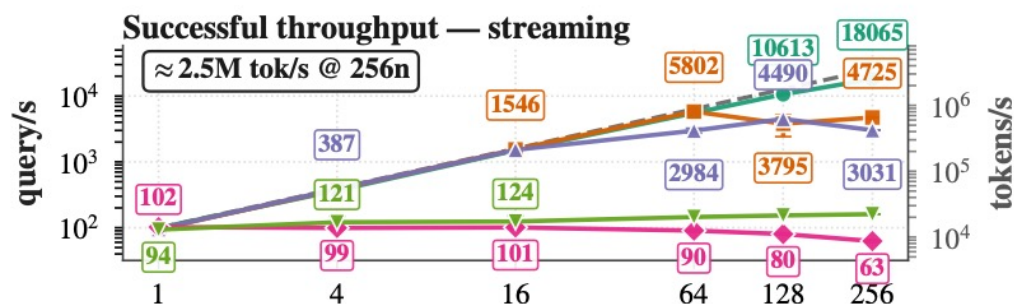
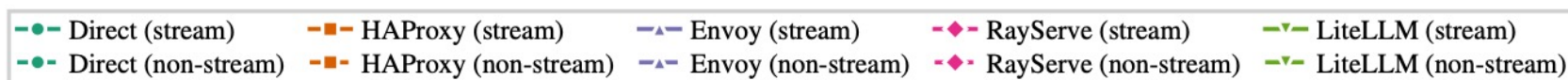
THINK → **REMEMBER** → **PLAN** → **ACT** → **OPERATE** → **TRUST**

ExaServe: LLM Serving on Exascale Systems



Wenyi Wang

Proxy scaling from 1 to 256 nodes (Llama-3-8B, 64 in / 64 out, 110 QPS per node offered).



<https://github.com/globus-labs/exaserve>

TTFT = Time To First Token; TBT = Time Between Tokens

Common needs across diverse applications

THINK

Interact with Models

- 1 Use diverse LLMs and foundation models; manage prompts, context, and structured outputs.

REMEMBER

Manage Memory & State

- 2 Maintain context, persistent memory, scientific state, artifacts, and knowledge across executions.

PLAN

Plan & Orchestrate

- 3 Decompose goals; coordinate tasks, agents, and workflows; manage dependencies, retries, and long-running work.

ACT

Act on the World

- 4 Invoke scientific tools, data services, compute resources, APIs, and physical instruments.

OPERATE

Deploy & Operate Agents

- 5 Run agents from edge to exascale; federate, scale, monitor, steer, pause, and recover.

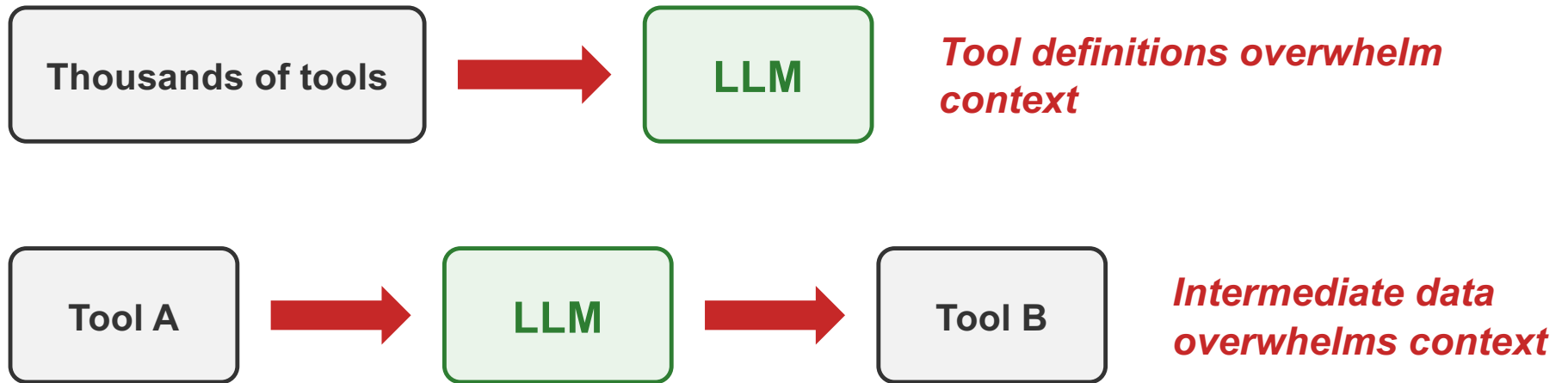
TRUST

Govern, Verify & Track

- 6 Manage identity, authorization, policy, evaluation, provenance, lineage, and auditability.

THINK → **REMEMBER** → **PLAN** → **ACT** → **OPERATE** → **TRUST**

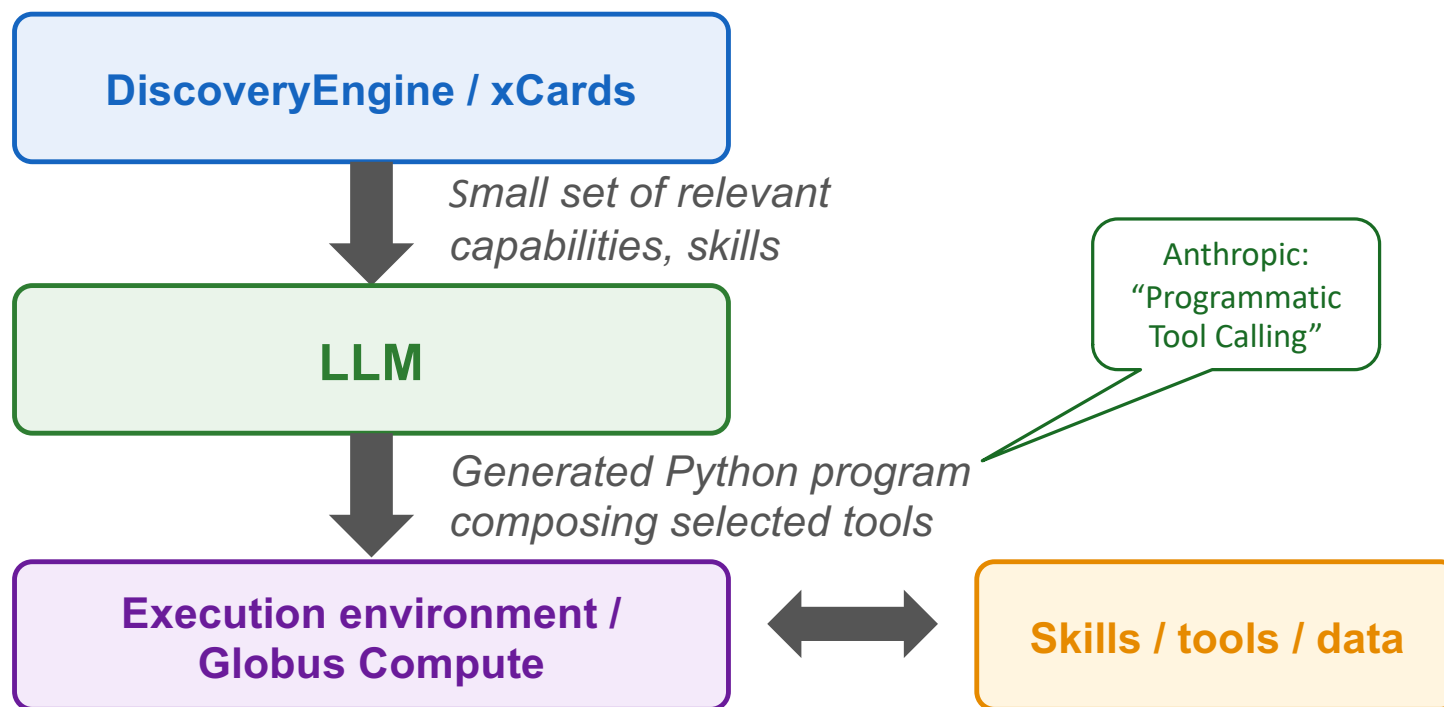
The problem with naïve tool calling (e.g., via MCP)



The LLM becomes both the catalog and the data bus; neither scales.

<https://www.anthropic.com/engineering/advanced-tool-use>

Separate discovery, reasoning, and execution



LLM chooses and composes;
skills guide LLM and provide relevant code;
generated Python means 1 program not N tool calls;
execution substrate moves data and maintains state.

Genesis Skills



Portable AI-agent workflows for science and engineering

Genesis Skills is a collection of reusable [Agent Skills](#) for scientific discovery and engineering workflows. Part of the [Genesis Mission](#), the catalog can be used with Claude Code, Codex, Cursor, OpenCode, Gemini CLI, and LangChain Deep Agents. Individual skills may still depend on client-specific features or external tools.

Skills encode domain expertise, connect to scientific data sources, and provide reproducible patterns for computational research.

What Are Skills?

Skills are markdown files (`SKILL.md`) with YAML frontmatter that teach an AI agent how to handle domain-specific tasks. Each skill provides:

- **Domain expertise** — Specialized knowledge for literature, HPC, plasma simulation, scientific-data metadata, and AI-system evaluation
- **Tool access** — Permissions to use specific tools (Bash commands, file operations, web APIs)
- **Structured workflows** — Step-by-step instructions for complex scientific tasks
- **Supporting resources** — Scripts, templates, and reference materials the agent can use

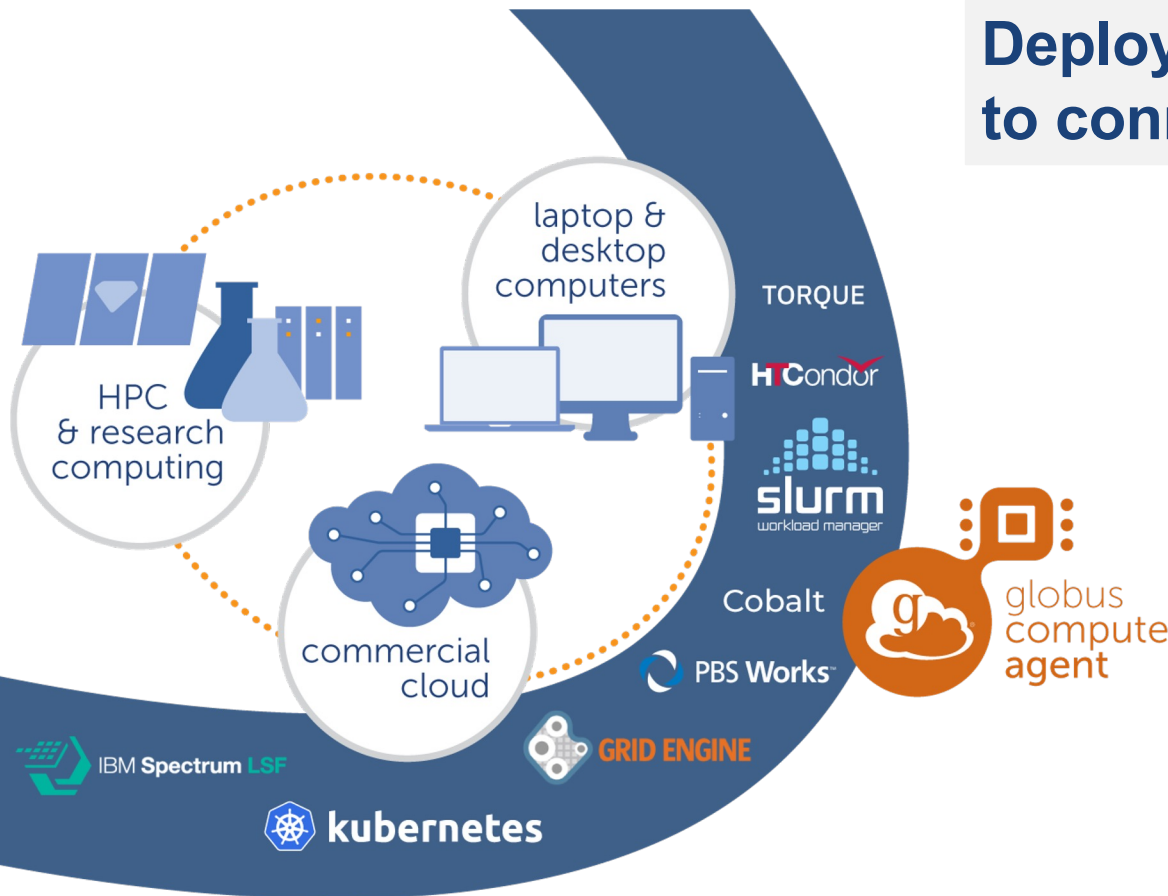
Agents can select a skill from a natural-language request. Client-specific menus and explicit invocation shortcuts vary by agent.

<https://github.com/Al-ModCon/genesis-skills>



Globus Compute multi-user endpoints

Deploy Globus Compute Software to connect to ecosystem



- Asynchronous Python task execution
- Persistent pilot that enables interactive (back-and-forth) execution
- Environment management
- Local policy enforcement
- Standard Globus identity mapping for authorization

<https://www.globus.org/compute>

Addressing tool calling challenges at DOE scale

- **Scalable code execution:** Execute agent-generated Python securely near data, instruments, and HPC resources; reuse environments and state rather than routing intermediate data through the model → **IRI + Globus Compute** (*Python-native execution, pilot jobs, environment/state management*)
- **Reusable skills:** Package domain expertise (e.g., simulation calibration procedure) as tested, composable capabilities rather than repeatedly reconstructing workflows → **Genesis Skills Repository**
- **Discovery and selection at scale:** Allow agents to choose among thousands of instruments, services, APIs, skills, and agents across a federation → **DiscoveryEngine + AgentCards + Cairn** (*capability discovery + reputation*)
- **Reliable tool use:** Detect silent failures, partial results, degraded services, and unexpected behavior; provide agents with actionable execution feedback → **Monitoring + observability**
- **Persistent execution state:** Preserve datasets, models, sessions, and intermediate results across agent/program invocations to avoid repeated transfer and computation → **Globus Compute** (*execution contexts / state handles*)

Common needs across diverse applications

THINK

Interact with Models

- 1 Use diverse LLMs and foundation models; manage prompts, context, and structured outputs.

REMEMBER

Manage Memory & State

- 2 Maintain context, persistent memory, scientific state, artifacts, and knowledge across executions.

PLAN

Plan & Orchestrate

- 3 Decompose goals; coordinate tasks, agents, and workflows; manage dependencies, retries, and long-running work.

ACT

Act on the World

- 4 Invoke scientific tools, data services, compute resources, APIs, and physical instruments.

OPERATE

Deploy & Operate Agents

- 5 Run agents from edge to exascale; federate, scale, monitor, steer, pause, and recover.

TRUST

Govern, Verify & Track

- 6 Manage identity, authorization, policy, evaluation, provenance, lineage, and auditability.

THINK → **REMEMBER** → **PLAN** → **ACT** → **OPERATE** → **TRUST**

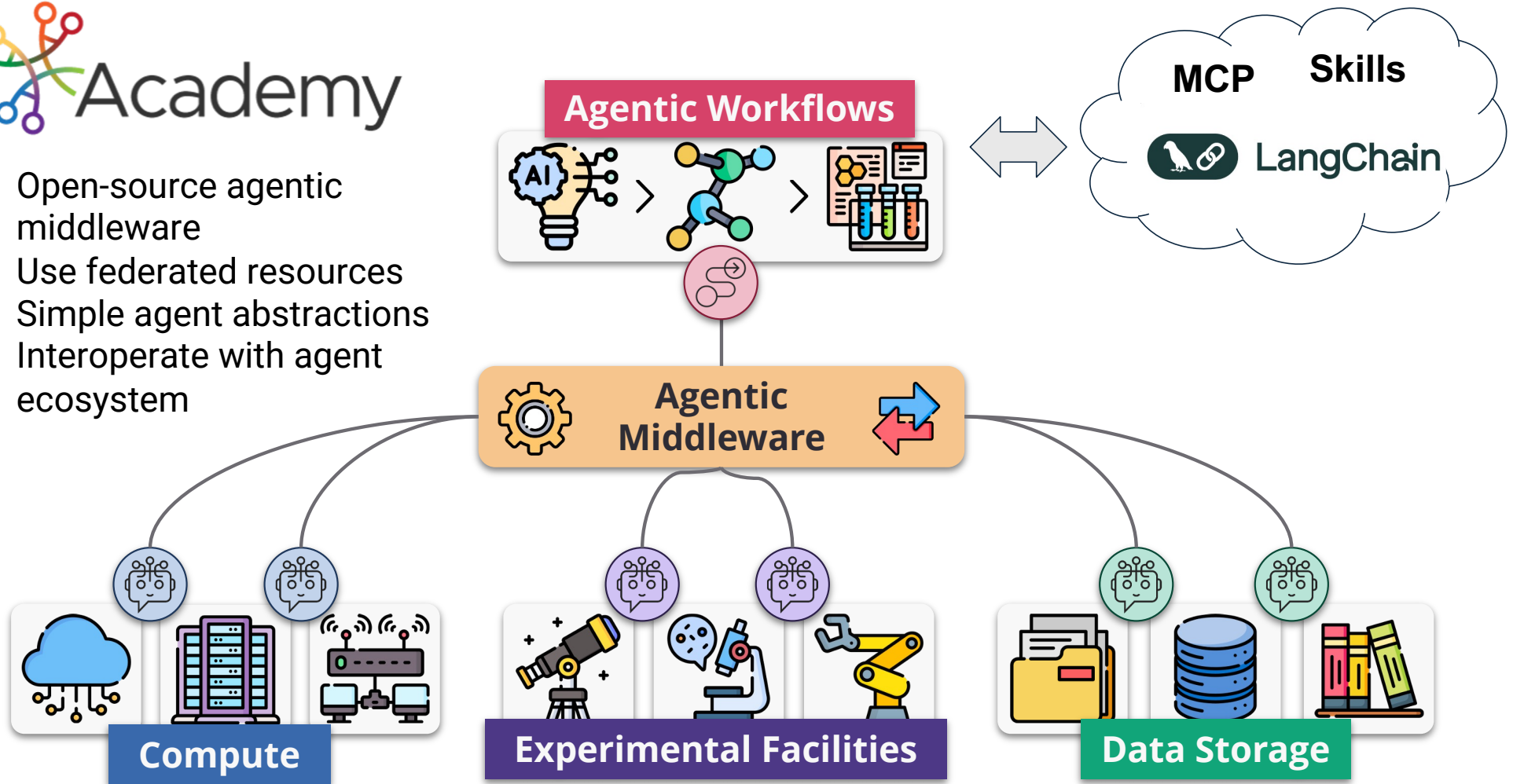
Deploying and operating agents: Why federation?

- **Science is inherently distributed:** Instruments, data, models, and compute live at different facilities, in different administrative domains
- **Data gravity & sovereignty:** Many datasets cannot move; agents must go to the data, and respect local policy and governance
- **Specialization & capability:** The right resource for each step is rarely in one place; leadership HPC, AI accelerators, beamlines, robotic labs
- **Long-running campaigns span sites:** Simulate at ALCF, analyze at NERSC, run the experiment at a light source; one agent team, many facilities
- ***Thus agents must be deployable, discoverable, persistent, and coordinated across facilities***

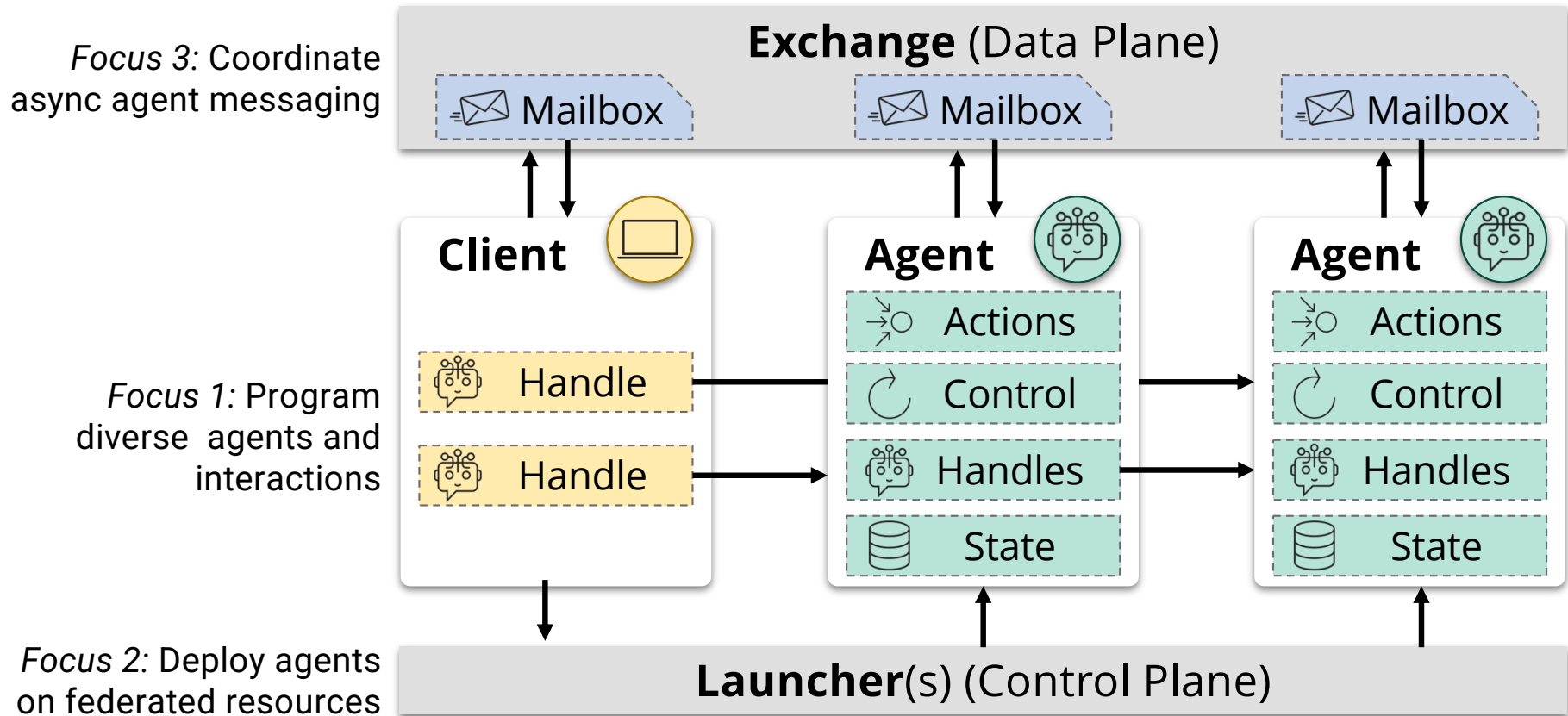
No mainstream agent framework supports this; this is the gap that Academy fills



- Open-source agentic middleware
- Use federated resources
- Simple agent abstractions
- Interoperate with agent ecosystem



Academy architecture



<https://docs.academy-agents.org/latest/concepts/>

Anatomy of an Academy agent



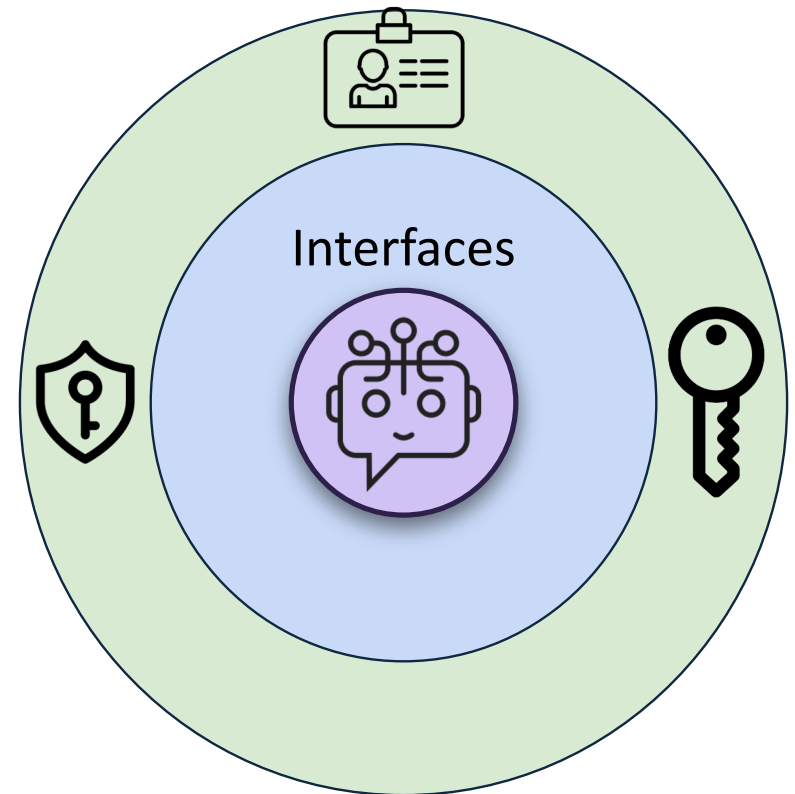
Core agent code (local or remote LLM, interface to prompt, interpret responses)



Wrapped with a set of interfaces that enable deployment, lifecycle management, communication, etc...



Wrapped with a unique identity and permissions/tokens to perform a set of authorized actions



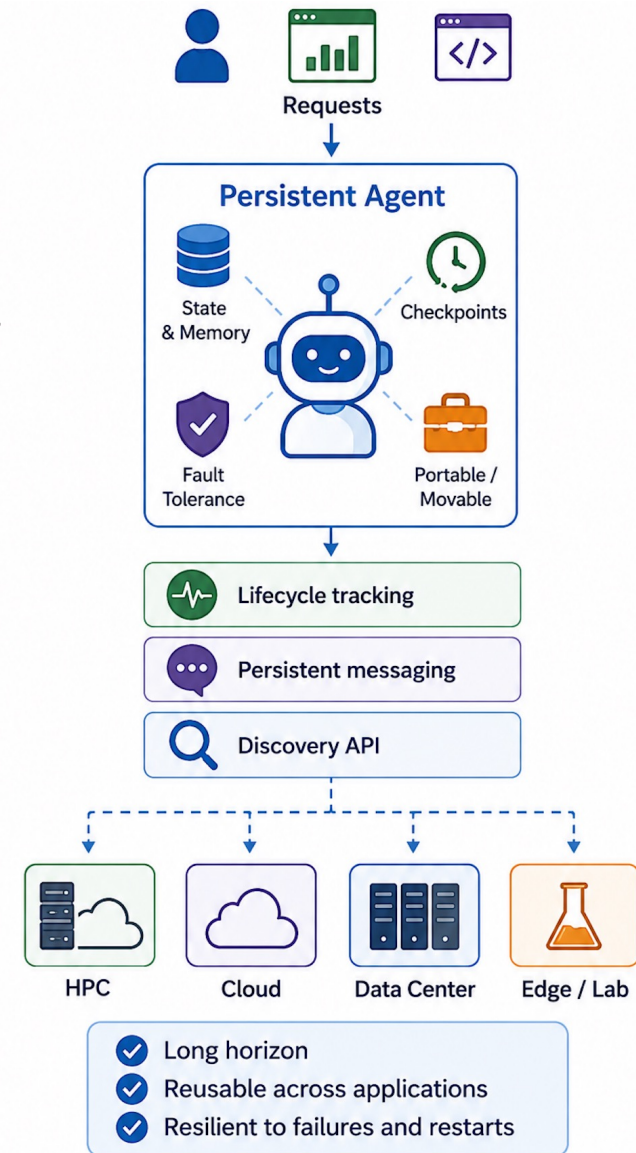
In progress: Creating and managing persistent, long-lived agents

We need stateful agents that run continuously, are fault tolerant, and can be resumed (or even moved between locations)

Persistent agents enable long-horizon, autonomous applications and reuse between applications

Toward long-lived managed agents:

- Control plane lifecycle tracking
- Persistent messaging: messages only removed after actions complete → safe restart
- Agent discovery API: find and reuse active agents across systems



Common needs across diverse applications

THINK

Interact with Models

- 1 Use diverse LLMs and foundation models; manage prompts, context, and structured outputs.

REMEMBER

Manage Memory & State

- 2 Maintain context, persistent memory, scientific state, artifacts, and knowledge across executions.

PLAN

Plan & Orchestrate

- 3 Decompose goals; coordinate tasks, agents, and workflows; manage dependencies, retries, and long-running work.

ACT

Act on the World

- 4 Invoke scientific tools, data services, compute resources, APIs, and physical instruments.

OPERATE

Deploy & Operate Agents

- 5 Run agents from edge to exascale; federate, scale, monitor, steer, pause, and recover.

TRUST

Govern, Verify & Track

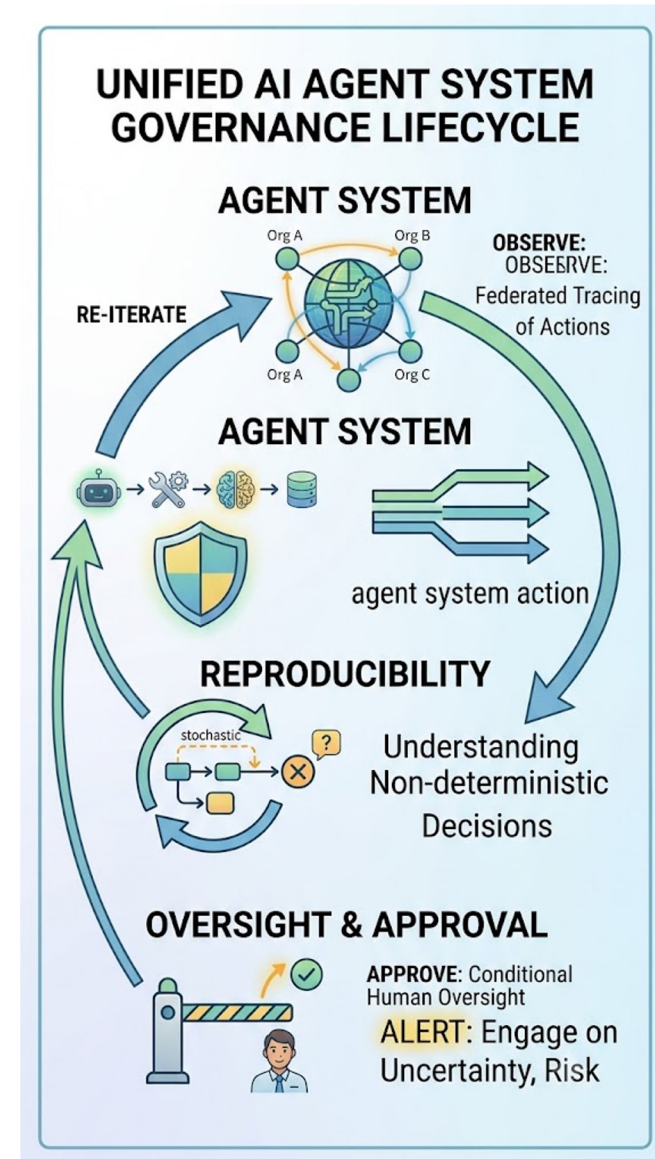
- 6 Manage identity, authorization, policy, evaluation, provenance, lineage, and auditability.

THINK → **REMEMBER** → **PLAN** → **ACT** → **OPERATE** → **TRUST**

Provenance, observability, reproducibility, human oversight

- Capturing full causal chain (which agent, which tool, which model, which data) for reproducibility
- Observability across a federated agents requires distributed tracing across boundaries
- Debugging “why did the agent do X” when the reasoning trace may be long, non-reproducible, stochastic
- Efficiently engaging humans when uncertain, unsafe, or to perform actions not explicitly approved previously

PROV-AGENT, FlowCept: Amal Gueroudji Renan Souza, Rafael Ferreira da Silva, et al.





Every agent leaves a **marker** for the next.

Cairn turns real ratings from agents into a live trust signal for any tool, data source, or peer agent, so your agent can choose what works and skip what doesn't.

Install the skill

Explore the scores →

TRUST DIFFERENT ENTITIES

brave-search-mcp	CAPABILITY		0.91
https://api.weather.gov	DATA SOURCE		0.88
github-mcp	CAPABILITY		0.84

<https://cairnscore.ai/>

Identity, resource rights, and resource management

- Current allocation models are designed for humans, not for thousands of agents operating on behalf of one (or many) humans
- Cannot approve every access decision, but cannot give unbounded authority
- We need new identity, authorization, resource allocation models to support these changing use cases

Scenario: Autonomous Materials Discovery

Dr. Chen's AI agent searches for battery electrolytes:

- Run 50,000 DFT simulations at ALCF
(consuming 200K node-hours)
- Store 15 TB of results at NERSC
- Query materials databases
(ICSD, Materials Project, AFLOW)
- Iterate autonomously for 3 weeks
refining candidates based on results



Discovery Agent
~1,000 decisions/day



500K hrs allocated



100 TB quota



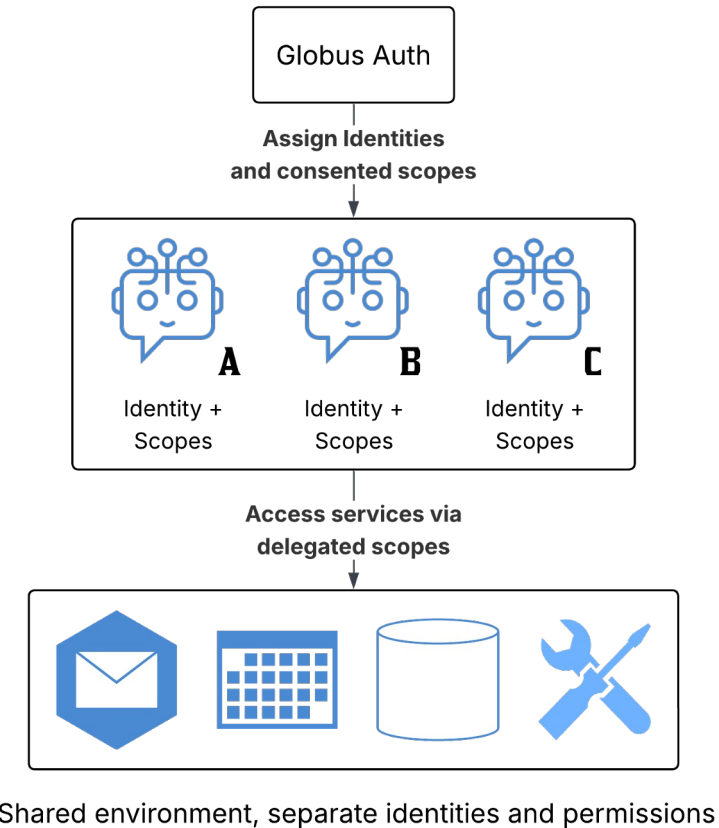
+ genomics data

An agent-first identity and authorization model

Challenges deploying persistent/federated agents:

- What is an agent's identity?
- How do we provide permission for an agent to perform actions?
- How do we adhere to local policies (sovereignty, governance)?
- How do we engage humans in the loop?

Approach: Each agent is assigned its own (Globus Auth) identity, enabling group-based consents to be delegated to sets of agents, and allowing each agent to operate within a defined set of consented scopes



Resource rights: Bounded, delegable authority for agents

- A **resource right** is a transferable token representing bounded authority over a scientific resource (e.g., compute allocations, storage capacity, instrument time, repository access, workflow execution, AI/inference services)
- Authority becomes an explicit, measurable, delegable, and revocable object, independent of the identity currently exercising it
- Lifecycle
 - Allocation authorities issue rights against specific allocations
 - Scientists delegate bounded subsets to agents and team members; agents may sub-delegate
 - Each successive delegation may only narrow constraints, never expand them
 - Resource providers validate presented rights against local policy at request time
 - Any link in a delegation chain can be revoked, immediately invalidating all downstream rights
- Bounds runaway agents : an agent can never consume more than it was delegated

Note: Requires resource providers to accept delegated resource rights

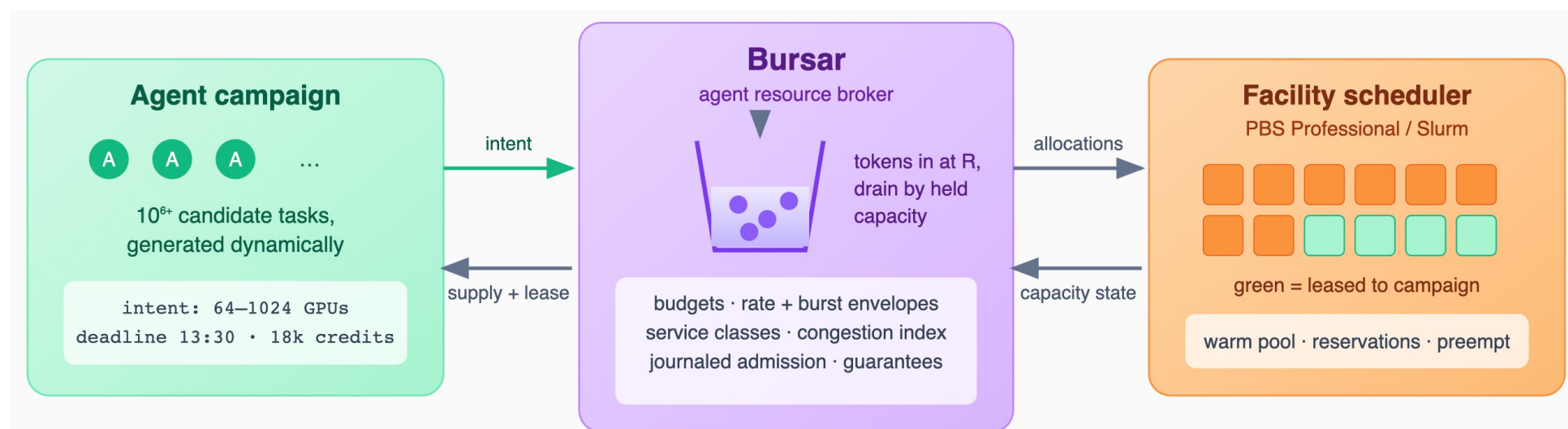
<https://agents4science.github.io/ResourceRights/>

Towards agent-native HPC resource management

- HPC assumes human-scale demand; agents **change the Scale**
- **Allocations aren't enough:** GPU-hour budgets control *total* use, not instantaneous demand, latency, or contention

Bursar: A thin Agent Resource Broker above Slurm/PBS

- Projects receive **resource envelopes** defining guaranteed/sustained rate, token-bucket burst capacity, credit budget, latency/service class
- Agents acquire **short-lived leases**; can run many fine-grained tasks within them
- Task creation cheap and unlimited; resource use metered, bounded, auditable
- **Key principle:** Govern **resource consumption rate**, not task count



Example: Five campaigns sharing 512 GPUs

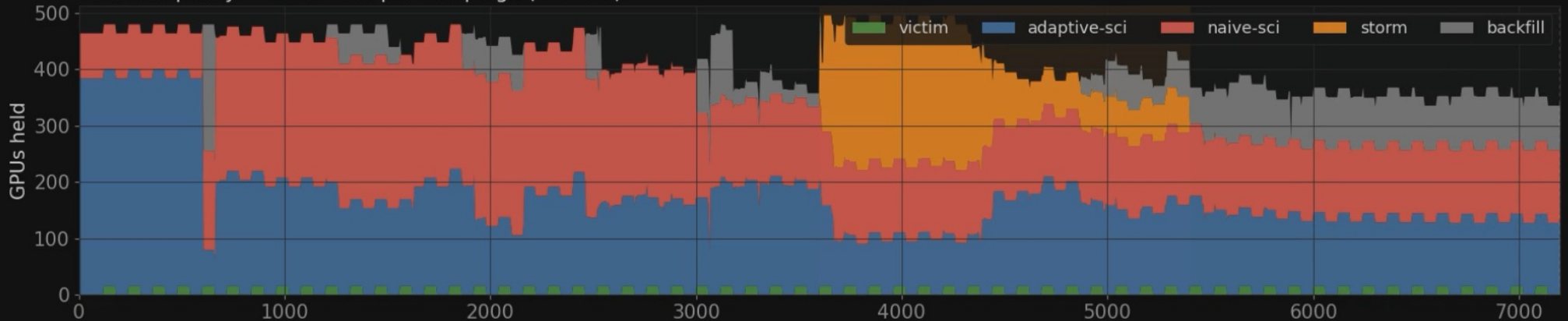
- **adaptive-sci** reads the live price of capacity and runs its most valuable work first, deferring the rest
- **naive-sci** has the *same task list* but always demands 256 GPUs and ignores prices
- **storm** is an adversary that hammers its 384-GPU ceiling for 30 minutes
- **Victim** is a small interactive user asking for 16 GPUs every 2 mins (fairness probe)
- **backfill** soaks up idle capacity and is evicted on 60 s notice.

Bursar — machine-speed back pressure on a 512-GPU pool

t = 2:00:00

five campaigns, two hours simulated — an adversarial burst storm (orange window) is throttled while a guaranteed campaign keeps 1-second lease grants
price 1.50x · storm 0 GPUs · adaptive 128 · naive 129 · victim 16

Pool occupancy — GPUs held per campaign (stacked)



- Adaptive 2.2x more science, 2.5x lower cost than naïve; utilization 82%

Summary: What we have learned from real deployments

- Scientific agentic systems must be **federated**, spanning instruments, HPC, and institutions; respecting data & compute sovereignty; exploiting specialization
- Productivity gains depend on **reducing deployment friction** and **cross-site barriers**
- LLM tool-calling alone is insufficient for complex scientific workflows: domain-aware construction is still required for correctness
- Existing agent frameworks often fail in science due to infrastructure mismatch and integration limits
- The biggest gains come from **dynamic, distributed orchestration of scientific resources and workflows**
 - Kamatar, Pauloski, et al., *Empowering Scientific Workflows with Federated Agents*, IPDPS 2026

Learn more and engage with CAF



CAF Agent of the Week



The **CAF Agent of the Week (AotW)** is an exemplary agentic application or individual agent that demonstrates superior architecture, use of common tools, and scientific impact. This much-sought-after recognition brings, on occasion, a free drink and a CAF hat for the winning team--but the true joy in winning is the glory that comes from recognition by your peers. Needless to say, many aspire to win, but few are chosen.

Discover the Agents of the Week

1. [OPAL — Plant Phenotyping Analysis](#)
2. [IGNOS — Fusion Foundation Model](#)
3. [Foam-Agent — Automating OpenFOAM CFD](#)
4. [LABMATE-Supervisor — Accelerating catalysis research](#)
5. [A2MC-Adaptive Multi-target Calibration for Earth System Models](#)
6. [AISAC-AI Scientific Assistant Core](#)

https://github.com/AI-ModCon/BaseCAF_agent_of_the_week

AI Agents for Scientific Discovery

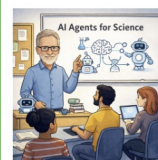
Agentic discovery combines reasoning-capable AI with scientific and engineering resources—such as literature collections, simulation codes, experimental platforms, and knowledge bases—to accelerate the pace of discovery. Recent advances in large language models (LLMs) and related technologies make it possible to build agentic systems that automate key aspects of scientific work: synthesizing information from the literature, generating and prioritizing hypotheses, designing and executing protocols, running simulations or experiments, and interpreting results.

An agent is a persistent, stateful process that acts on behalf of a user or system. An agent may:

- Observe inputs or events
- Plan (decide on) actions using a policy (rules or LLM)
- Act: Execute tools or interact with other agents
- Learn: Update state to adapt over time



We can think of an agent as a *scientific assistant that can reason, act, and coordinate on our behalf*.



Teaching
AI Agents for Science curriculum (University of Chicago, Autumn 2025).



Agentic Capabilities

Capabilities
Deployment patterns from local prototypes to massively parallel HPC inference.



Applications
Example applications of agentic systems for scientific discovery.

<https://agents4science.github.io/>

Roadmap: Toward scalable, trustworthy agentic infrastructure

- **Persistence & scale:** Hardened Academy releases on AmSC infrastructure; thousands of long-lived agents per facility; multi-tenant inference (Exaserve) across DOE AI systems
- **Tools & skills:** Code-execution MCP over IRI + Globus Compute; Genesis Skills repository; tool discovery via DiscoveryEngine, AgentCards, and Cairn
- **Identity & rights:** Agent-first Globus Auth identities in production; resource rights delegation service pilot at ALCF
- **Observability & trust:** Cross-framework provenance (Flowcept / PROV-AGENT); Cairn reputation and monitoring at scale; efficient human oversight
- **Engage:** academy-agents.org | agents4science.github.io | cairnscore.ai

Directions

Goal: Make every project strengthen the next

**A national scientific AI ecosystem
should compound what it learns**

Shared contracts	Models, xCards, tool interfaces, evidence and production handoff.
Complete campaigns	Fund data, computing, experiments and integration together.
Proof of transfer	Reward validated science, reuse and sustained operation.

Outcomes we are driving towards

- ▶ **An AI that proposes a theory**

not just a fit to data, but a falsifiable mechanism that predicts new phenomena and survives experimental test.

- ▶ **An AI that designs and runs its own experiments**

closing the loop from hypothesis to instrument control to result interpretation to next experiment.

- ▶ **An AI that reliably says 'I don't know'**

with calibrated uncertainty that scientists trust enough to act on — including knowing when not to act.

- ▶ **Scientific AI as public infrastructure**

open models, shared compute, common standards — so every lab in every country can participate.

- ▶ **Incentive reform**

where replication, negative results, data curation, and maintenance are valued as highly as novel discoveries.

**The deepest challenge is that
AI for science is not a model problem.
It is a scientific-system problem.**

Data · Theory · Experiments · Compute ·
Incentives · Governance · Human Judgment

All of these have to improve together.

Ian Foster · foster@anl.gov · Argonne National Laboratory