NSF LEADERSHIP-CLASS COMPUTING FACILITY

The TPC Academic Scientific Inference Group: Perspectives from Around the World

SC Asia/HPC Asia

Osaka, January, 2026

Dan Stanzione

TACC TEXAS ADVANCED COMPUTING CENTER

TEXAS The University of Texas at Austin

The Inference Group

- An informal group in TPC to focus on the role of inference in science, and particularly inside academic and quasi-academic infrastructures.
- Inference is a different beast, however, that may require new modalities to support – and we all lack experience running this at scale (We have run batch for decades, but “AI Factories” building tokens is not something we have done as much of).
- So:
  - What are the use cases we see for (Scientific) inference services in research , teaching, and operations for (mostly academic-ish) HPC Centers?
  - How do we provision and support these services and the software stacks around them?
  - Are we all doing the same thing? Can we share practices or even services?
  - What sort of utilization/uptake are we seeing?
  - What data and privacy issues have we encountered?

A Quick History of This Discussion

- First formal session at TPC25, last July in San Jose, CA
- A second session as a BOF at SC25 in November in St. Louis
- During this time, we have had speakers from Asia, Europe, and the US.
- Universities, a few labs and national infrastructures.
  - Asia: Taiwan CHPC, Japan AIST, Singapore
  - Europe: Finland (CSC), Portugal (MACC)
  - USA: TACC, SDSC, NCAR, Argonne
  - A number of other sites in the rooms to listen.

A Quick History of This Discussion

- The focus has been on identifying the problems and common challenges.
- No established technical directions for new development, yet, but some themes are becoming clear based on the two sessions, and a some slack and email discussions.

What have we learned about operations so far, and what are the challenges?

- Every site is running inference services, at some level or another.
- All the sites believe a lot more is coming – but for most of us, it remains a relatively small ”special” workload so far.
  - Often, a set of agents is run temporarily as a big batch job(s), and is stood up and torn down for each experiment.
  - Except for chatbots, mostly for support, code generation, etc. – not a huge load given our user base.
- Most sites are exploring dedicated hardware for inference workloads.
  - GPUs are universal, but there is a lot of experimentation with SambaNova, other platforms.
- Most work is labeled “Pilot” at this point.

Observations and Challenges(2) .

- The current model for standing up an inference service (especially an LLM), both vendor platform and open source, isn't well-suited for multi-tenancy, security, protected data, etc.
- As a result, many sites (at least 4!) are building frameworks to support running many model endpoints, often based on something like LightLLM, with some added features.
  - Impacts on auth, identity, scheduling, etc. Need load-balancers for endpoints, multiple user spaces, logs for audit/analysis, etc.

Observations and Challenges(3).

- We don’t have a great feel for how much of inference workloads look like a permanent service (a chatbot, a cloud-like API endpoint), and how much is tied to specific experiments/campaigns/runs.
  - (Some examples in a couple of slides)
  - This is critical for sizing resources and for scheduling – do models need to be resident, or can they be swapped?
- Our best guess:
  - We need both!
  - Largest runs may be ephemeral – but things like services to query large datasets may need to be run in a cloud-model.

Observations and Challenges(4).

- Not knowing the mix, we have lots of scheduling and resource management challenges.
  - The previously mentioned problem of resident models or loading from “out of core”, and provisioning compute for that.
  - Most end users coming from cloud services who might be willing to run “temporary” services want them to schedule as containers with Kubernetes.
  - To share endpoints, we also need lots of RAG storage – and this is likely not Lustre (graph stores, etc.).
  - Failure/restart modes are different.
- We may have enough hardware for all this, but we can’t share and utilize it with anywhere near the efficiency we can run batch HPC/training workloads.
  - Hence the scheduling issues.

Observations and Challenges(6).

- Not sure that a lot of what “scaling” means in the commercial context (for inference services) applies the same way to our use cases.
- It is unlikely we will have a coding assistant, chatbot, or other interactive services that need to scale to thousands of simultaneous users and super high sustained token rates.
  - A reasonable trained LLM for coding or chat can fit on a GB10 for $4k, as long as you are the only one using it.
  - Much as we might like it to, an astrophysics-specific coding assistant will not have 100,000 simultaneous users.
- Much of the commercial “GPU sharing/efficiency” space doesn’t make our assumptions about reproducible performance – and we don’t know if our users really care.

Observations and Challenges(7).

- Our Compliance models won't work anymore.
  - We've historically done a good job of controlling the things we can control, and making lots of bad behaviour the user's problem (at least for liability/criminal purposes).
    - E.g., if they copy a restricted code to an open space and put a bunch of students on it, they violate the end-user agreement, and the problem is often contained in that group…
    - And auditors consider this "settled law".
  - If a user gives restricted data to your chatbot service, and the LLM behind it spits that out to a different user… who is responsible?
    - This is not settled at all – meaning we have risk.
- Note – this is about compliance, it has nothing to do if the AI is correct or Trustworthy – that's a separate problem not unique to us!

Integration of NSF NCAR's Compute and Data (stolen from Thomas Hauser, NCAR)

Research relevant outputs

Boundary conditions

Trained Model

GDEX Geoscience Data Exchange

- **Community reference datasets optimized for ingestion by AI model applications**

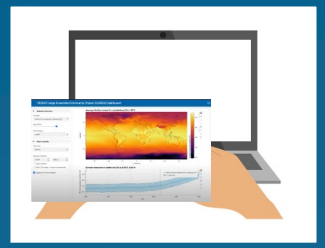
Data Ingest Pipelines

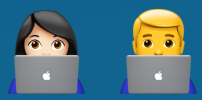
CASPER

jupyterhub

CIRRUS
NSF NCAR Cloud Infrastructure for Remote Research, Universities and Scientists

Model Inference





AI Model Training Configuration

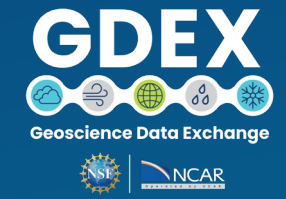
NSF NCAR Operated by UCAR

NSF NCAR AI ROADMAP

Inference Pilot – Making AI/ML Accessible in Earth System Science

- Inference platform
  - Multiple models
  - Choice of initialization datasets
  - Democratize access
- Evaluate accessibility and scientific utility
  - Ease of use
  - Model intercomparison
  - Inform our infrastructure and future services

- NCAR's Geoscience Data Exchange (GDEX)
  - Data Commons
  - Data products used in AI/ML workflows
  - Data fully accessible on our compute platforms
  - Integrated with Open Science Data Federation (OSDF)



LUMI AI Factory

Utilizing Inference for Science Services Within the Academic Community

Aleksi Kallio, Director, LUMI AI Factory Service Center

(Courtesy of Aleksi at CSC)

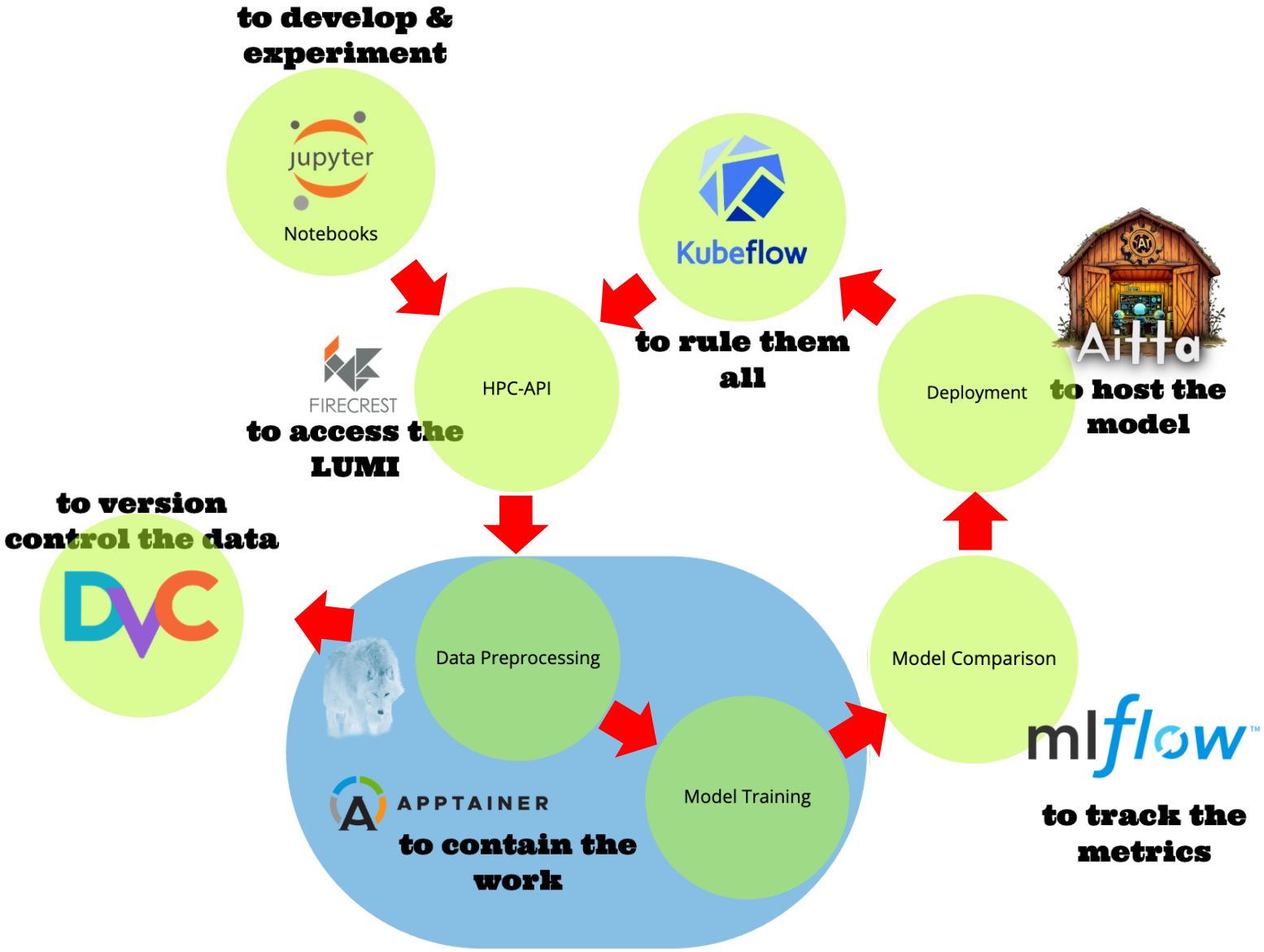
LUMI AI Factory

Inference services in LUMI AI Factory

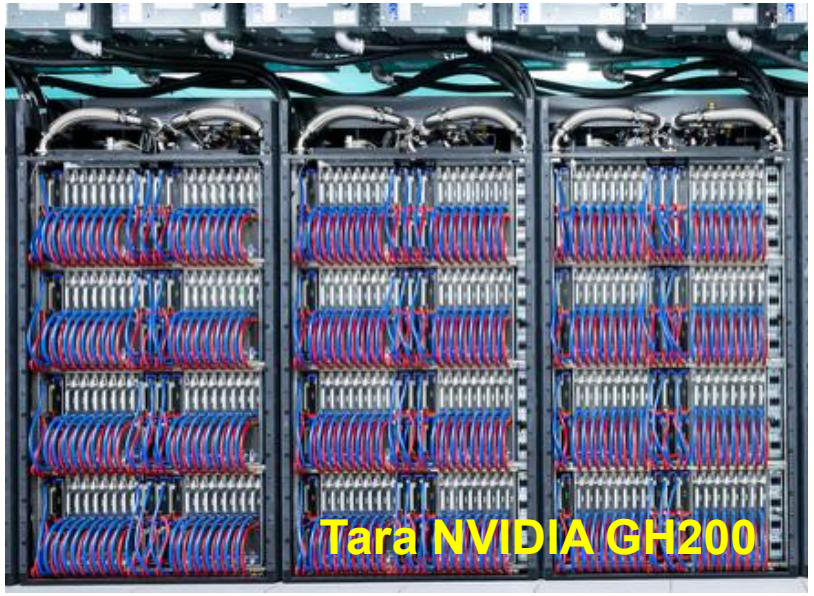
- Aitta: general purpose scalable AI inference service for R&D
  - Access AI models at scale
  - Unified access to models through scalable API
  - Experiment, prototype, and deploy
  - Inference as part of workflows (batch)
  - Supports Slurm as backend
  - Developed in-house
- Other frameworks will be provided as well, including EXA4MIND (by IT4I and METU)

(Courtesy of Aleksi at CSC)

LUMI AI Factory



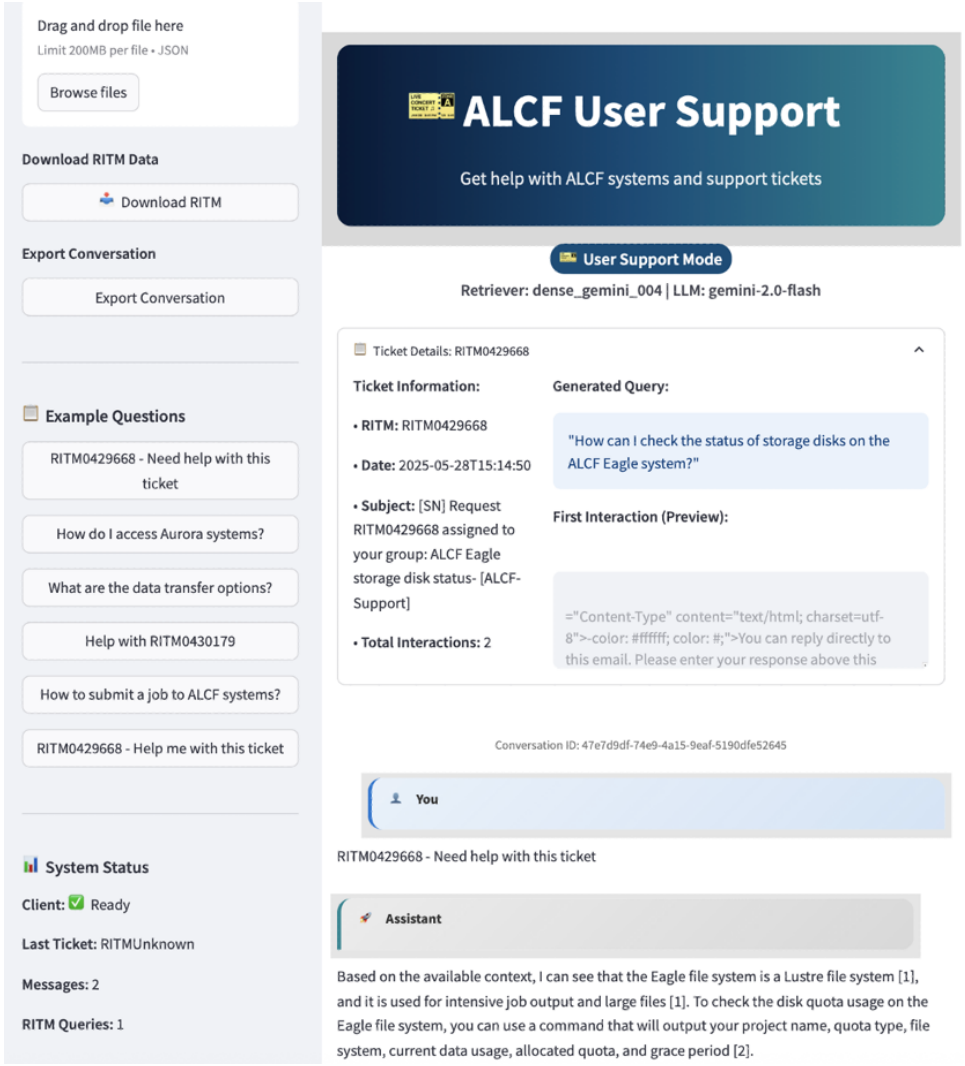
ALCF IS DEPLOYING DIVERSE INFERENCE SYSTEMS FOR SCIENCE



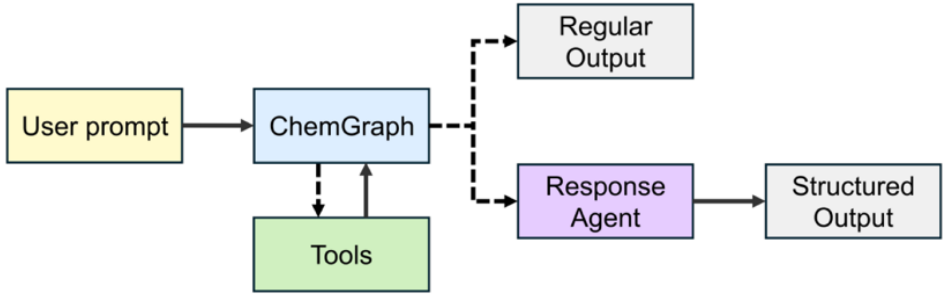
SOME INFERENCE SERVICE USE-CASES

https://docs.alcf.anl.gov/services/inference-endpoints/

Courtesy of Venkat from Argonne



ChemGraph – Thang et al. https://www.arxiv.org/pdf/2506.06363



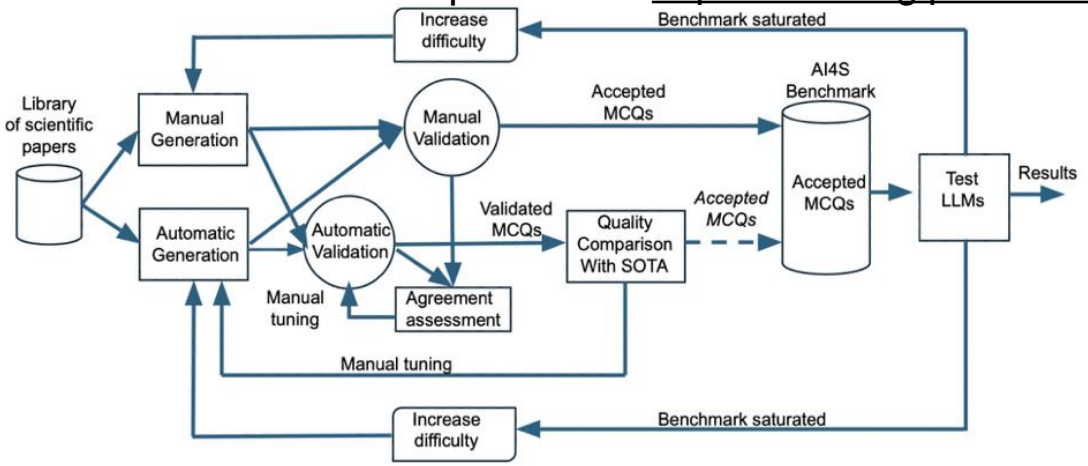
Workflow and Cheminformatics

- LangGraph
- ASE
- RDKit
- PubChemPy

Simulation Backends

Semi-empirical	Ab initio	ML Potentials
- xTB	- NWChem	- MACE
- EMT	- ORCA	- UMA

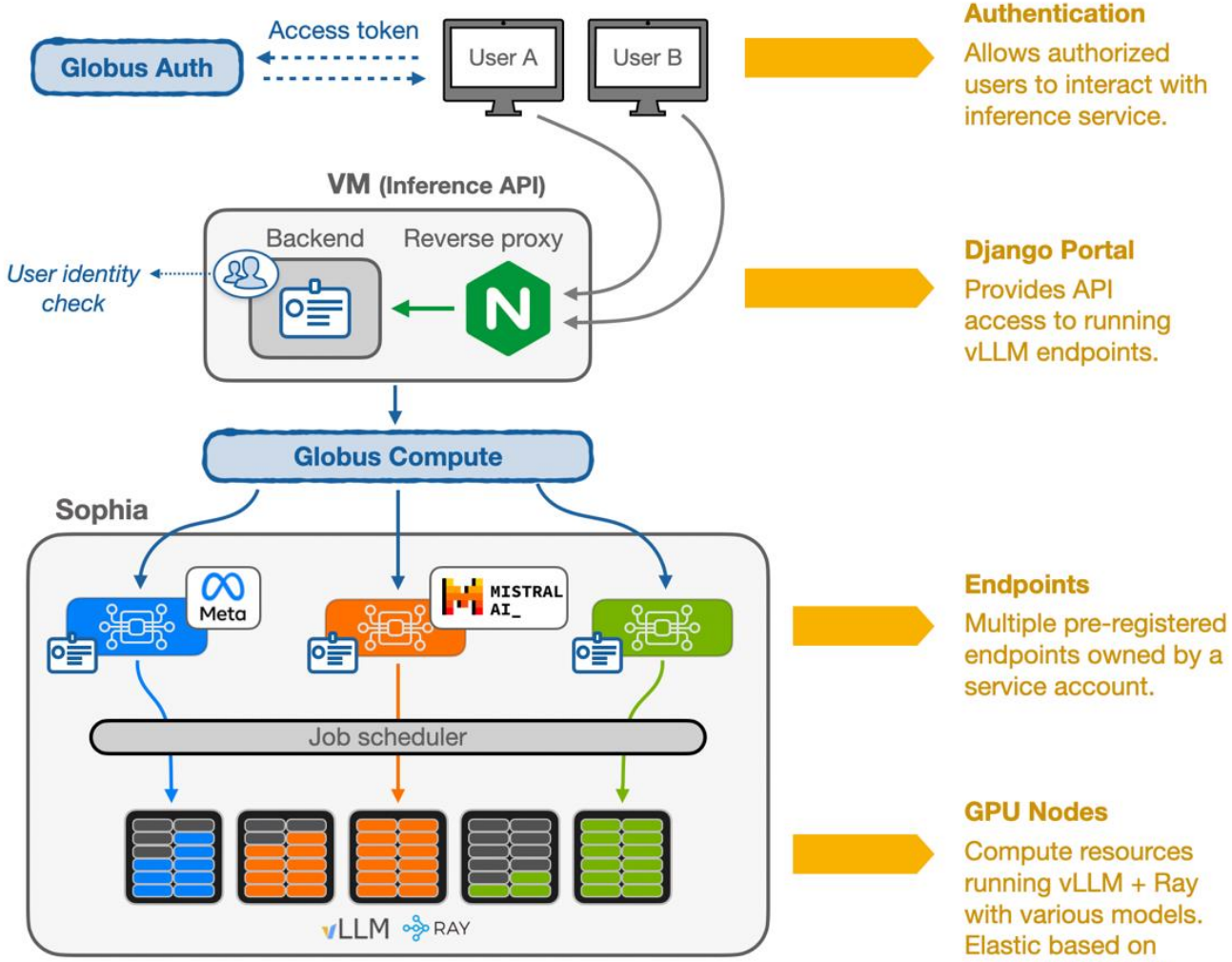
AuroraGPT-EAIRA – Capello et al. https://arxiv.org/pdf/2502.20309



Current Status: ~200 users, ~10 million requests, over 11 billion tokens generated

FIRST: **F**ederated **I**nference **R**esource **S**cheduling **T**oolkit

Courtesy of Venkat from Argonne

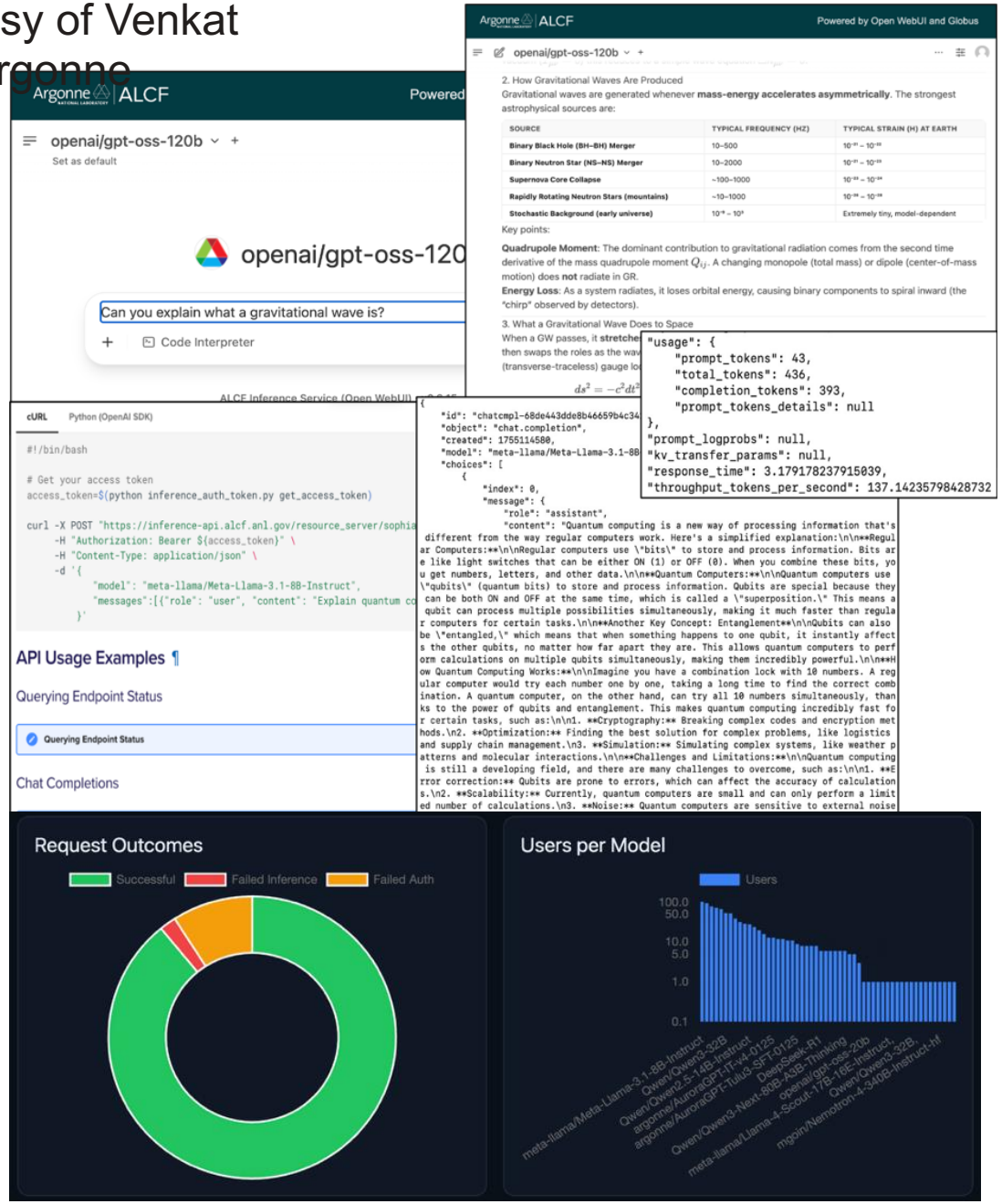


Authentication
Allows authorized users to interact with inference service.

Django Portal
Provides API access to running vLLM endpoints.

Endpoints
Multiple pre-registered endpoints owned by a service account.

GPU Nodes
Compute resources running vLLM + Ray with various models. Elastic based on requested model.



https://auroragpt-anl.github.io/inference-gateway/

Where to go from here?

- Simply the exchange of ideas is, to me, very valuable, even if we do nothing but provide updates a couple of times a year.
  - It affects our operations, I suspect it impacts everyone else participating too – and everyone they talk to at other sites.
- But, if we wanted a technical outcome. . . There are promising avenues to pursue.
  - Shared front-end for inferences services – we are building ~4 already.
  - Resource Managers, Schedulers, and their associated *policies*.

Thanks!

dan@tacc.utexas.edu