Parallel Architectures and Applications

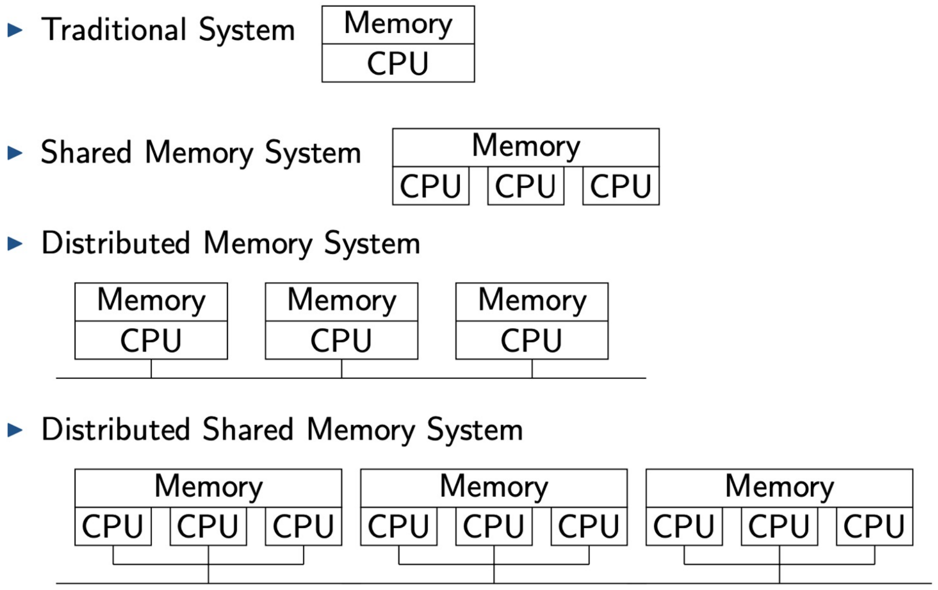
Lecture 1

Sunita Chandrasekaran
Associate Professor, University of Delaware
PDC Summer School, Aug 2023

Materials also prepared by Dr. Felipe Cabarcas, Postdoctoral Fellow, UDEL

Parallel Processing

"For over a decade prophets have voiced the contention that the organization of a single computer has reached its limits and that truly significant advances can be made only by interconnection of a multiplicity of computers." **Gene Amdahl in 1967**.



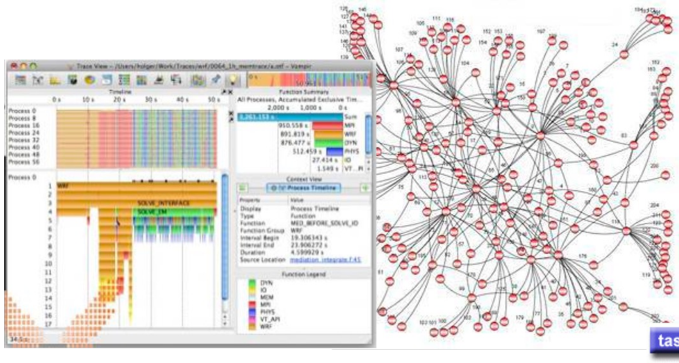
- Traditional System
  Memory
  CPU
- Shared Memory System
  Memory
  CPU CPU CPU
- Distributed Memory System
  Memory Memory Memory
  CPU CPU CPU
- Distributed Shared Memory System
  Memory Memory Memory
  CPU CPU CPU CPU CPU CPU CPU CPU CPU

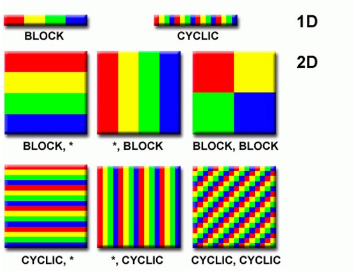
Parallel Processing isn't that easy!

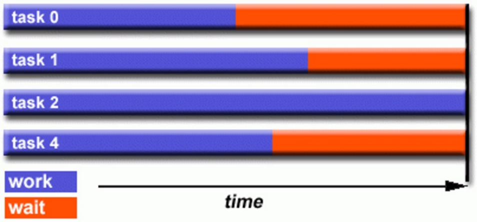
- Challenge 1
  - Finding the limited parallelism available in programs
- Challenge 2
  - Dealing with high cost of communication between threads

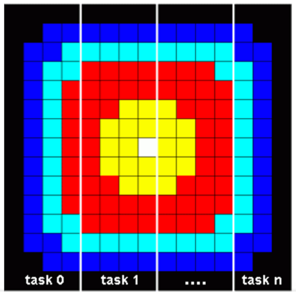
Common concepts for parallel processing

- Load Balancing
- Partitioning
- Data Dependencies
- Cache Coherency
- Cache Consistency
- Synchronization
- Communication
- Parallel Scaling
- Thread, Task, Data, Bit-level parallelism
- Performance analysis and tuning









Some goals of parallel processing

- Keep all the threads busy
- Good load balancing
- Explore different types of granularity
- Avoid too much synchronization between threads (sometimes requires re-writing of the program)

Different frameworks/models

- Directive-based programming models
  - OpenMP and OpenACC
- Lower-level programming frameworks
  - CUDA, OpenCL, HIP
- Kokkos, Raja, alpaka, UPC++, Charm++, Chapel, Intel TBB, HPX, OmpSs, OpenMPI, MPL, SYCL, Coarrays and so on

Focusing on Directive-based models

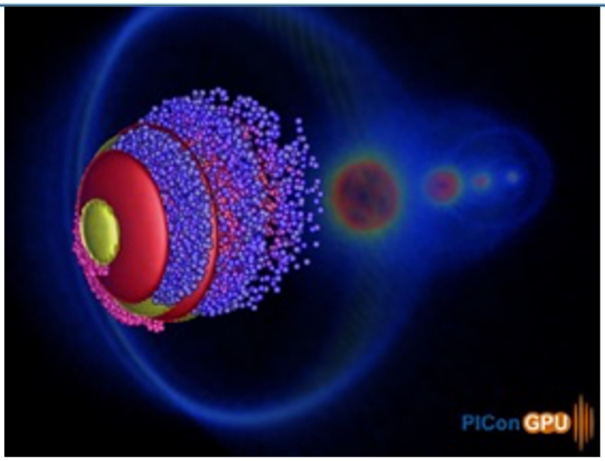
- Single node multicore system (shared memory processing)
  - OpenMP threading or OpenACC multicore

- Multi-node system (Distributed memory processing)
  - MPI only
  - OpenMP/OpenACC within node + MPI across node

- Heterogeneous system (Multi-node + Accelerators)
  - OpenMP/OpenACC for multicore + OpenMP/OpenACC for Accelerators + MPI across nodes

Why should parallelism/acceleration matter?

Accelerating a Plasma Physics (PIConGPU) Code on Frontier

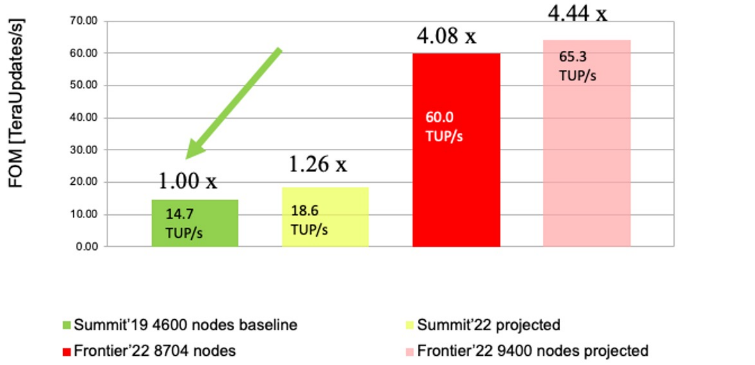


Motivation

- Need for high energy laser particle accelerators
- Applications in radiation therapy of cancer
- Fundamental studies of warm-dense matter and high-energy density physics.

Approach

- Uses alpaka - a C++17, templated metaprogramming
- Supports multi-threading and accelerators (OpenMP >4.5 + OpenACC + SYCL)
- Algorithmic improvements including optimized laser functor, new field background algorithm, new laser algorithm
- Numerous bugs filed and solutions worked out
- PIConGPU runs on Frontier, Summit, JUWELS, Perlmutter & others

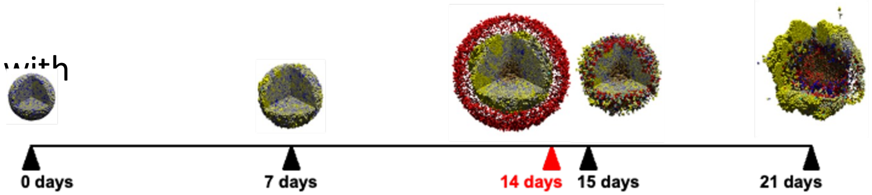


Leinhauser, Matthew, René Widera, Sergei Bastrakov, Alexander Debus, Michael Bussmann, and Sunita Chandrasekaran. "Metrics and design of an instruction roofline model for AMD GPUs." ACM Transactions on Parallel Computing 9, no. 1 (2022): 1-14.

Accelerating a Bio Physics (PhysiCell) Code on NVIDIA A100s

CPU-only: 9 hours 30 min

GPU: 1 hour 42 min



Motivation

- For modeling complex multiscale biological systems with many cell types
- Modeling cell behaviors vary with conditions
- Allow 3D multiscale simulations of cancer and diseases

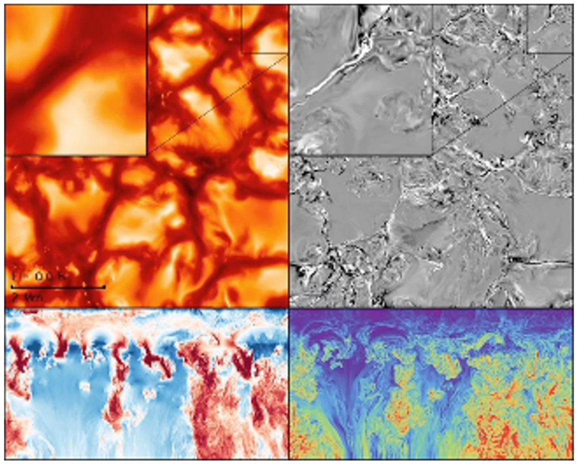
Approach

- Uses OpenACC Directive-based programming model
- Profiled code using NSight Sys and Compute
- Moved compute-intensive functions to GPUs
- Original algorithm preserved while acceleration
- NVIDIA HPC SDK 21.3, A100 GPUs; 37.5X better than single core
- Enabling many long simulations - explore dynamics, forecast disease progression over weeks and months

Sim Dataset	60 Sim minutes	180 sim minutes	360 Sim minutes
OMP CPU 1 core	524.6083s	1511.1268s	3107.043s
OMP CPU 64 cores	66.0669s	201.9457s	404.9028s
ACC CPU 64 cores	57.993s	167.4116s	330.3394s
Manual GPU V100	94.2378s	159.4965s	257.9657s
Manual GPU A100	140.6413s	216.9927s	325.707s
Managed GPU V100	23.903s	57.4191s	107.7914s
Managed GPU A100	21.3251s	45.9034s	82.7607s

Matt Stack, Paul MacLin, Robert Searles, Sunita Chandrasekaran, “OpenACC Acceleration of an Agent-Based Biological Simulation Framework” IEEE CiSE

Accelerating a Solar Physics (MURaM) Code on NVIDIA A100s

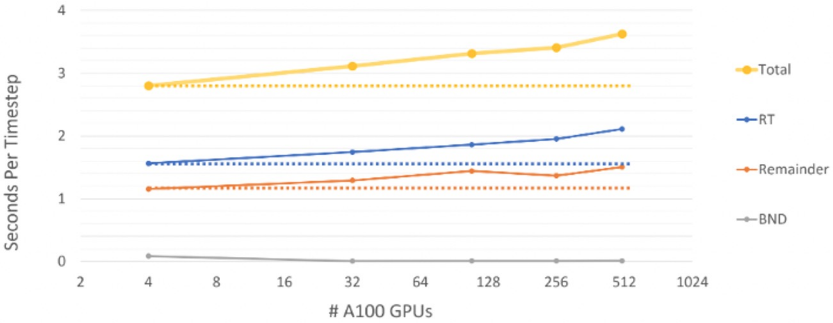


Motivation

- Enabling the study of scaling of MURaM on large scale Machines
- Accelerate radiation transport function is critical as it corresponds to some of the high resolution simulations of the photosphere

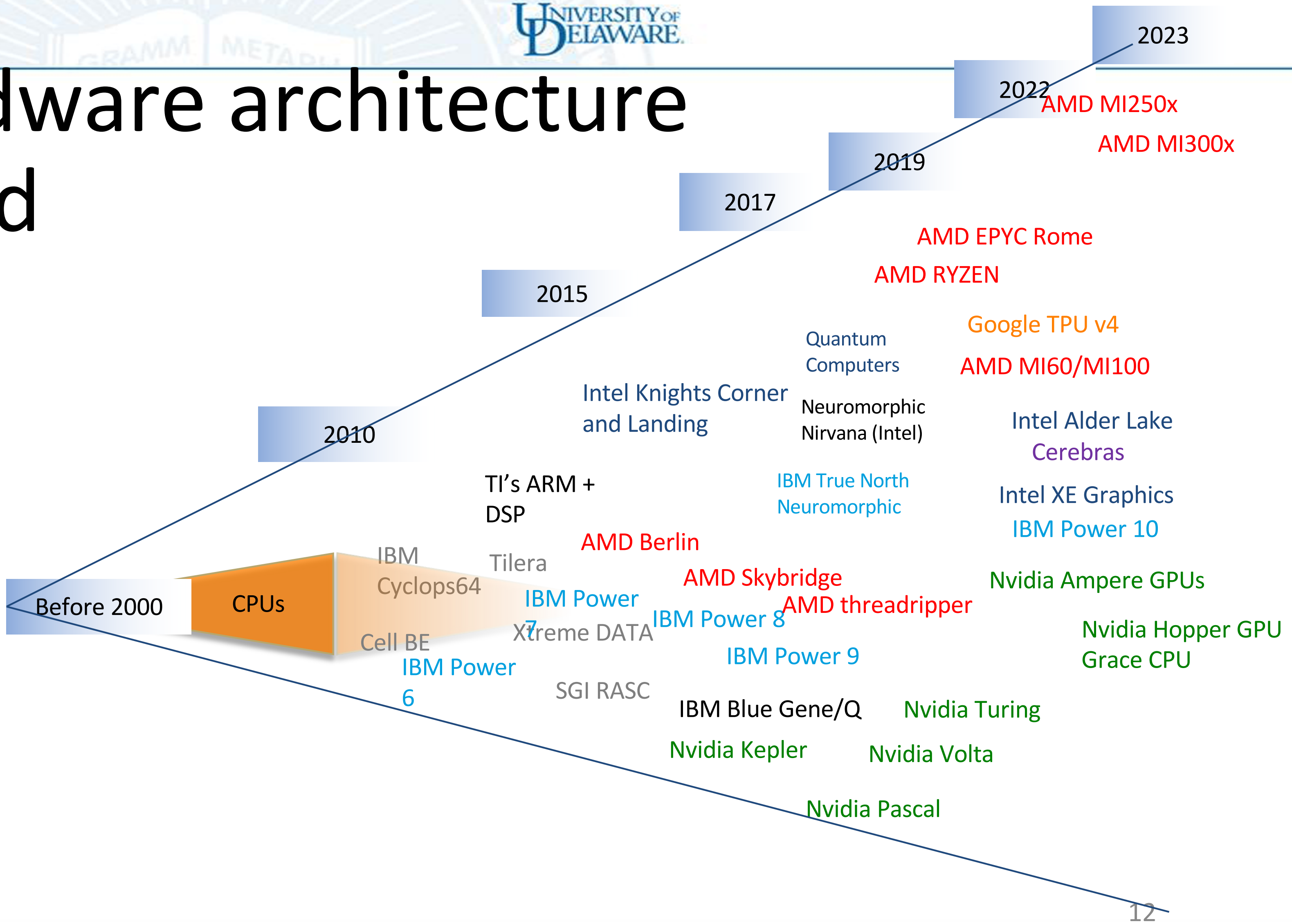
Approach

- Uses OpenACC Directive-based programming model
- Profiled code using NSight Sys and Compute
- Moved compute-intensive functions to GPUs
- Several other code enhancements were made
- NVIDIA HPC SDK 21.3, A100 GPUs; Weak Scaling
- 1 A100 GPU as much throughput as 90-100 CPU cores



Eric Wright, Cena Miller, Damien Przybylski, Matthias Rempel, Shiquan Su, Supreeth Suresh, Rich Loft, Sunita Chandrasekaran. Refactoring the MPS/University of Chicago Radiative MHD(MURaM) Model for GPU/CPU Performance Portability UsingOpenACC Directives. In Proceedings of the Platform for Advanced Scientific Computing Conference (PASC), pp. 1-12. 2021. https://dl.acm.org/doi/abs/10.1145/3468267.3470576

Hardware architecture trend



Before 2000

CPUs

2010

2015

2017

2019

2022

2023

IBM Cyclops64

Cell BE

IBM Power 6

Tilera

IBM Power 7

Xtreme DATA

SGI RASC

TI's ARM + DSP

AMD Berlin

Intel Knights Corner and Landing

AMD Skybridge

IBM Power 8

IBM Power 9

IBM Blue Gene/Q

Nvidia Kepler

Nvidia Pascal

Quantum Computers

Neuromorphic Nirvana (Intel)

IBM True North Neuromorphic

AMD threadripper

Nvidia Volta

Nvidia Turing

AMD EPYC Rome

AMD RYZEN

Google TPU v4

AMD MI60/MI100

Intel Alder Lake

Cerebras

Intel XE Graphics

IBM Power 10

Nvidia Ampere GPUs

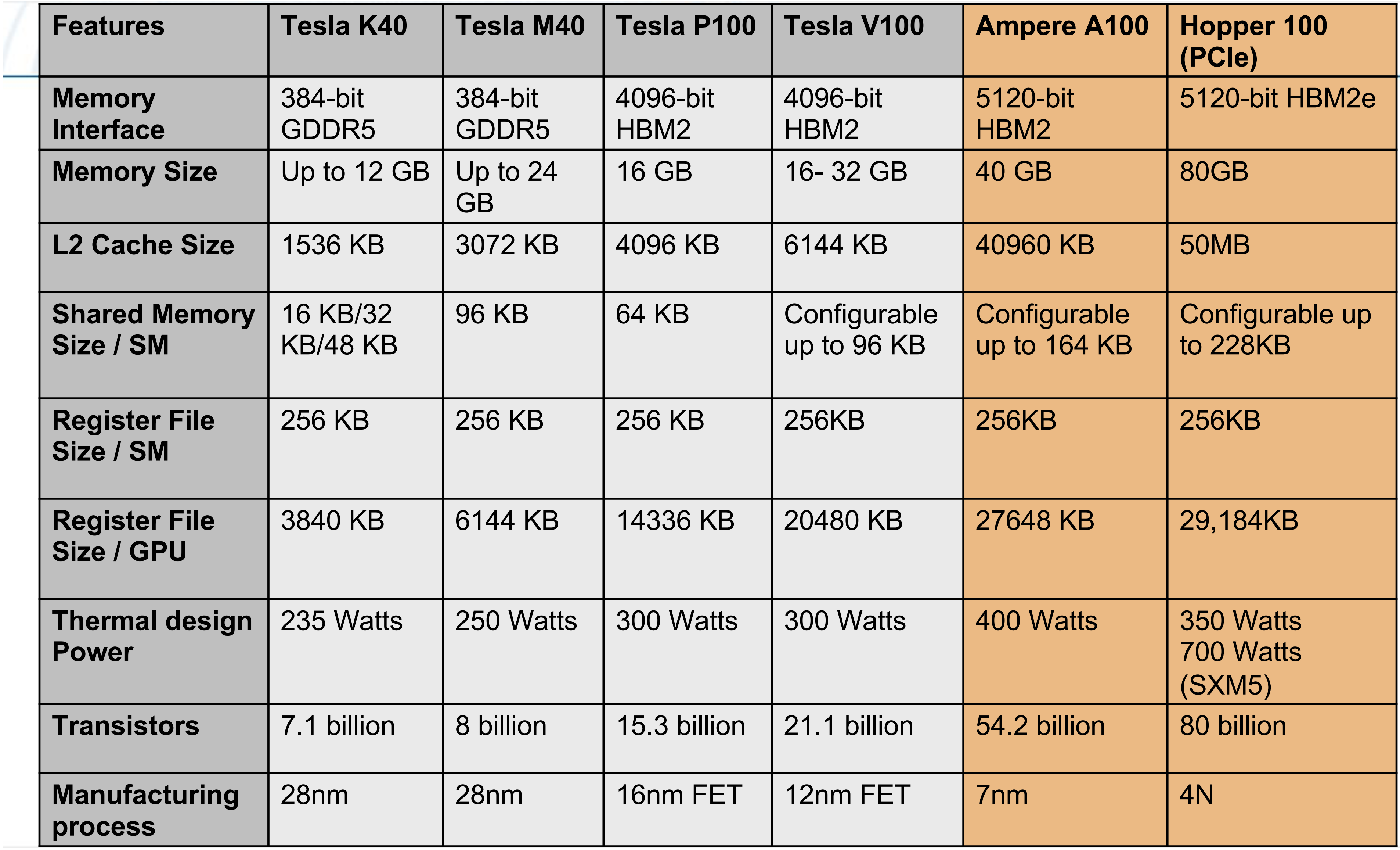
Nvidia Hopper GPU Grace CPU

AMD MI250x

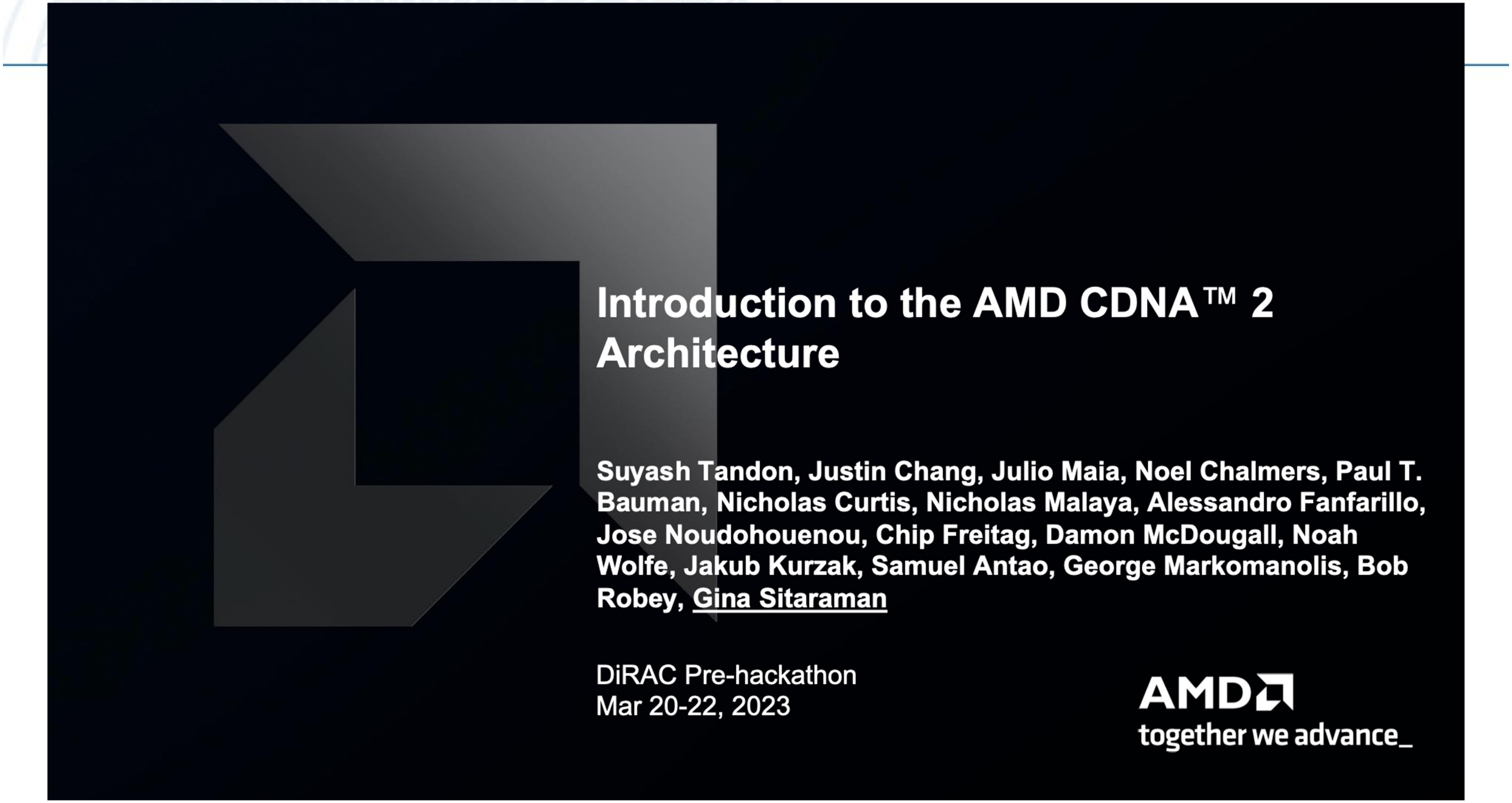
AMD MI300x

Basics of a GPU architecture

- NVIDIA GPUs
- AMD GPUs
- Intel GPUs



Features	Tesla K40	Tesla M40	Tesla P100	Tesla V100	Ampere A100	Hopper 100 (PCIe)
Memory Interface	384-bit GDDR5	384-bit GDDR5	4096-bit HBM2	4096-bit HBM2	5120-bit HBM2	5120-bit HBM2e
Memory Size	Up to 12 GB	Up to 24 GB	16 GB	16- 32 GB	40 GB	80GB
L2 Cache Size	1536 KB	3072 KB	4096 KB	6144 KB	40960 KB	50MB
Shared Memory Size / SM	16 KB/32 KB/48 KB	96 KB	64 KB	Configurable up to 96 KB	Configurable up to 164 KB	Configurable up to 228KB
Register File Size / SM	256 KB	256 KB	256 KB	256KB	256KB	256KB
Register File Size / GPU	3840 KB	6144 KB	14336 KB	20480 KB	27648 KB	29,184KB
Thermal design Power	235 Watts	250 Watts	300 Watts	300 Watts	400 Watts	350 Watts 700 Watts (SXM5)
Transistors	7.1 billion	8 billion	15.3 billion	21.1 billion	54.2 billion	80 billion
Manufacturing process	28nm	28nm	16nm FET	12nm FET	7nm	4N

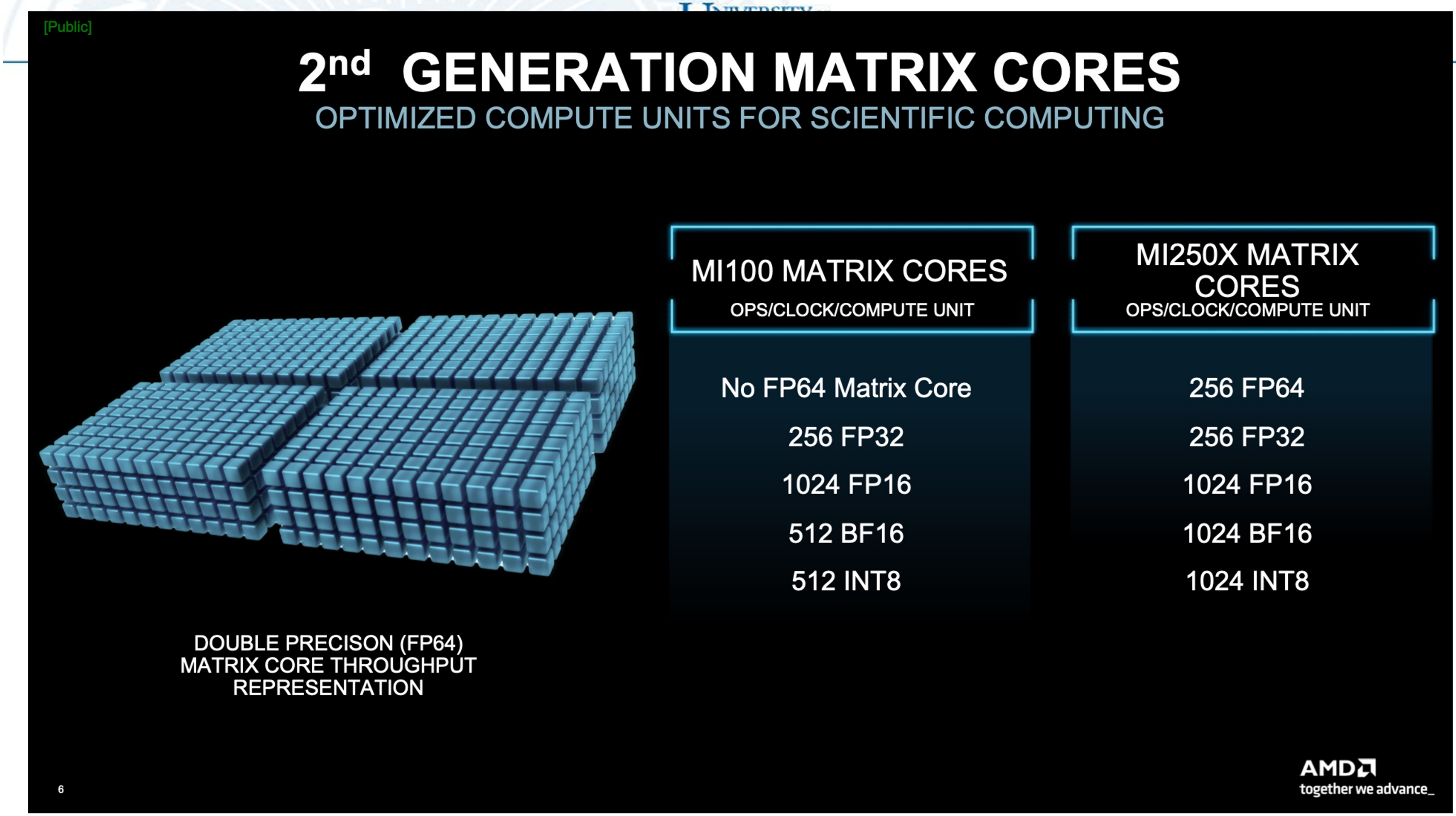


Introduction to the AMD CDNA™ 2 Architecture

Suyash Tandon, Justin Chang, Julio Maia, Noel Chalmers, Paul T. Bauman, Nicholas Curtis, Nicholas Malaya, Alessandro Fanfarillo, Jose Noudohouenou, Chip Freitag, Damon McDougall, Noah Wolfe, Jakub Kurzak, Samuel Antao, George Markomanolis, Bob Robey, Gina Sitaraman

DiRAC Pre-hackathon
Mar 20-22, 2023

AMD
together we advance_



[Public]

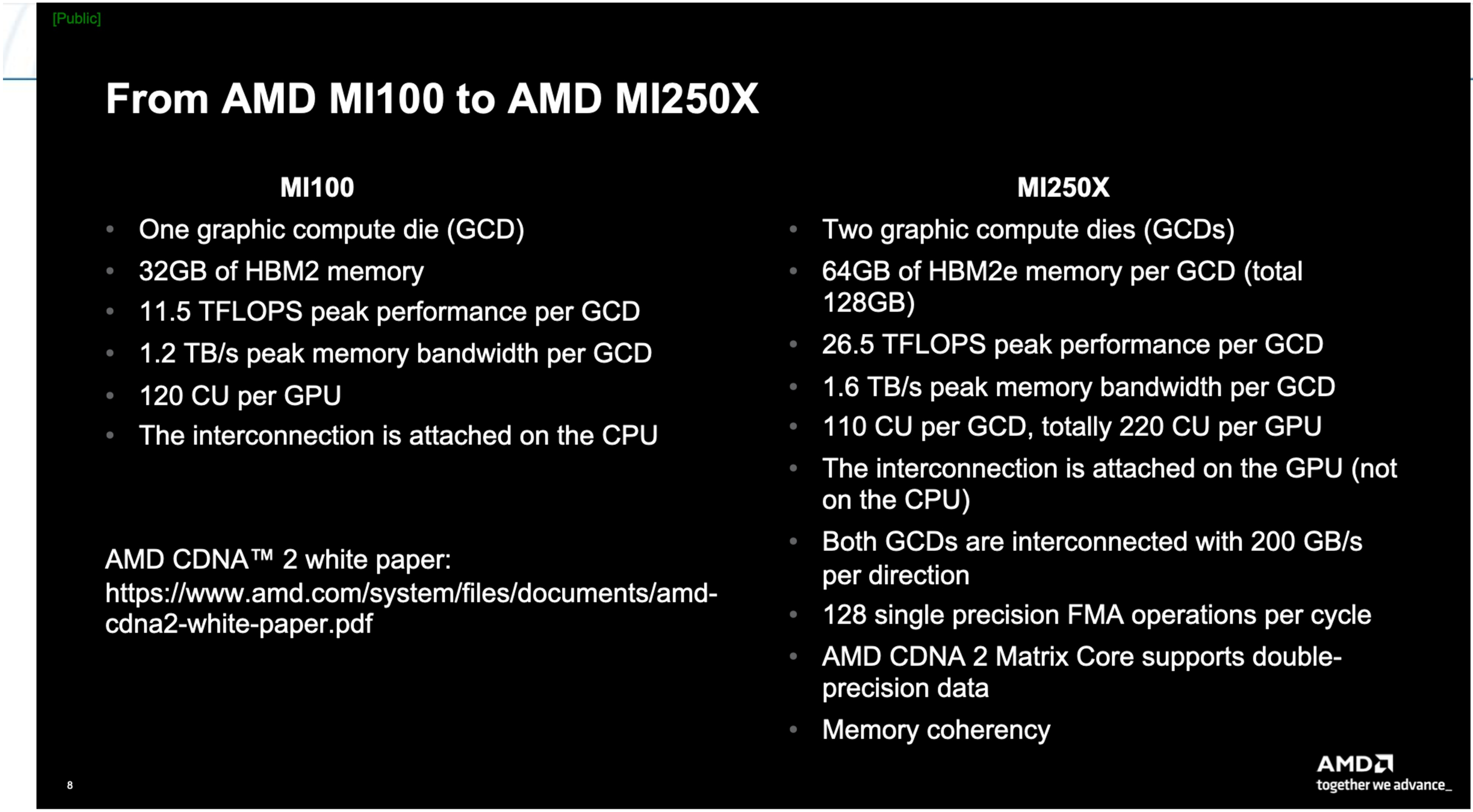
2nd GENERATION MATRIX CORES

OPTIMIZED COMPUTE UNITS FOR SCIENTIFIC COMPUTING

MI100 MATRIX CORES OPS/CLOCK/COMPUTE UNIT	MI250X MATRIX CORES OPS/CLOCK/COMPUTE UNIT
No FP64 Matrix Core	256 FP64
256 FP32	256 FP32
1024 FP16	1024 FP16
512 BF16	1024 BF16
512 INT8	1024 INT8

DOUBLE PRECISON (FP64)
MATRIX CORE THROUGHPUT
REPRESENTATION

AMD
together we advance_



[Public]

From AMD MI100 to AMD MI250X

MI100

- One graphic compute die (GCD)
- 32GB of HBM2 memory
- 11.5 TFLOPS peak performance per GCD
- 1.2 TB/s peak memory bandwidth per GCD
- 120 CU per GPU
- The interconnection is attached on the CPU

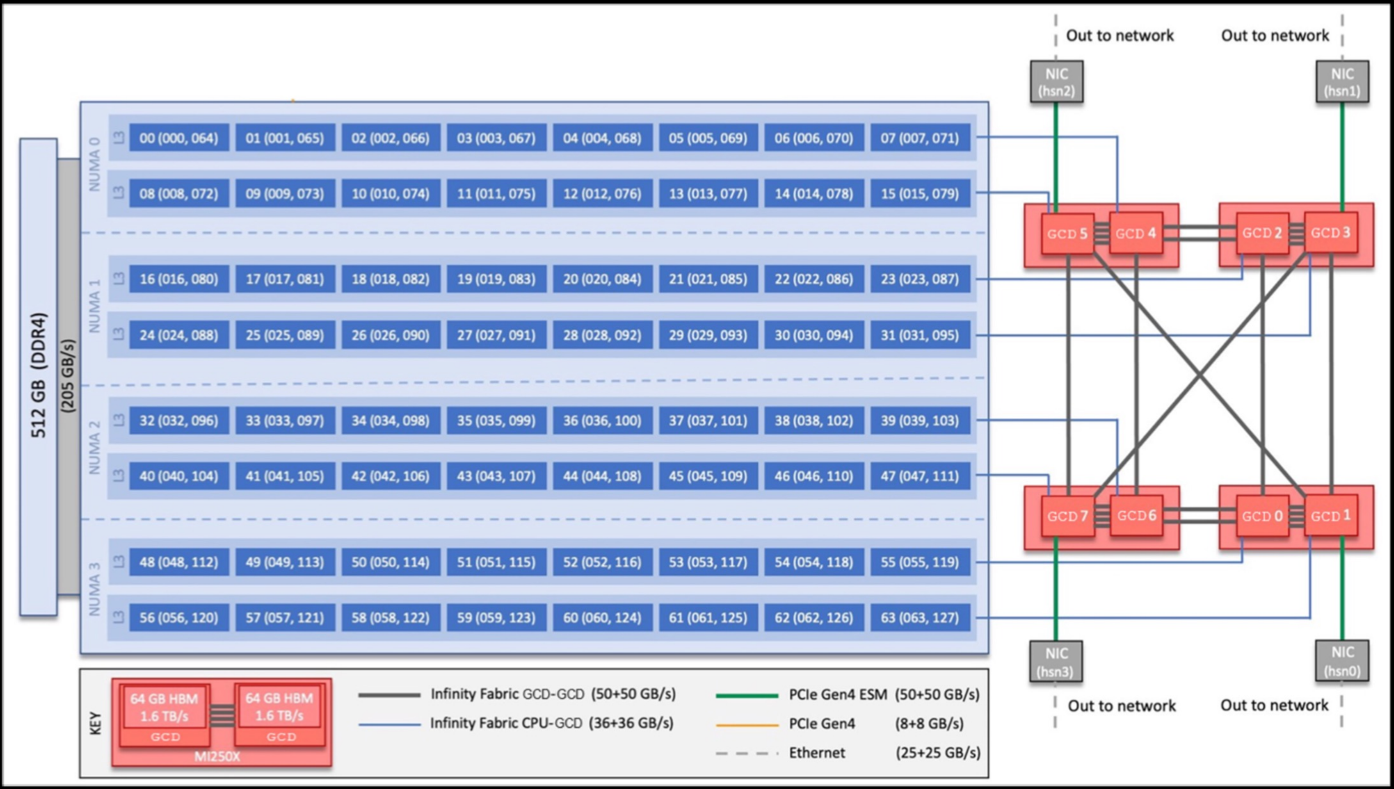
AMD CDNA™ 2 white paper: https://www.amd.com/system/files/documents/amd-cdna2-white-paper.pdf

MI250X

- Two graphic compute dies (GCDs)
- 64GB of HBM2e memory per GCD (total 128GB)
- 26.5 TFLOPS peak performance per GCD
- 1.6 TB/s peak memory bandwidth per GCD
- 110 CU per GCD, totally 220 CU per GPU
- The interconnection is attached on the GPU (not on the CPU)
- Both GCDs are interconnected with 200 GB/s per direction
- 128 single precision FMA operations per cycle
- AMD CDNA 2 Matrix Core supports double-precision data
- Memory coherency

AMD
together we advance_

MI250X Node Architecture

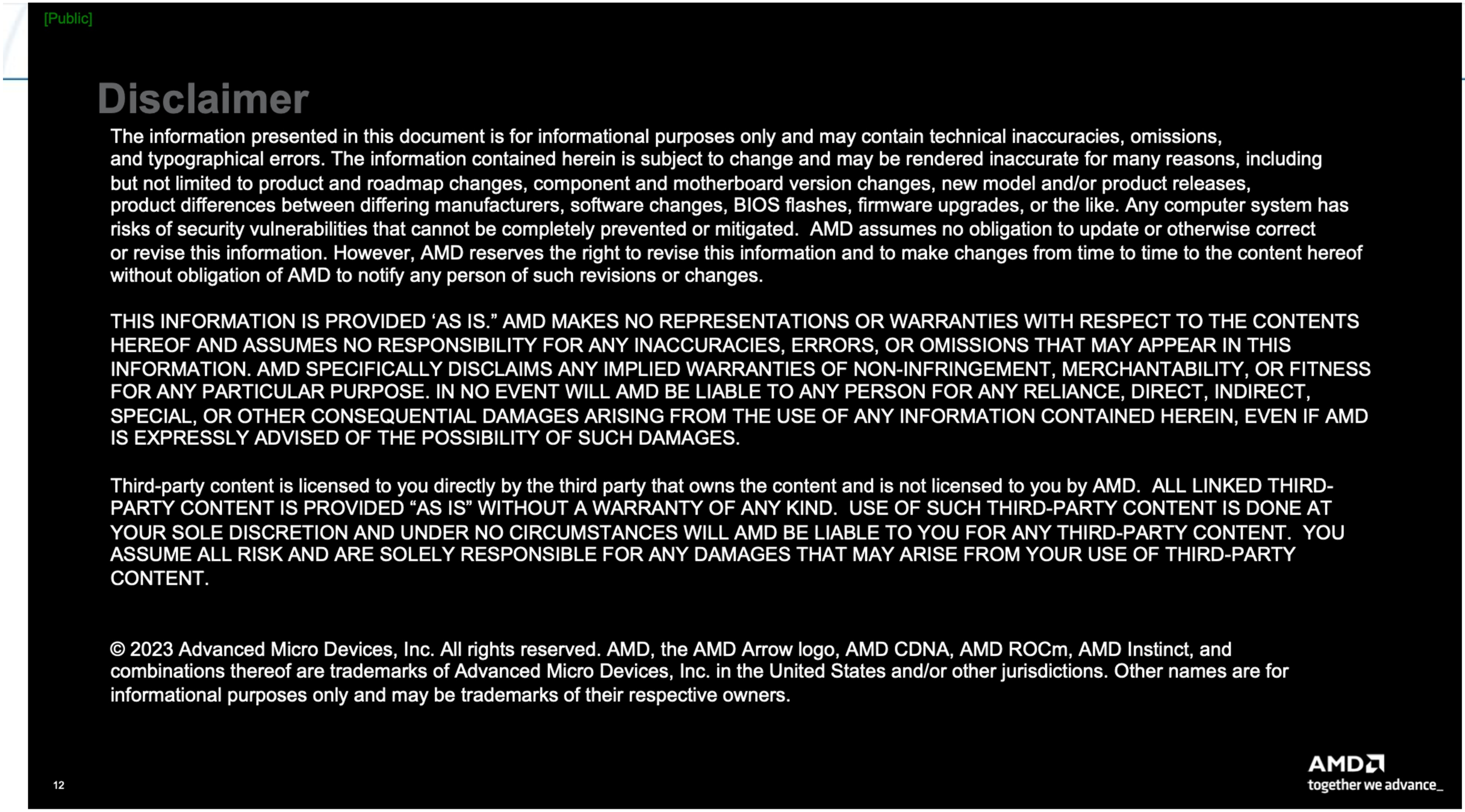


NUMA 0	L3	00 (000, 064)	01 (001, 065)	02 (002, 066)	03 (003, 067)	04 (004, 068)	05 (005, 069)	06 (006, 070)	07 (007, 071)
	L3	08 (008, 072)	09 (009, 073)	10 (010, 074)	11 (011, 075)	12 (012, 076)	13 (013, 077)	14 (014, 078)	15 (015, 079)
NUMA 1	L3	16 (016, 080)	17 (017, 081)	18 (018, 082)	19 (019, 083)	20 (020, 084)	21 (021, 085)	22 (022, 086)	23 (023, 087)
	L3	24 (024, 088)	25 (025, 089)	26 (026, 090)	27 (027, 091)	28 (028, 092)	29 (029, 093)	30 (030, 094)	31 (031, 095)
NUMA 2	L3	32 (032, 096)	33 (033, 097)	34 (034, 098)	35 (035, 099)	36 (036, 100)	37 (037, 101)	38 (038, 102)	39 (039, 103)
	L3	40 (040, 104)	41 (041, 105)	42 (042, 106)	43 (043, 107)	44 (044, 108)	45 (045, 109)	46 (046, 110)	47 (047, 111)
NUMA 3	L3	48 (048, 112)	49 (049, 113)	50 (050, 114)	51 (051, 115)	52 (052, 116)	53 (053, 117)	54 (054, 118)	55 (055, 119)
	L3	56 (056, 120)	57 (057, 121)	58 (058, 122)	59 (059, 123)	60 (060, 124)	61 (061, 125)	62 (062, 126)	63 (063, 127)

- 64 cores on a single socket CPU
- 4 MI250X GPUs, each with 2 GCDs
  - Each GCD is presented as a GPU device to `rocm-smi`
- 512 GB of DDR4 RAM
- Infinity Fabric™ links between GCDs and between GCDs and CPU cores
- 4 NICs attached to odd numbered GCDs

Courtesy: https://docs.olcf.ornl.gov/systems/frontier_user_guide.html#frontier-compute-nodes

AMD
together we advance_



Disclaimer

The information presented in this document is for informational purposes only and may contain technical inaccuracies, omissions, and typographical errors. The information contained herein is subject to change and may be rendered inaccurate for many reasons, including but not limited to product and roadmap changes, component and motherboard version changes, new model and/or product releases, product differences between differing manufacturers, software changes, BIOS flashes, firmware upgrades, or the like. Any computer system has risks of security vulnerabilities that cannot be completely prevented or mitigated. AMD assumes no obligation to update or otherwise correct or revise this information. However, AMD reserves the right to revise this information and to make changes from time to time to the content hereof without obligation of AMD to notify any person of such revisions or changes.

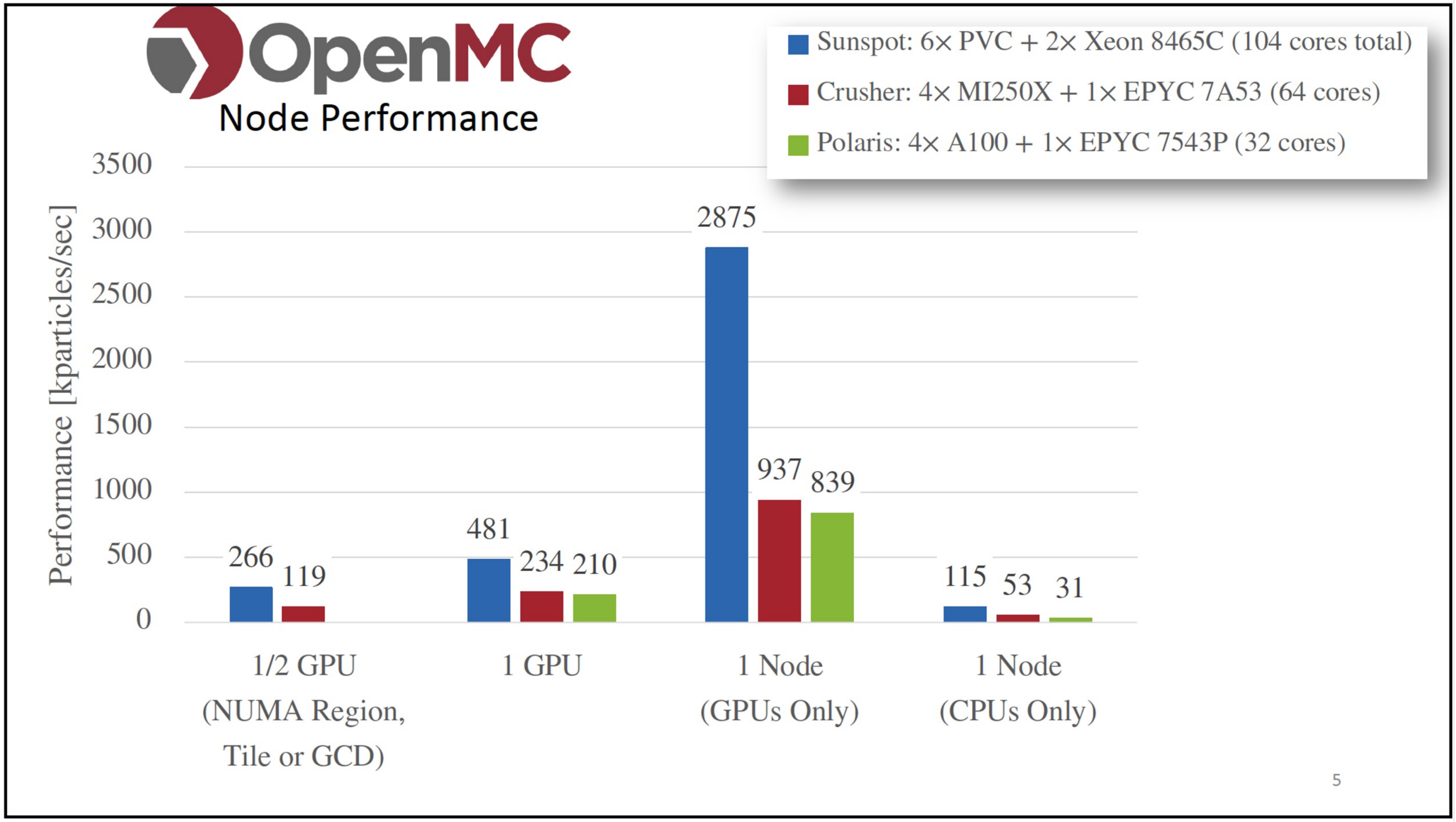
THIS INFORMATION IS PROVIDED 'AS IS." AMD MAKES NO REPRESENTATIONS OR WARRANTIES WITH RESPECT TO THE CONTENTS HEREOF AND ASSUMES NO RESPONSIBILITY FOR ANY INACCURACIES, ERRORS, OR OMISSIONS THAT MAY APPEAR IN THIS INFORMATION. AMD SPECIFICALLY DISCLAIMS ANY IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, OR FITNESS FOR ANY PARTICULAR PURPOSE. IN NO EVENT WILL AMD BE LIABLE TO ANY PERSON FOR ANY RELIANCE, DIRECT, INDIRECT, SPECIAL, OR OTHER CONSEQUENTIAL DAMAGES ARISING FROM THE USE OF ANY INFORMATION CONTAINED HEREIN, EVEN IF AMD IS EXPRESSLY ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Third-party content is licensed to you directly by the third party that owns the content and is not licensed to you by AMD. ALL LINKED THIRD-PARTY CONTENT IS PROVIDED "AS IS" WITHOUT A WARRANTY OF ANY KIND. USE OF SUCH THIRD-PARTY CONTENT IS DONE AT YOUR SOLE DISCRETION AND UNDER NO CIRCUMSTANCES WILL AMD BE LIABLE TO YOU FOR ANY THIRD-PARTY CONTENT. YOU ASSUME ALL RISK AND ARE SOLELY RESPONSIBLE FOR ANY DAMAGES THAT MAY ARISE FROM YOUR USE OF THIRD-PARTY CONTENT.

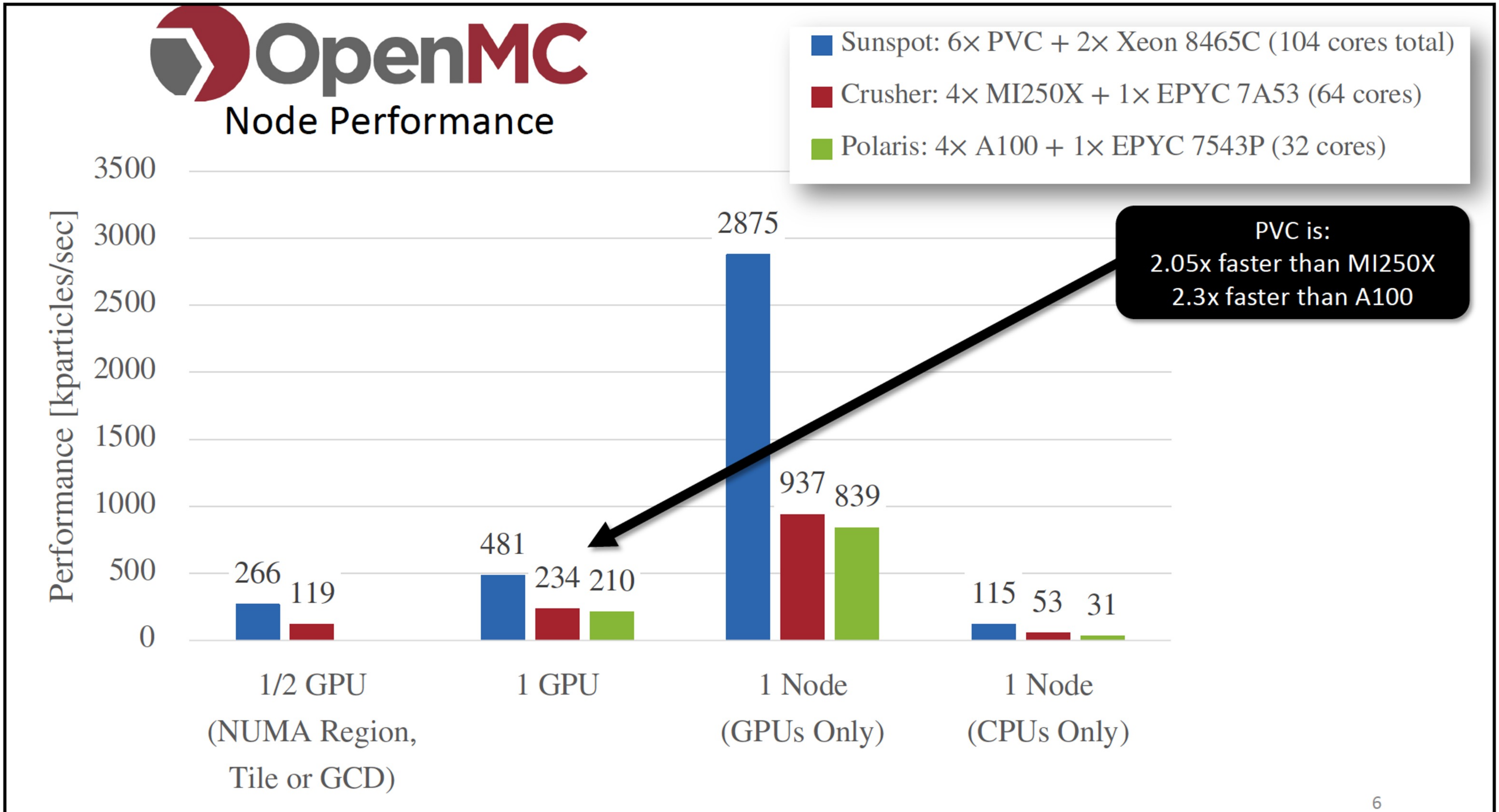
Intel GPUs

- The Aurora supercomputer at Argonne National Laboratory is now fully equipped with all
  - 10,624 compute blades, boasting 63,744 Intel® Data Center GPU “PVC” Max Series - Ponte Vecchio``
  - 21,248 Intel® Xeon® CPU Max Series processors.
  - 2 times the performance of AMD MI250X GPUs on OpenMC, and near linear scaling up to hundreds of nodes
- PVC GPU
  - 8 slices, 128 Xe-cores, 128 ray tracing units, 8 hardware contexts, 8 HBM2e controllers, and 16 Xe-Links, 400MB L2, 64KB l1, 128GB memory, 3,277GB/s Bandwidth, 52.43 TFLOPS FP16(half)

OpenMC application performance

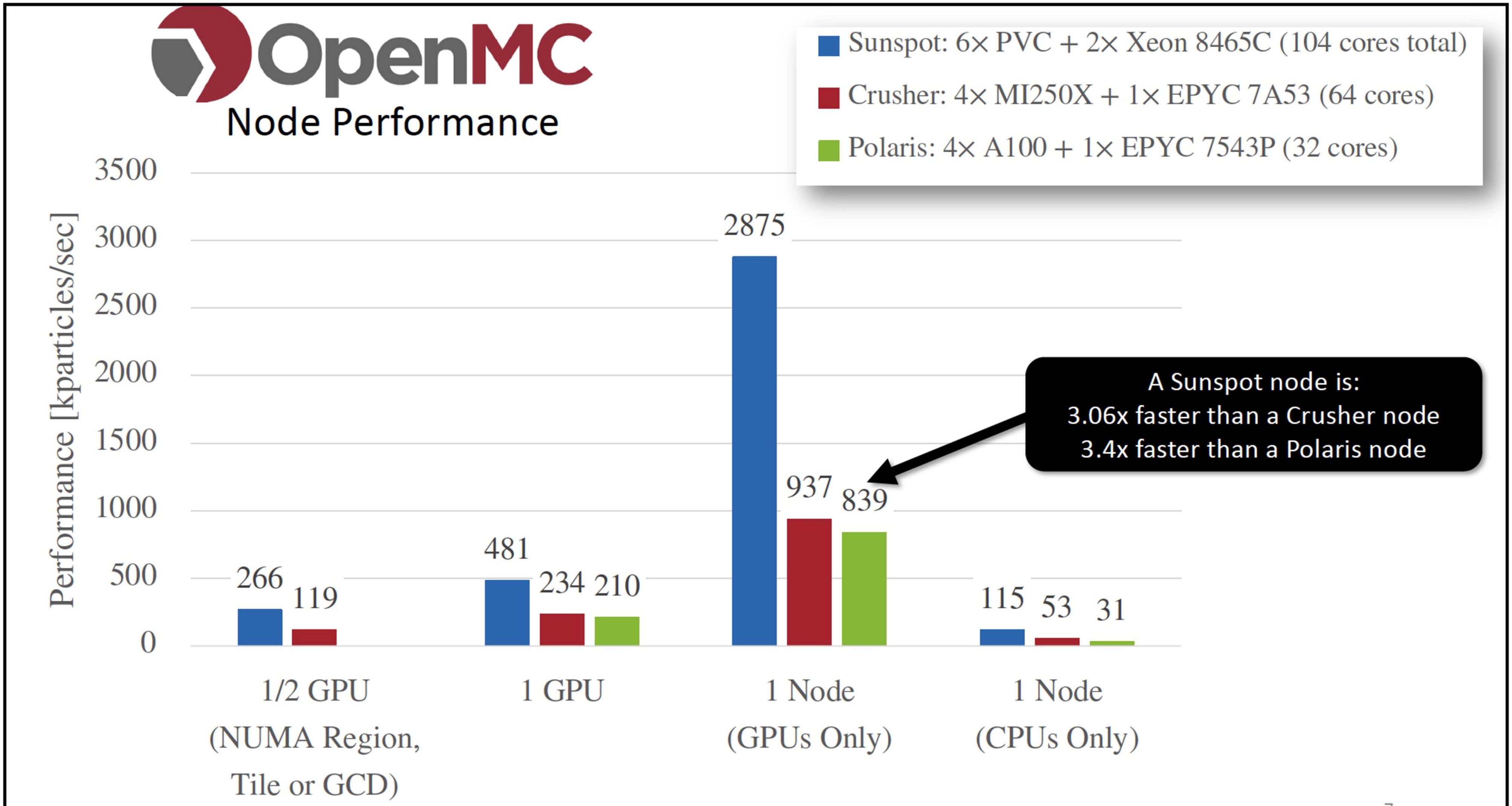


Node Performance



PVC is:
2.05x faster than MI250X
2.3x faster than A100

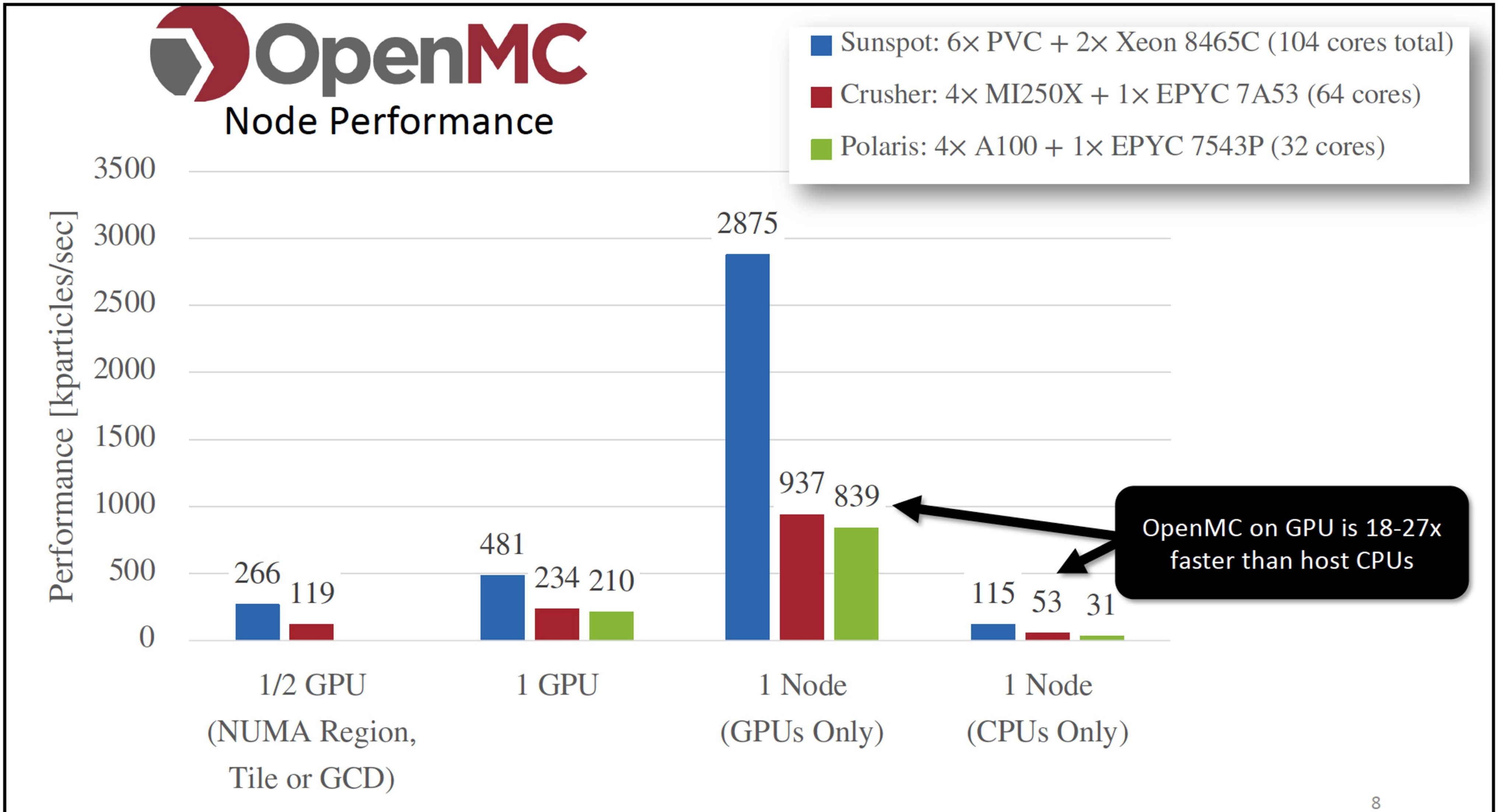
OpenMC
Node Performance



A Sunspot node is:
3.06x faster than a Crusher node
3.4x faster than a Polaris node

OpenMC

Node Performance



OpenMC on GPU is 18-27x faster than host CPUs