NSF LEADERSHIP-CLASS COMPUTING FACILITY

The NSF Leadership Computing Facility – what's coming up soon, and challenges for the future

Dan Stanzione

OU Supercomputing Symposium

September, 2025

TACC TEXAS ADVANCED COMPUTING CENTER

TEXAS The University of Texas at Austin

Thanks for the invitation back!

- I believe it's been *22* years since I started coming to this excellent symposium
- You keep inviting me to speak (9th year in a row, and I don't know how many times overall) – so I'm struggling for new things to say ☺.

What is TACC?

- We are the UT and UT System Research Computing Facility
- We are also the largest NSF-funded national computing center for open science
  - As well as NIST, NASA, NIH, DARPA, DOD, etc.
- 200+ Dedicated staff
- Altogether, ~20k servers, >1M CPU cores, 1k GPUs
- About seven billion core hours over several million jobs per year – for 3,000 projects and ~40,000 users per year.

Federal Investments in TACC are over $1B in last 10 years; and over $1B slated for next 10 years. While we are a national provider, we have *by far* the most computing resources of any University in the country (and often the world), and will continue to through the 2020s.

Who uses TACC?

- At UT, about 70% of NSF grant recipients, and >50% of NIH grant recipients, are TACC Users.
  - (~$199M to UT Austin in 2024 *not* including awards directly to TACC).
- Around the country, users doing unclassified research at more than 400 institutions use TACC on over 3,000 projects for year.
  - Including a number of startups; large industry use us more for tech pathfinding.
- Since it's inception in 2001, TACC has had well over 100k users – 90k of which are students.
  - >30k use the resources in any given year.
- TACC users have 4 Nobel prizes, many Gordon Bell prizes, and countless "first of its kind" computational achievements.
- Access to TACC is provided through:
  - NSF ACCESS and NSF Leadership Computing Programs.
  - NAIRR Pilot (National AI Research Resource).
  - UT-Austin and UT-System programs.
  - Direct investment from partner institutions (Texas A&M, Texas Tech, North Texas, etc.).
  - Direct contracts through other agencies (though NSF provides some courtesy time to NIH, DOE, NASA, etc.).

The NSF Leadership Class Computing Facility remains on Track

- Despite an incredibly challenging year for science funding.
  - At a time when everyone realizes the power of computing (read:AI) can transform society, the US is the only government in the world pulling back public sector investment… not to mention all the other science areas.
- The schedule and the budget remain the same.
- Amazingly, the machine configuration is almost exactly the same!

Stampede 3 in Context: TACC Compute hardware

The big systems in 2025

Resource	CPU type	#Nodes/Sockets/Cores	GPU Type	# GPUs
Frontera	Xeon (Cascade Lake)	8400/16800/470,400	RTX (Volta)	360
Lonestar-6	AMD Epyc	600/1200/76,800	NV A100	255
Stampede-3	Xeon (Sapphire Rapids)	2,024/4,048/150,080	Intel PVC	80
Vista	**ARM/Grace**	**840/1080/77,760**	**NV H100**	**600**
Horizon	*ARM/Grace-next*	*6750/11400/1.02M*	*Blackwell*	*4,000*

- Rough total peak power, 9.5MW
- Rough total average power, ~6MW
- Plus cooling power
- Horizon will add 13MW

AI Inference endpoint hosting available very soon

Smaller NSF Platforms:
Jetstream – Cloud
Chameleon – CS Testbed

Other Compute Platforms:
Cyclone - Kubernetes

Storage Platforms:
Ranch – Archive
Corral – Published Collections
Stockyard. -- Sitewide Work

Worth Mentioning

Hype aside…

We have 12,000 plus CPU nodes

We have less than 1,000 GPU nodes

CPU nodes still have longer wait times ☺.

Vision – The NSF Leadership Class Computing Facility

- A more capable follow-on to the current (aging) NSF leadership systems
- A more holistic, long term, and collaborative view of how we support “leadership applications”.
- An NCAR-like leader and anchor for the NSF computational science and engineering community, existing in the context of other NSF and University investments in research computing.
- A broader view of HPC, with associated systems and services:
  - Simulation, Analytics, AI, of course.
  - Instruments/Edge/IoT
  - Interactive, Urgent, Automated, and Batch
  - Data Lifecycle and Reproducibility
- Workforce Development for a diverse technology and science community of researchers
- Robust Public Outreach

Horizon (The First LCCF System)

- Accelerated: 2,000 GPU nodes with a Grace socket and two Blackwell GPUs, 800Gb Infiniband (non-blocking).
- CPU: 4,750 Vera-Vera nodes, with two 88-core Vera sockets, 400Gb Infiniband.
- Primary compute capability will be :
  - ~300 PF double precision (10x Frontera) (15-20x for most apps).
  - ~10 EF ”TF32” precision.
  - ~20 EF bfloat16 precision.
- Shared filesystem with VAST, Infiniband connected, roughly 400 usable PB, 100% Solid State. (8/16TB/s W/R).
- The system will be housed in a co-location facility, built to spec, with 15MW in the initial buildout, 20MW available.

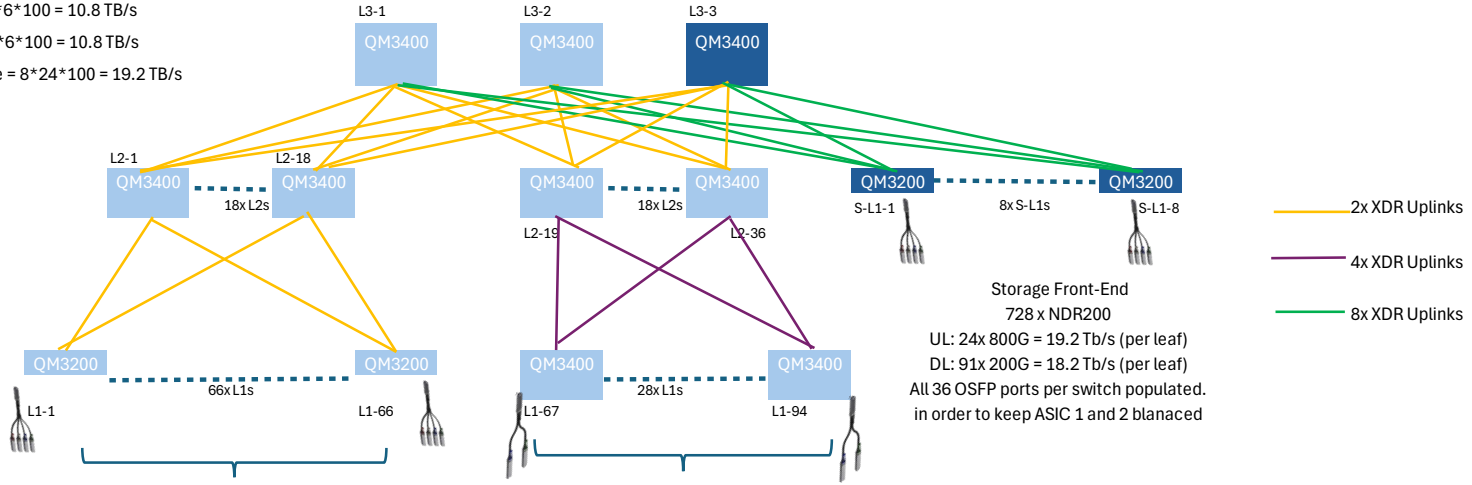
TACC Horizon - Three Level Design

L2 to L3 BW from Islands

VV=18*6*100 = 10.8 TB/s

GB=18*6*100 = 10.8 TB/s

Storage = 8*24*100 = 19.2 TB/s



Storage Front-End
728 x NDR200
UL: 24x 800G = 19.2 Tb/s (per leaf)
DL: 91x 200G = 18.2 Tb/s (per leaf)
All 36 OSFP ports per switch populated.
in order to keep ASIC 1 and 2 blanaced

VV Nodes
4750x XDR400
Non-Blocking
UL: 36x 800G = 28.8 Tb/s (per leaf)
DL: 72 x 400G = 28.8 Tb/s (per leaf)

GB Nodes
2000x XDR (800G) Socket Direct
Non-Blocking
UL: 72x 800G = 57.6 Tb/s (per leaf)
DL: 72x 800G = 57.6 Tb/s (per leaf)

NOTES:

Compute Switches : 66x QM3400s; 66x QM3200s

Storage Switches : 1x QM3400s; 8x QM3200s

VV and GB Single Rack SU with 1 leaf per rack

Storage is 28 racks. Storage L1s will be one every 3.5 racks

GB Nodes will require Socket Direct due to PCIe gen 5

Open Ports (XDR Logical Ports)

- All Compute L1 = 0; Storage L1 : 0
- VV L2 = 2 per L2; GB L2 = 26 per L2
- L3 = 8 per L3

Storage Back End Networking designed and provided by VAST

Storage front-end are **QSFP112** NDR200

And a Little More on the Storage

- For me, 1985 was the first year I got a hard drive.
- I'm hopeful 2025 was the last year I get a hard drive.
- Horizon storage is *all solid state*.
  - From a bandwidth perspective, we will have 25x the total of all three scratch filesystems provided on Frontera.
  - But for the small file access patterns we see most of the time, we think we will see even more to most apps.
  - And… fingers crossed… the interruption time to do file system issues (even the short non-fatal ones) should go way down.

Distributed Centers

- AUCC – Diverse pathways to Data Science and Computing
  - Leverage five HBCUs, that are co-investing in data science as a focus area.
- NCSA - Understand how to exploit new chips for AI for our application mix
  - And in particular how to improve I/O to them, often an accelerator afterthought.
- PSC – Data Intensive Computing
  - Data Mirror for published archives, a focus on protected data and FAIR access.
- SDSC – Testbeds focused on a subset of Science Cases:
  - ML Inference in scientific workflow
  - Exabyte size instrument data workflows
  - Democratization of access (via PATH/OSG).


RANCH
DELL

The AI disruption

- AI *might* replace *some* of our HPC simulation workload, but how much is debatable (and it's almost definitely not all).
- AI *probably* will be a significant disruptor in the overall science workflow.
- AI has *DEFINITELY* disrupted the scientific computing marketplace.

So what Issues Do we Face Moving Forward with Horizon and other AI-driven systems?

- Nodes are fatter (including in *cost*).
- We have lots more network paths – in both kind and quantity.
- We have a lot more networking software “options”.
- Unified Memory is a simpler programming model… but optimization will be harder (especially to get fast messages).

We have bigger, more expensive nodes.

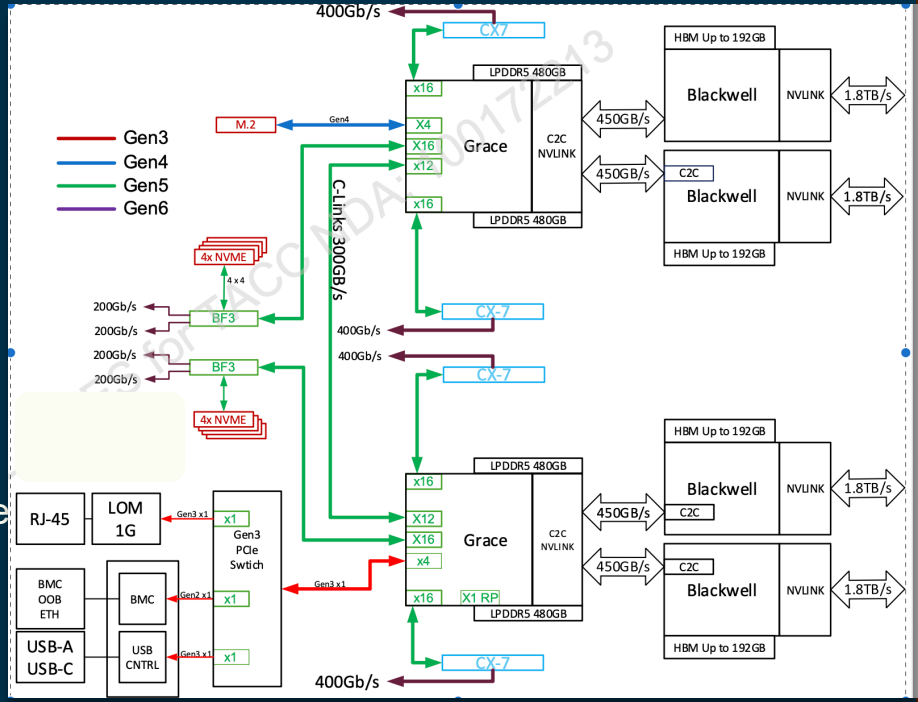
- Rough math on “per node” pricing (all in, including storage and network):
  - Ranger, 2006: $7,600 (~4k Quad Socket Nodes) -- ~2k per CPU socket
  - Stampede, 2012: $5,000 (6400 dual Socket) -- ~2k per CPU socket
  - Stampede2, 2017: $5,000 (6k mixed dual and single socket) -- ~$3.5k/socket
  - Frontera, 2019: $7,300 (~8k dual socket nodes) -- ~$3.5k/socket
  - Typical NVL-72 rack today: $205,000 (18 dual socket, quad, GPU) ~$100k, per CPU socket.
- So, on a per node basis, we’ve gone up ~30x in node price (for a much more capable node) in ~5 years.
  - Implications?

Implications of raising node costs

- In case you haven't noticed, government funding for higher ed and open science in general is not increasing.
- Whether you get an NVL-4 or NVL-72 solution, your basic "hardware" node is two boards, each with a Grace and two Blackwells (and getting denser in Rubin).
  - We're going to treat that as 2 nodes to mitigate this some.
- The nodes have much more capacity… but if you want to optimize performance on a multi-node application, your basic allocation unit is a node.
  - Yes, you can put a bunch of VMs on, or use kernel limits, or containers, or any number of things… but you still are sharing *something* : Network interface, memory bandwidth, GPU-CPU bandwidth, etc.
  - Optimize MPI performance on a node with 8 containers if you only see your container, and you have *no idea* what the other 7 are doing to the network.
  - Do we limit each one to 1/8th the bandwidth through QoS mechanisms we don't really have?
  - What if other containers are using more power, and your slows down?
- A lot of "512 node" machines are about to become 32 node machines… and we aren't ready for that (scheduling, QoS, power management, etc.). Wait times will go up.

OK, let's say you can afford some nodes… let's talk about the network.

- A nameless, "typical" NVL-72 node via NVIDIA today.
- Count those interfaces per node:
  - 11 external interfaces per node!
  - 4 different speeds and latencies...before you think about topology.
  - NVLink, Infiniband, ethernet, plus DPU-based IB.
  - Let's not forget the CPU-CPU and CPU-GPU connectivity.
- Think about setting up libraries for "optimal" routing of messages in this environment.
- Odds users build their Python app containers to pick the correct network?
  - Hint: when there were two choices, they were already near zero.



If you got the network right, be sure things come from the right memory.

- For many things, unified memory is a wonderful thing.
- But for performance, actually knowing where your data is is key to reducing copies, getting transfer rates, etc.
- In the node on the previous slide, we have one address space:
  - But two physically separate LPDDR memories
  - Four physically separate HBM memories.

Consider all the options.

- Oversimplifying somewhat, but “old” GPU nodes:
  - Data is in CPU or GPU memory (one CPU is probably directly connected to PCI NIC, one is not).
  - You can send a message from the CPU memory, or “GPU direct” to copy over PCI directly to NIC.
- Now:
  - Your data can be of any of the six memory locations on the node.
  - You can message over NVLINK or IB (let’s just stick to messages).
  - Which is better for a small message/barrier, sending over NVLINK from CPU memory, or sending over , IB from CPU memory?
    - Or staging from CPU to GPU memory then using NVLINK?
    - Or staging from GPU to CPU memory then using IB?
    - Oh, wait, you have different node lists for each of those networks, so you should optimize differently in-rack or out-of-rack – but in-rack for NVLINK may not mean the same TOR for IB switching.
    - Which IB link? What if it’s on the other GB board?
- There are lots of options – dozens or more.
- But the software will magically figure this out right? And with unified memory it doesn’t matter.

Let's take an example from a "simpler" Grace-Hopper configuration.

Consider these three loops, doing a Python Matrix Multiply:

```
m, n, k = 65536, 32768, 8192
A = np.random.rand(m, n)
B = np.random.rand(n, k)
C = np.random.rand(m, k)
start = time.time()
D = cp.matmul(cp.asarray(A),
cp.asarray(B) )
elapsed = time.time() - start
print(f"MatMul:  Time: {elapsed:.2f}s")
```

Time: 2 seconds

```
m, n, k = 65536, 32768, 8192
A = np.random.rand(m, n)
B = np.random.rand(n, k)
C = np.random.rand(m, k)
start = time.time()
D = cp.matmul(cp.asarray(A),
cp.asarray(B) )
elapsed = time.time() - start
print(f"MatMul:  Time: {elapsed:.2f}s")
```

Time: 1.3 seconds

```
m, n, k = 65536, 32768, 8192
A = cp.random.rand(m, n)
B = cp.random.rand(n, k)
C = cp.random.rand(m, k)
start = time.time()
D = cp.matmul(cp.asarray(A),
cp.asarray(B) )
elapsed = time.time() - start
print(f"MatMul:  Time: {elapsed:.2f}s")
```

Time: 1 second

*The left and middle loops are exactly the same, but I turned on Python Managed Memory Support

Unified Memory is a good thing… but it doesn't necessarily make system software easier!

And Let's not Forget different Communication Layers.

- We've found a lot of ways to do a broadcast or an allReduce.
- We have MPI (and its various flavors).
- Shmem (and its various flavors)
- And now NCCL – the NVIDIA Collective Communication Library
  - Low level optimization across PCI and NVLINK.

- And, of course, all of these things need support for C, C++, Fortran, and Python, and probably more.
  - And let's not even start on the disaster that is Python package management. . .

And hey, those are just the network path challenges. . .

- I/O patterns look very different than they used to.
- It's a Python-driven world… exclusively in AI, but it seems to be spreading fast as the tool of choice.
- Supporting Inference as a persistent service (or not!).
- And then there is the big stuff:
  - Power
  - No 64 bit performance improvements moving forward, from pretty much anyone.
  - The Role of AI in scientific computation
  - The continued existence of open science investment, and Academia.
  - Research computing slowly strangled by the AI behemoth.
  - AI-generated Code, and how to do V&V in the new world of automated code.
- And you know, if we manage that (or even if we don't), we're going to have to talk about chiplets, and quantum…

Let me Quickly Re-iterate my two driving questions going forward.

- Because they still need to happen, though there is more evidence now.
- We desperately need robust research activities in two areas:
  - **AI Hardware for Science:**
    - ”Beowulf for AI” -- Exploit the chips being made for AI to do general scientific computing.
      - We won’t survive if we don’t
  - **AI Full Stack Efficiency:**
    - There are ways to scale up computation other than “throw billions of dollars at it”, and our community, where we didn’t have billions of dollars to throw, is pretty good at it.
    - We need to save the world from bankruptcy, rather financial or carbon.
    - I recently read an analysis that the average residential power bill in Ohio is $15/month higher because of new datacenter demand.

AI Hardware for Science: Why do I think this is Possible?

- Because we have done it before, and programmers in our community are generally pretty clever.
  - Without hacking in the molecular dynamics space in academia, it's possible GPUs would be doing (*gasp*) Graphics Processing.
  - If we let people know that INT8 operations are order 2k faster than 64 bit operations, don't you think they will figure something out?
  - See the Ozaki Scheme among other for emulation methods for FP64.
  - See a bunch of the Gordon Bell papers in the last few years for schemes exploiting mixed precision.

AI Stack Efficiency: Why do I think this can be done?

- It's mostly what DeepSeek did!
  - Nothing earth-shattering algorithmically, just pushing lots of techniques in both traditional optimization, and AI optimization (e.g. MoE) to the extremes.
- The huge focus on "biggest and first" has obviously left efficiency in the dust.
- When $ become scarcer (or export controls cut off top chips), optimization will have time to catch up – at all levels of the stack.

Come explore the new Systems!

Turns out there will be some interesting optimization work to do!

- Vista is in production (Grace, Grace-Hopper nodes).
  - Stampede3 has SXM H100 with x86, and Lonestar6 has H100 PCI if you need to do comparisons!
- We will add an NVL-72 Blackwell queue hopefully before SC25.
- Horizon-GPU Early User should start by March or so.
- We know you will do cool things with them!

NSF LEADERSHIP-CLASS COMPUTING FACILITY

Thanks!

dan@tacc.utexas.edu

TACC TEXAS ADVANCED COMPUTING CENTER

TEXAS The University of Texas at Austin