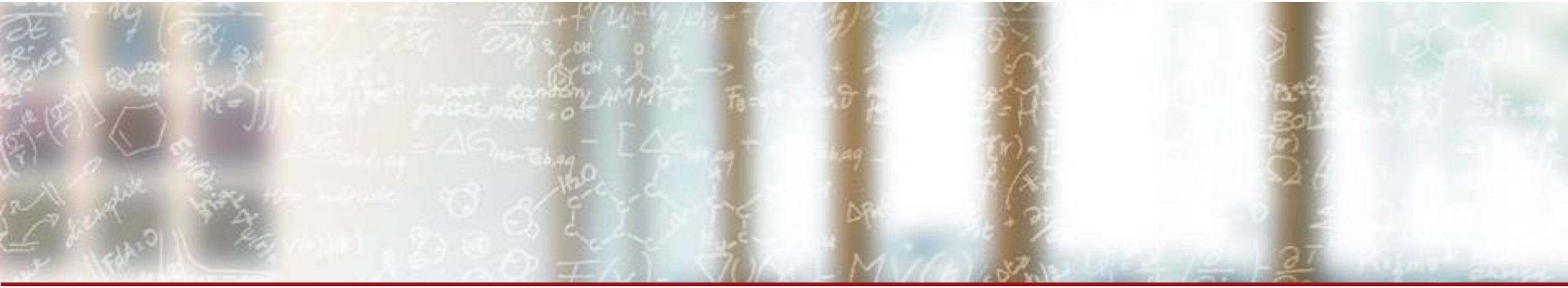
CSCS
Centro Svizzero di Calcolo Scientifico
Swiss National Supercomputing Centre

ETH zürich



A Journey to Provide GH200

Cray User Group 2025

Mark Klein, Thomas Schulthess, Jonathan Coles, Ben Cumming, and Miguel Gila, CSCS

May 7, 2025

A unit of the Swiss Federal Institute of Technology, ETH Zurich

In the Beginning

Alps

- Successor to Piz Daint platform
- Designed from ground up for programmability of resources for workflows – Infrastructure as Code
- Continued support for classic supercomputing use cases
- Additional support for AI, ML and data-driven workflows
- Based on Cray-HPE Shasta Supercomputers
- Mix of different HW

The Journey to the floor (in Brief)

- 2020H2
  - CSCS picks GH200 as architecture for Alps scale-out
- April 2021
  - Announced to world at GTC
- June 6, 2023
  - Introduced our user community to the new Architecture with a “Preparing for the Migration from Daint-GPU to Grace-Hopper” Webinar

NVIDIA’s New CPU to ‘Grace’ World’s Most Powerful AI-Capable Supercomputer

Swiss National Supercomputing Center’s Alps System to enable breakthrough research in a wide range of fields.

April 12, 2021 by Dion Harris

The Journey to the floor (in Brief)

- Q4 2023
  - 21 Cabinets installed and connected to PreAlps staging system in preparation

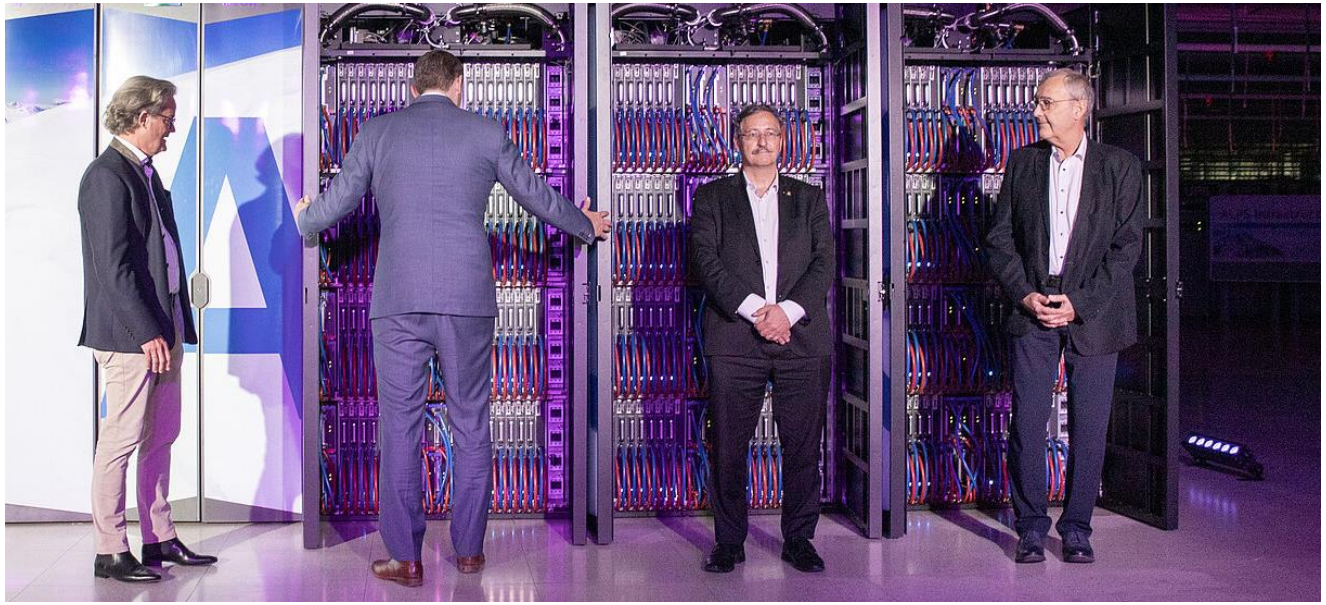
- November 2023
  - Visit to Chippewa Factory to understand power consumption of actual HW
    - Reported results at CUG2024
    - Changed from 21->24 cabinets (populating 7/8s for additional power to blade)
    - Some cold-plate issues discovered under load

The Journey to the floor (in Brief)

- Jan 24, 2024
  - First GH blade shipment arrives at CSCS
  - Immediately installed and start to work on
- Jun 4, 2024
  - Final GH blades arrive at CSCS

The Journey to the floor (in Brief)

- June 2024
  - Recabled GH cabinets into main Alps
- July 2024
  - Acceptance testing passes
- Sept 2024
  - Inauguration of Alps



Early Issues (and their fixes/workarounds)

Large number of components that need to play together nicely

- Unified memory makes hunting memory leaks “fun” (months of investigation)
  - After updating to a new release (linux kernel, slingshot drivers, GPU drivers) noticed large amounts of memory stuck on the GPU
  - First suspected some kernel bug due to OS page migration being held on the GPU
    - Would see a lot of leftover memory after jobs: system processes, /tmp files taking up HBM memory
- Multiple workarounds
  - Numactl is your friend (bind memory to numa 0-3)
  - Add a prolog/epilog to allocate large% of the available memory on the GPU after dropping caches and drain if unable to do so
    - 96% is probably “safe” cutoff; but 90% was determined to be good enough with installed drivers
      - Will revisit threshold as GPU drivers are updated
  - Newer NVIDIA drivers handle this better
    - linux-managed file cache can end up in HBM and it won't be reclaimed by cudaMalloc issue is fixed in 570.114

Large number of components (continued)

- Still seeing memory being held after LLM jobs
  - Played around with module parameters
  - **gdrdrv use_persistent_mapping=0** seemed to prevent situation
- Multiple workarounds (pick one)
  1. The **gdrdrv use_persistent_mapping=0**
  2. Update to at least gdrdrv **2.4.4**
     - (Really no reason to not keep this current, CSCS currently at **2.5** for other bugs)
- Read more at https://github.com/NVIDIA/gdrcopy/issues/313

TCP tuning and Libfabric

- TCP performance isn't great by default
  - Read the slingshot tuning guide for suggestions
    - Why aren't these the default?
- Keep SHS up to date
  - Don't be afraid of building own from github repos
    - But read the commit logs
    - Example: FI_OPT_CUDA_API_PERMITTED (NCCL plugin needs this on libfabric > 1.18)
      - Libfabric 1.22.0 on SHS 11.1.0 doesn't implement FI_OPT_CUDA_API_PERMITTED
      - Libfabric 1.22.0 on SHS 12.0.0 does…
      - Libfabric 2.0 on libfabric upstream
  - HPE packages do not seem well tested on GH200
    - SHS 11.1.0 pycxi includes an arm64 binary lib with "sfence"
      - breaks cxi_healthcheck on GH (fixed again in 12.0.0)

NCCL Performance Variability

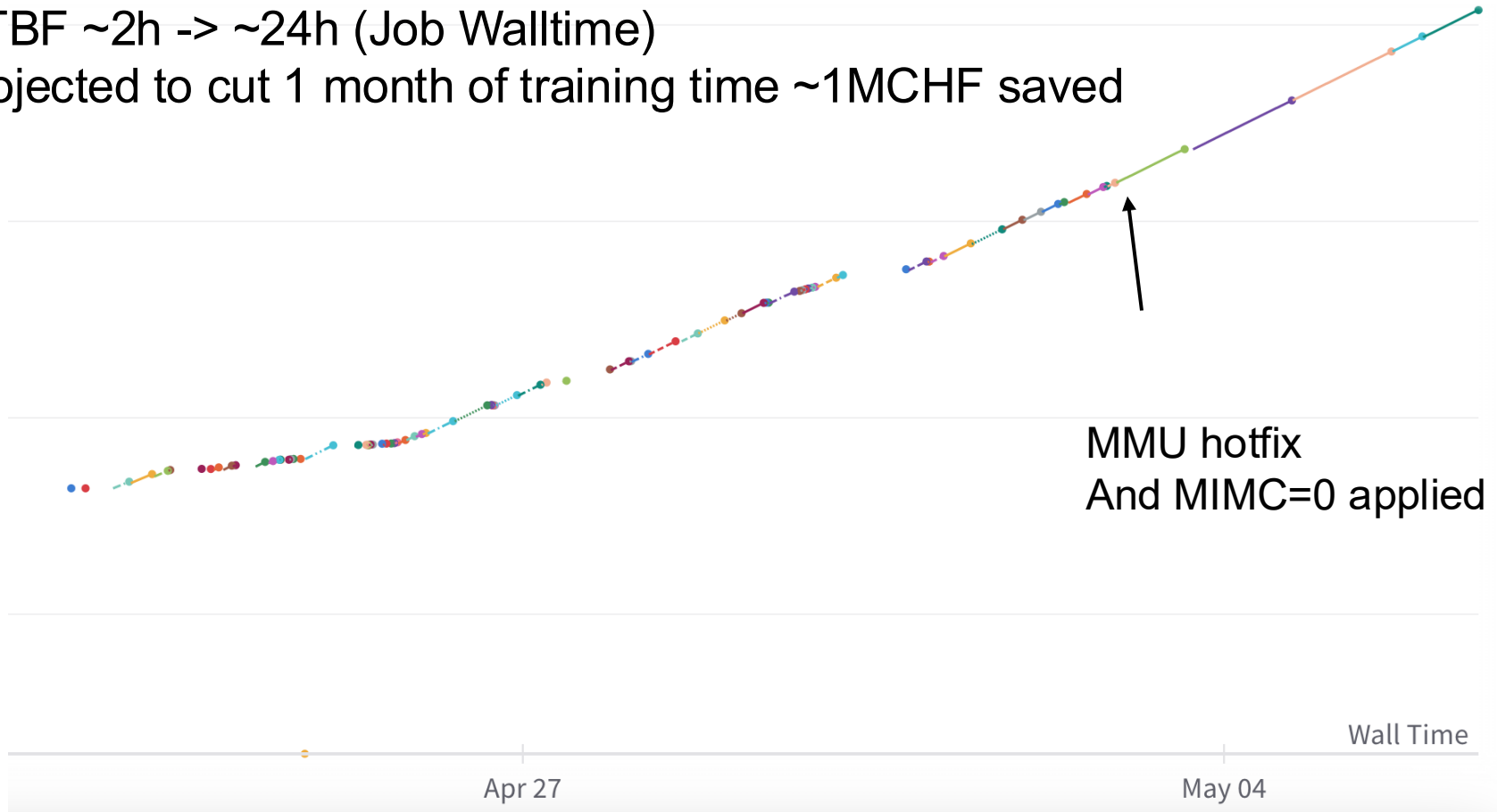
- Large variability was seen in NCCL performance at scale
  - Overall Slingshot Health
  - Slingshot Drivers
  - GPU Drivers
- Sweeping multiple runs across different combinations of cores
  - Any time a rank used cores 44, 111, 183, 256
  - Large amounts of “NVRM: Going over RM unhandled interrupt threshold for irq” messages in console
- Three workarounds possible
  - Isolate cores 44, 111, 183, 256 from workloads
  - Change parameter **nvidia-uvm.ko uvm_perf_access_counter_mimc_migration_enable=0**
  - Deploy a driver that isn’t over a year old
    - Should be fixed in the COS 25.03.x release (**565.57.01**)

Panic! At the Kernel

- Our 512 node LLM training job seeing a MTBF ~2h
  - Still seeing a lot of crashes even after mitigating other problem
  - Signatures varied, but did see a few mentioning: mmu_interval_notifier_remove
    - Found: https://lore.kernel.org/lkml/20220420163516.ab713a22af375788a541f045@linux-foundation.org/T/#mc8a44a4bc12b02b62eb78194ba4a36c42beb4583
    - Patched in kernel April 2022, very simple fix, safely backportable
    - Doesn't currently show up in https://docs.nvidia.com/grace-patch-config-guide.pdf (maybe it's too old)
- Workaround (either)
  - Use a kernel newer than **3 years** old
  - Patch the function in existing kernel (or be creative)
    - CSCS implemented function as kernel module and patched nvidia_uvm to call it as a hotfix

Results (512node LLM training progress)

- Dataset April 30, 2025 22:00 CEST -> May 07, 2025 07:00 CEST
  - Not a single node failure (One unrelated slurm controller failure)
  - MTBF ~2h -> ~24h (Job Walltime)
  - Projected to cut 1 month of training time ~1MCHF saved

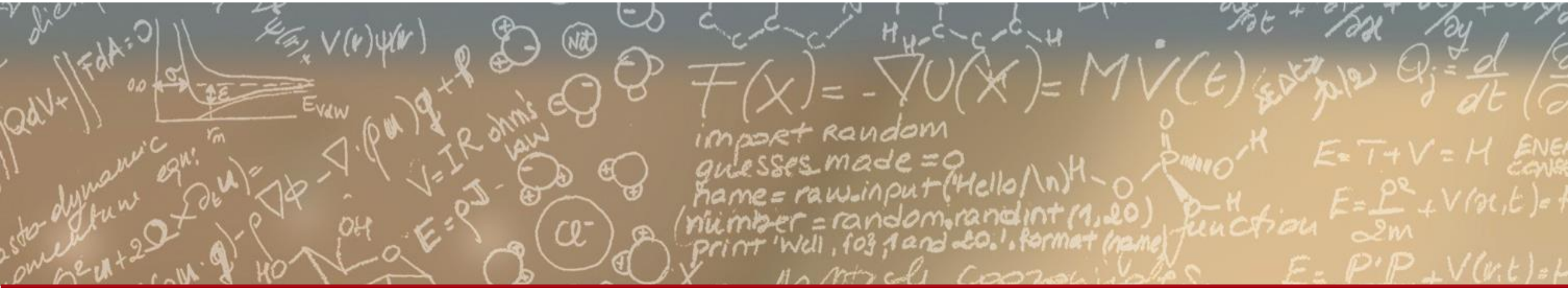


Fighting bugs that have already been solved

- “Ancient” software is painful on new HW architectures
  - COS 24.10.x w/ COS Base 3.1 ARM
    - Linux Kernel 5.14.21
      - Released in 2021
    - NVIDIA Driver 550.54.15
      - Released March 2024
- The MMU problem was fixed in kernel 5.18, The MIMC IRQ slowdown is fixed in newer NVIDIA drivers
- Both thankfully fixed in the COS 25.03.x w/ COS Base 3.3
  - Linux Kernel 6.4
    - Only 2 years old!
  - NVIDIA Driver 565.57.01
    - Only 6 months old!

Still too old!

- Clear that CSCS needs to break away from COS
  - Multiple months of multiple engineering FTEs used to investigate problems that have already been solved in newer releases
  - NVIDIA recommends to us at least kernel 6.8
    - Grace kernel patch guide (https://docs.nvidia.com/grace-patch-config-guide.pdf)
  - Better OS/GPU memory handling in NVIDIA driver 570.114
    - Fixes are rarely backported to previous releases



Thank you for your attention.