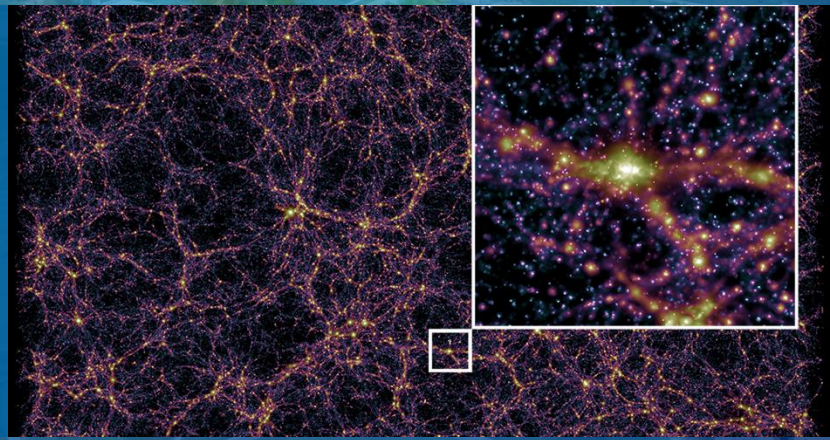
HACC I/O AT SCALE ON AURORA: EARLY EXPERIENCES ON LEVERAGING DAOS



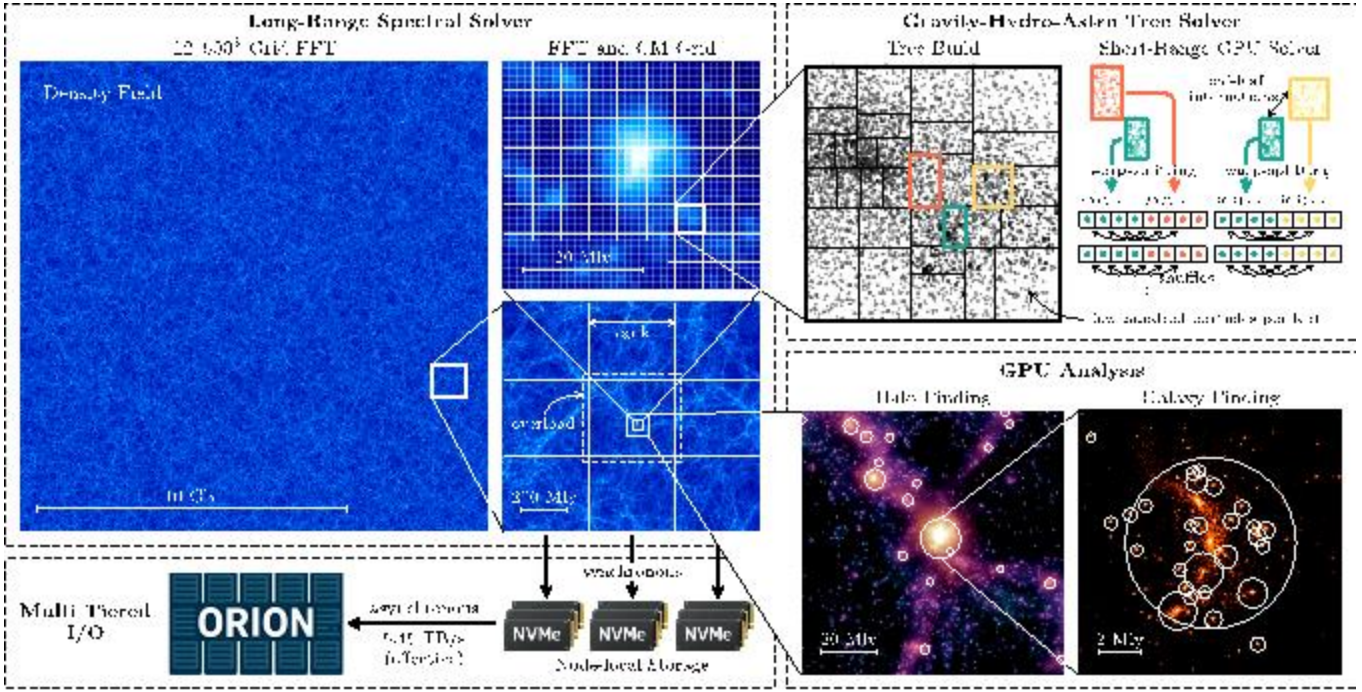
ESTEBAN RANGEL
Assistant Computational Scientist
CPS Division

U.S. DEPARTMENT of ENERGY
Argonne National Laboratory is a U.S. Department of Energy laboratory managed by UChicago Argonne, LLC.

Argonne NATIONAL LABORATORY
Argonne Leadership Computing Facility

FRONTIER-E SIMULATION

I/O at scale for cosmological simulations like HACC



OVERVIEW

HACC on Aurora

Simulation configuration

- NP=23,760^3 (over 13.4 trillion particles!)
- 500 steps
- Full range of *in situ* analysis
- 8100 Aurora ECBs
  - 1 MPI rank per GPU stack
- Total simulation time estimated: ~50 hours wall time

HPC environment

- Aurora ECB
  - 2 Intel Xeon CPU Max Series processors: 64GB HBM on each, 512GB DDR5 each
  - 6 Intel Data Center GPU Max Series, 128GB HBM on each
  - tmpfs `\tmp`
  - 128 node DAOS cluster (daos_user)

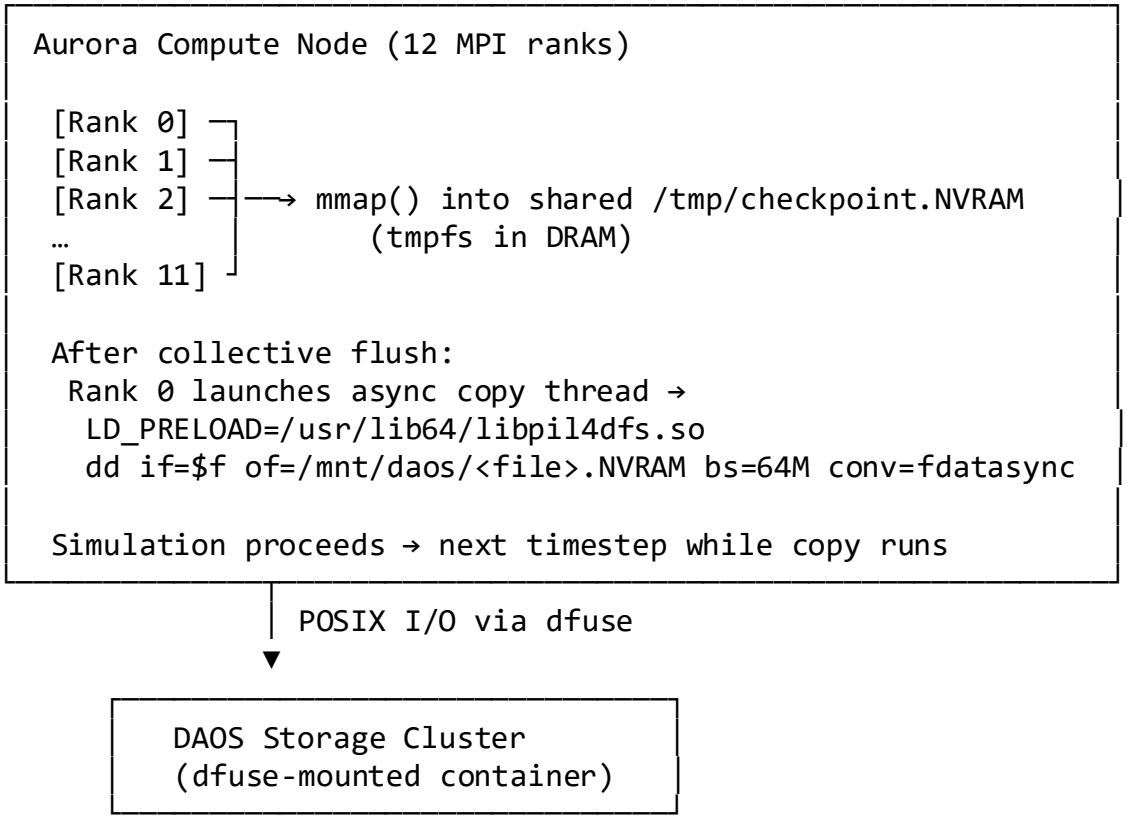
CHALLENGE

I/O at Scale

- 8,100 Aurora ECBs (12 MPI ranks/node)
- 13 trillion particles ~500 TB checkpoint, keeping 3 checkpoint window, ~1-1.6 PB of total checkpoint storage
- 500 simulation steps (~6 minutes per step) affected by imbalance and clustering later in the simulation 50hr+ Wall time
- *in situ* analysis at 100 of the timesteps generates many small to medium files
- Estimated 4PB of total storage is needed for all data products

STRATEGY

HACC Checkpointing



STRATEGY

Checkpoint Workflow

- Each node's 12 MPI ranks **memory-map** (mmap) a shared file in /tmp (tmpfs).
- All ranks write their particle blocks directly into the mapped region — no write() syscalls.
- The file (checkpoint.NVRAM) is **overwritten in place** each checkpoint.
- Local rank 0 spawns an **asynchronous copy thread** to persist data to DAOS:
  - `LD_PRELOAD=/usr/lib64/libpil4dfs.so dd if=$f of=$dir/$(basename "$fnew") bs=64M conv=fdatasync status=none`
- Copy uses **POSIX I/O via dfuse**; compute continues concurrently.
- No double buffering required — compute phase exceeds copy time

DAOS CONFIGURATION ON AURORA

HACC Checkpointing

Please consult the DAOS documentation for the latest information -- results presented here may be outdated. https://docs.alcf.anl.gov/aurora/data-management/daos/daos-overview/

daos container create --type POSIX ${DAOS_POOL} ${DAOS_CONT} --file-oclass=EC_16P2GX --dir-oclass=RP_3G1 --properties=cksum:crc32,srv_cksum:on,rd_fac:2,ec_cell_sz:131072 --chunk-size=2097152

Parameter	Setting / Value	Notes / Rationale
DAOS Access Mode	POSIX via dfuse mount on /mnt/daos	Transparent integration with HACC async copy thread
Library preload	LD_PRELOAD=/usr/lib64/libpil4dfs.so	Enables high-throughput POSIX streaming through dd
Container Type	POSIX	Matches dfuse interface
Pool / Container Classes	--file-oclass=EC_16P2GX / --dir-oclass=RP_3G1	File data uses erasure-coding (16 data + 2 parity); directories replicated 3×
Properties	cksum:crc32, srv_cksum:on, rd_fac:2, ec_cell_sz:131072	End-to-end CRC32 verification; 128 KiB EC cell; redundancy factor 2
Chunk Size	--chunk-size=2097152 (2 MiB)	Tuned to align with 64 MiB copy blocks and reduce metadata
Copy Command	dd if=$f of=$dir/$(basename "$fnew") bs=64M conv=fdatasync status=none	Block-aligned, synchronous flush for durability
File Layout per Node	1 × ~50 GB checkpoint (checkpoint.NVRAM)	12 ranks → 1 file → 8100 files total at 8100 nodes
Checksums	Checksums **on**	Verified integrity

RESULTS

Checkpoint on 8100 Aurora ECBs

Operation	Path / Target	Data Volume	Time (s)	Throughput (TB/s)
Checkpoint Read	DAOS (POSIX Read)	416 TB	—	**7.57**
Write to DDR (tmpfs)	Ranks → /tmp (mmap)	416.6 TB	—	**237.36**
Asynchronous Copy to DAOS	tmpfs → DAOS via dfuse	416 TB	**111**	**3.75**

Key Observations

- **tmpfs bandwidth** (237 TB/s) ≫ **DAOS** bandwidth → checkpoint creation fully hidden by compute.
- **DAOS copy** sustained 3.75 TB/s aggregate at 8 K nodes → scales effectively with node count.
- **Read-back from DAOS** achieved 7.6 TB/s → read path faster than write, confirming balanced layout.
- **Effective I/O overlap:** > 95 % of checkpoint time hidden behind compute.

COLLABORATIVE DEBUGGING

with DAOS Engineers

Area	Issue	Outcome / Next Step
Agents	Leftover daos_agent daemons on some nodes	Cleanup script under review with ALCF Ops
Clush	High overhead and noisy SSH output	Disabled -B, tuned SSH opts, fan-out = 512
Dfuse Mounts	8 K concurrent mounts > 20 min	Split into 1 K blocks → ~5 min total
Disconnects	Pool disconnect took 15–20 min	DAOS ticket (open)
Metrics	Mismatched connect/disconnect counts	Reported to DAOS team for follow-up

SUMMARY & ACKNOWLEDGMENTS

Hybrid I/O Design

- Node-local **mmap + tmpfs** enables DRAM-speed checkpoint writes.
- **Asynchronous DAOS copy** hides backend latency—no compute interruption.

Scalability & Performance

- Scaled to **8 K Aurora nodes**, ≈ **416 TB** per checkpoint.
- Sustained **3.75 TB/s write** and **7.6 TB/s read** through DAOS EC_16P2GX.
- Over 95 % of I/O time overlapped with simulation compute.

U.S. DEPARTMENT of ENERGY
Argonne National Laboratory is a U.S. Department of Energy laboratory managed by UChicago Argonne, LLC.
Argonne NATIONAL LABORATORY | Argonne Leadership Computing Facility

ACKNOWLEDGMENTS

- **Collaborators**
  - **DAOS Engineering Team:** Johann Lombardi, Alex Kulyavtsev, Samir Raval,
  - **ALCF:** Paul Coffman, Kevin Harms.
  - **HACC Team:** Esteban Rangel, Nick Frontiere, Adrian Pope, Michael Buehlmann, JD Emberson


Argonne
NATIONAL LABORATORY
Argonne Leadership
Computing Facility