LA-UR-24-32608

Approved for public release; distribution is unlimited.

Title:	Parthenon
Author(s):	Ryan, Benjamin Ransom; Miller, Jonah Maxwell; Grete, Philipp; Roberts, Luke Forrest; Prather, Benjamin Scott; Prajapati, Nirmal Amrutbhai; Mullen, Patrick Dean; Long, Alex Roberts; Glines, Forrest; Sorkin, Marissa
Intended for:	TACC Open Hackathon, 2024-10-08 (Austin, Texas, UNITED STATES)
Issued:	2024-11-25

Parthenon

Ben Ryan (LANL)

Philipp Grete (Hamburg Obs.)

Ben Prather (LANL)

Patrick Mullen (LANL)

Forrest Glines (NVIDIA)

Jonah Miller (LANL)

Luke Roberts (LANL)

Nirmal Prajapati (LANL)

Alex Long (LANL)

Marissa Sorkin (NVIDIA)

Team Members

Mentors

OpenACC
More Science, Less Programming

OPEN HACKATHONS

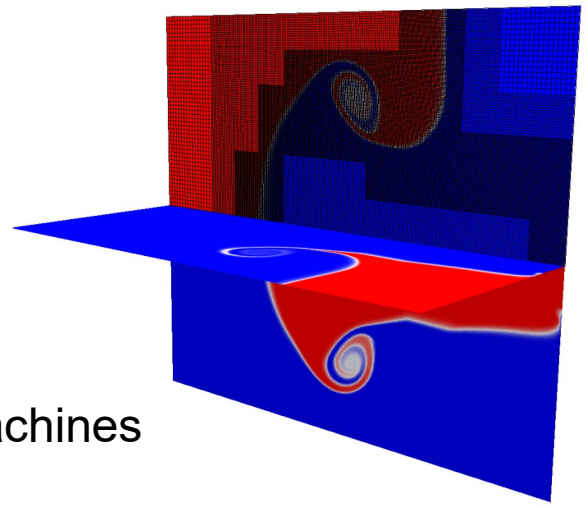
Parthenon

Parthenon* is an open-source** performance portability library for high performance multiphysics applications written in C++.

Provides data abstractions that support block-adaptive mesh refinement and domain decomposition for physics algorithms that use cell-, face-, edge-centered fields, explicit and implicit solvers, and particle methods.

Dependencies:

- Kokkos for performance portability (CUDA backend)
- MPI for inter-/intra-node parallelism
- HDF5 for I/O (openPMD support in development)
- Catch2 for testing



Here we are focusing on improving parallel scaling on GPU HPC machines

Downstream applications: AthenaPK, Phoebus, Artemis, RIOT, Jaybenne, …

OpenACC More Science, Less Programming

OPEN HACKATHONS

*Grete, P. et al., The International Journal of High Performance Computing Applications, 2023; 37(5):465-486

**https://github.com/parthenon-hpc-lab/parthenon

Goals

Major

- Improving parallel scaling on many (~100+) GPUs (now ~70% in most downstream codes). There's good evidence (AthenaPK) we can get to 90%+ efficiency.

Minor

- Improving single-node single large meshblock performance
- Identifying scaling bottlenecks in particles infrastructure

Parthenon

Ben Ryan (LANL)

Jonah Miller (LANL)

Philipp Grete (Hamburg Obs.)

Luke Roberts (LANL)

Ben Prather (LANL)

Nirmal Prajapati (LANL)

Patrick Mullen (LANL)

Alex Long (LANL)

Forrest Glines (NVIDIA)

Marissa Sorkin (NVIDIA)

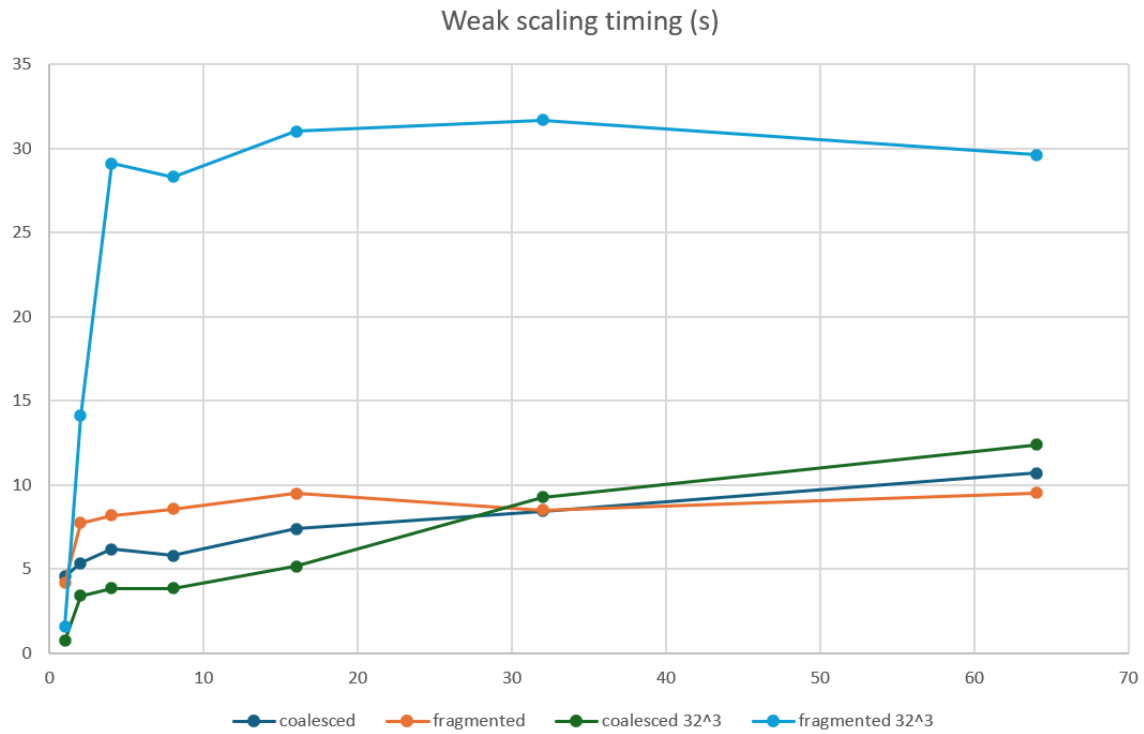
Team Members

Mentors

OpenACC
More Science, Less Programming

OPEN HACKATHONS

Profiler Output

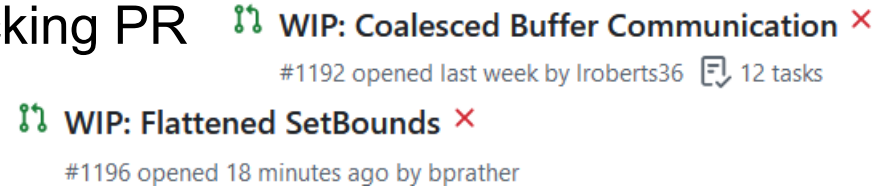


8 x 128^3 block
SetBounds 0.32 s
SendBoundBufs 0.26 s

1 x 256^3 block
SetBounds 1.13 s
SendBoundBufs 0.96 s

Progress and Goals

- What have you accomplished?
  - Advection example with tunable fragmentation of MPI communications
  - First draft of coalesced buffer packing PR
  - Single-level buffer packing draft
- What are your goals for today?
  - Fix buffer packing PR
    - Measure scaling efficiency on advection problem
  - Fix single-level buffer packing PR
  - Profile particles example at scale
  - Fix memory leak due to View of Views



Problems and Solutions

- What problems are you currently facing?
  - MPI performance quite poor for our application with UCX workarounds; our performance trends are qualitative

- Have you resolved any problems (or found bugs) that others might find useful?
  - Kokkos View of Views memory leak/failure in Kokkos 4.4+

Parthenon

Ben Ryan (LANL)

Jonah Miller (LANL)

Philipp Grete (Hamburg Obs.)

Luke Roberts (LANL)

Ben Prather (LANL)

Nirmal Prajapati (LANL)

Patrick Mullen (LANL)

Alex Long (LANL)

Forrest Glines (NVIDIA)

Marissa Sorkin (NVIDIA)

Team Members

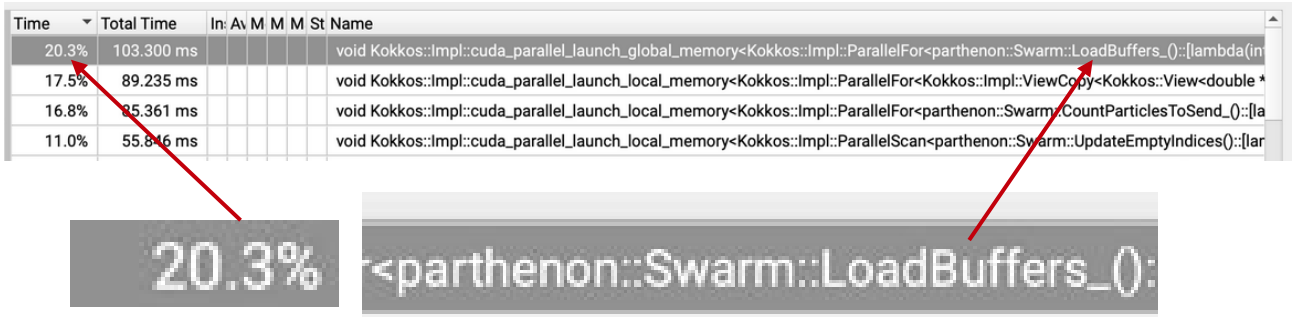
Mentors

OpenACC
More Science, Less Programming

OPEN HACKATHONS

|-> 3.19e-01 sec 4.2% 99.4% 0.0% 0.6% 6.28e+02 200 boundary_communication.cpp::252::SetBounds

Profiler Output



Latency Issue

This kernel exhibits low compute throughput and memory bandwidth utilization relative to the peak performance of this device. Achieved compute throughput and/or memory bandwidth below 60.0% of peak typically indicate latency issues. Look at Scheduler Statistics and Warp State Statistics for potential reasons.

The following table lists the metrics that are key performance indicators:

Metric Name	Value	Guidance
gpu__compute_memory_throughput.avg.pct_of_peak_sustained_elapsed	20.005	20.005 < 80.000
sm__throughput.avg.pct_of_peak_sustained_elapsed	3.42045	3.420 < 80.000

Progress and Goals

- What have you accomplished?
  - Coalesced MPI messaging PR cycles on multiple GPUs
    - Confirmed scaling issue with RDMA support on Jedi G-H machine
  - Flattened boundary function cycling
  - Identified performance bottleneck in particles example (atomics in buffer packing)
- What are your goals for today?
  - Profile coalesced MPI messaging PR
  - Profile flattened boundary function PR
  - Rewrite particles buffer packing code, measure improvements

Problems and Solutions

- What problems are you currently facing?
  - How to sort particles into multiple buffers without atomics (or with atomics, performantly)
  - Working on getting our code cycling with new UCX 1.17.x module
- Have you resolved any problems (or found bugs) that others might find useful?

Parthenon

Ben Ryan (LANL)

Jonah Miller (LANL)

Philipp Grete (Hamburg Obs.)

Luke Roberts (LANL)

Ben Prather (LANL)

Nirmal Prajapati (LANL)

Patrick Mullen (LANL)

Alex Long (LANL)

Forrest Glines (NVIDIA)

Marissa Sorkin (NVIDIA)

Team Members

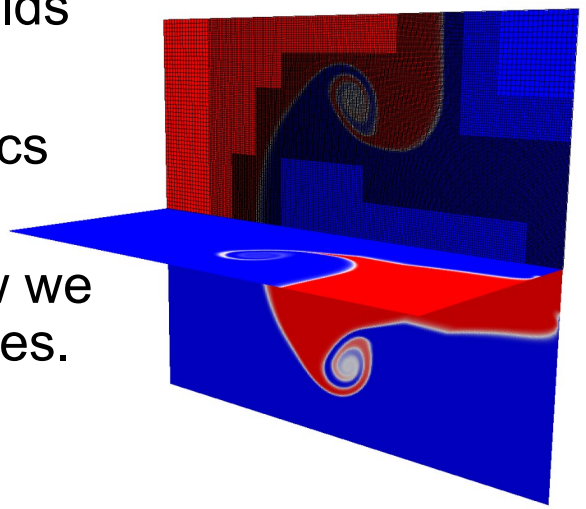
Mentors

OpenACC
More Science, Less Programming

OPEN HACKATHONS

Parthenon

- Parthenon* is an open-source** performance portability library for high performance multiphysics applications written in C++.
- Provides data abstractions that support domain-decomposed block-adaptive mesh refinement calculations for cell-based fields and particles
- Aim is to accelerate the process of writing complex multiphysics codes
- At Hackathon, focused on communications infrastructure: how we send messages, and how we pack and unpack those messages.



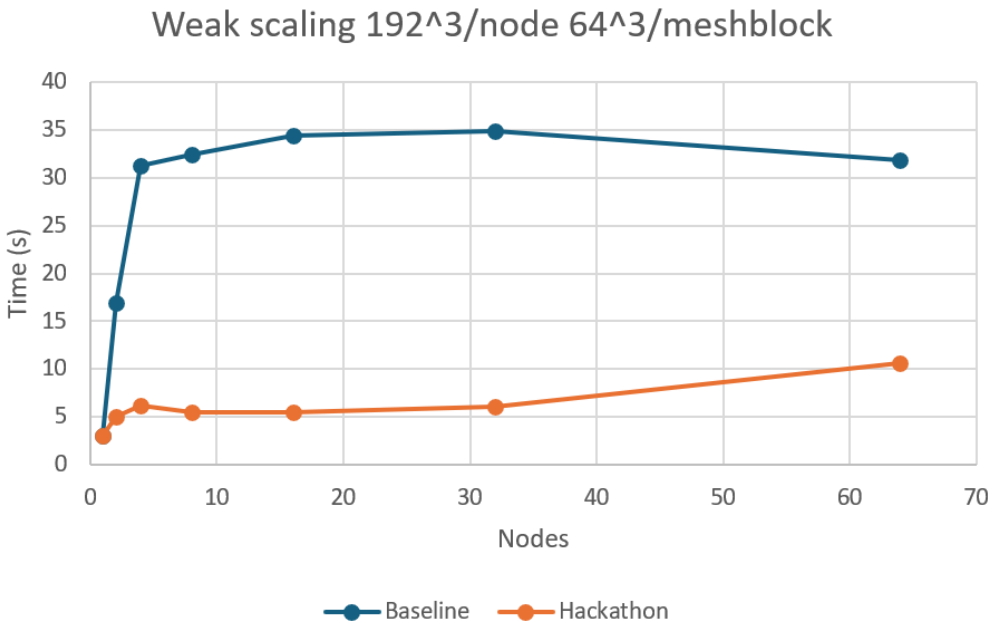
*Grete, P. et al., The International Journal of High Performance Computing Applications, 2023; 37(5):465-486

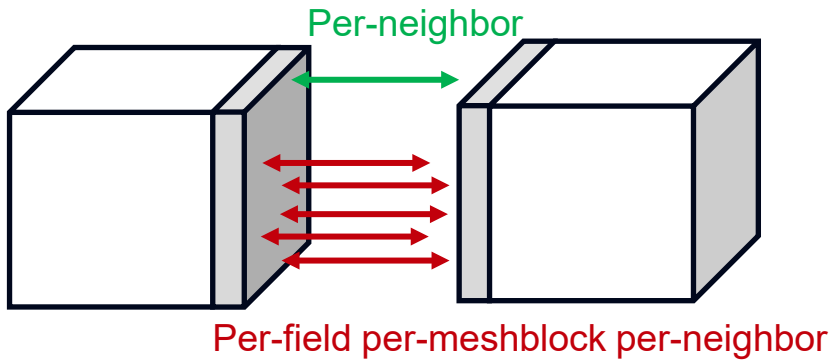
**https://github.com/parthenon-hpc-lab/parthenon

Evolution and Strategy

- Our goal: expose representative Parthenon examples to a modern system (Vista) and computing experts (NVIDIA, TACC), learn best practices for profiling
- We had initial ideas about what our issues were based on scaling properties of different implementations, kernel timings
- Applying Nsight to particles example identified most expensive kernel (particle buffer loading) and cause (stalled warps)

Results: Coalesced MPI Communication





- What were you able to accomplish?
  - Rewrote MPI communication to populate coalesced buffers for communication
  - ~3x speedup (qualitative) from 90x reduction in MPI messages
- What did you learn?
  - MPI is great but can't be taken completely for granted

OpenACC More Science, Less Programming
OPEN HACKATHONS

Energy Efficiency: Coalesced MPI Communication

INPUTS	
Baseline	GPU
Baseline GPU Type	4x H100 80GB SXM5
Baseline GPU # GPUs	1
Final GPU Node	4x H100 80GB SXM5
Final # GPUs	1
Application Speedup	3.0x

Node Replacement	3.0x

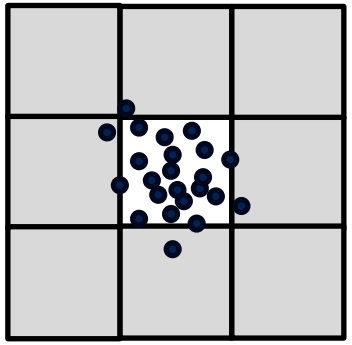
ANNUAL ENERGY SAVINGS PER GPU NODE			
	4x H100 80GB SXM5	4x H100 80GB SXM5	**Power Savings**
Compute Power (kWh/year)	109,167	36,389	72,778
Networking Power (kWh/year)	11,563	3,854	7,709
Total Power (kWh/year)	**120,730**	**40,243**	**80,487**

$/kWh	$ 0.18
Annual Cost Savings	$ 14,487.64
3-year Cost Savings	$ 43,462.92

OpenACC More Science, Less Programming

OPEN HACKATHONS

Results: Particle Buffer Loading



- What were you able to accomplish?
  - Rewrote particle buffer loading from atomic operations to list sorting
  - 15% speedup
- What did you learn?
  - High ratio of atomic writes to memory locations is bad
- Particle transport methods can use up a lot of cycles; 15% efficiency gain is a major win

Energy Efficiency: Particle Buffer Loading

INPUTS	
Baseline	GPU
Baseline GPU Type	GH200
Baseline GPU # GPUs	1
Final GPU Node	GH200
Final # GPUs	1
Application Speedup	1.2x

Node Replacement	1.2x

ANNUAL ENERGY SAVINGS PER GPU NODE			
	GH200	GH200	**Power Savings**
Compute Power (kWh/year)	11,847	10,302	1,545
Networking Power (kWh/year)	1,108	964	145
Total Power (kWh/year)	**12,955**	**11,265**	**1,690**

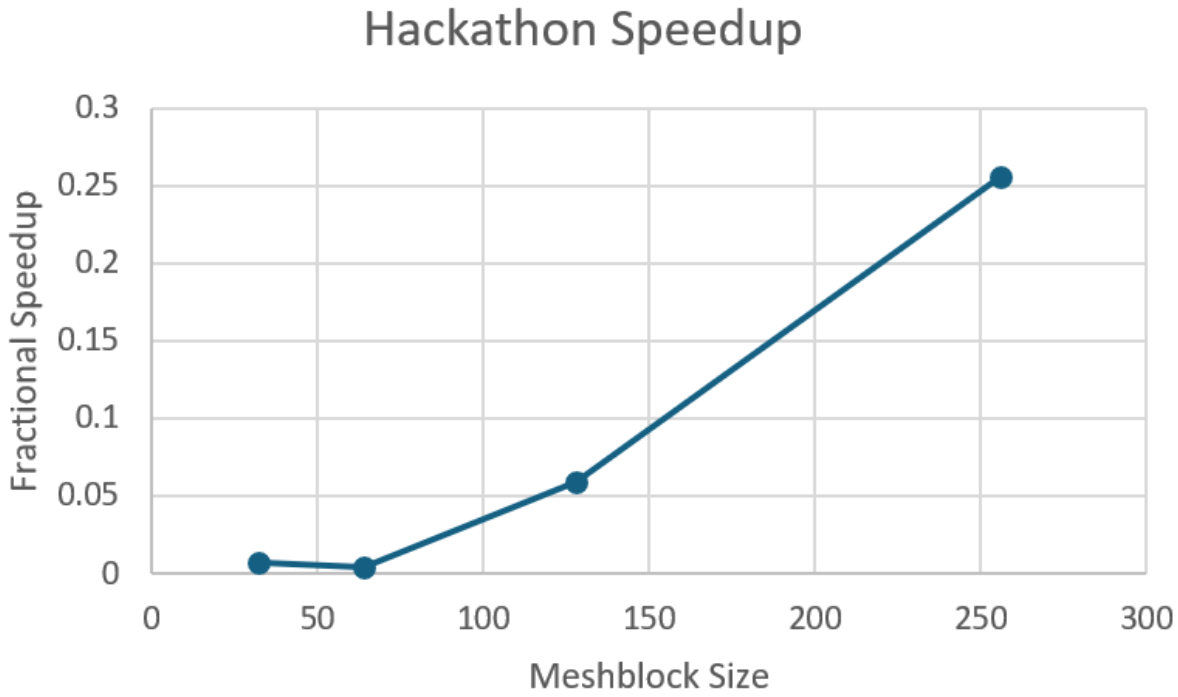
$/kWh	$	0.18
Annual Cost Savings	$	304.16
3-year Cost Savings	$	912.49

OpenACC More Science, Less Programming

OPEN HACKATHONS

Results: Flattened boundary updates

- What were you able to accomplish?
  - Rewrote buffer unpacking to increase number of teams
- What did you learn?
  - Hierarchical parallelism is sensitive to the structure of your data! (Many meshblocks versus single meshblocks)
  - Team size can be important (10% total speedup in AthenaPK by specifying team size in flux kernels)



OpenACC More Science, Less Programming

OPEN HACKATHONS

Energy Efficiency: Flattened boundary updates

INPUTS	
Baseline	GPU
Baseline GPU Type	GH200
Baseline GPU # GPUs	1
Final GPU Node	GH200
Final # GPUs	1
Application Speedup	1.3x

Node Replacement	1.3x

ANNUAL ENERGY SAVINGS PER GPU NODE			
	GH200	GH200	**Power Savings**
Compute Power (kWh/year)	12,877	10,302	2,575
Networking Power (kWh/year)	1,205	964	241
Total Power (kWh/year)	**14,082**	**11,265**	**2,816**

$/kWh	$ 0.18
Annual Cost Savings	$ 506.94
3-year Cost Savings	$ 1,520.82

OpenACC More Science, Less Programming

OPEN HACKATHONS

What problems have you encountered?

- RDMA issue on Vista meant we had poor absolute scaling

Wishlist

What do you wish existed to make your life easier?

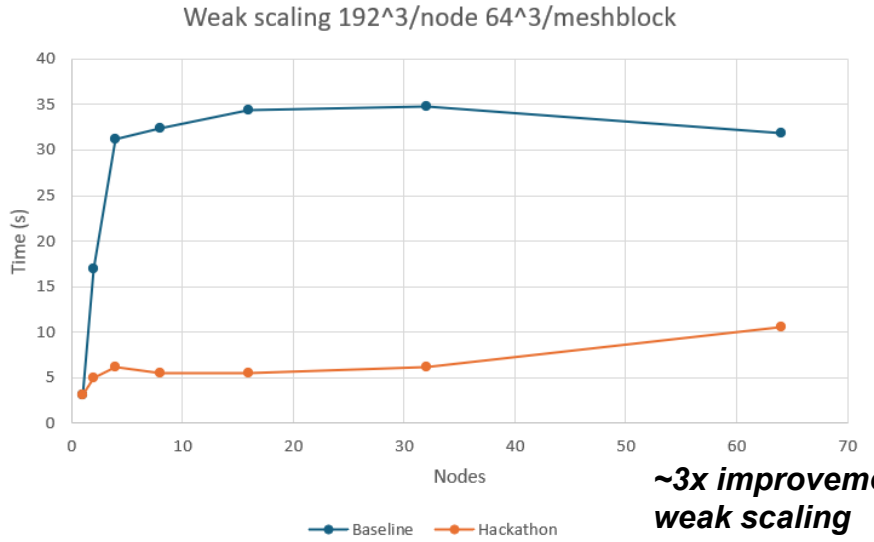
- A place to share reproducers to be looked at by experts when code/machine issues arise

Final Thoughts

- Was this Open Hackathon worth it? YES
  - Mentor interactions were very helpful
  - Focused time and location for the team to solve these problems
- Will you continue development?
  - Verify code changes in detail, merge
    - Support sparse fields with MPI changes
  - On to the next most concerning parts of the code
- What sustained resources or support will be critical for your work after the event?
  - Nsight tools

Application Background

- Parthenon is a library for writing performance-portable block-adaptive mesh refinement multiphysics codes
- Support for field- and particle-based methods
- Infrastructure must be very efficient



~3x improvement in weak scaling

Hackathon Objectives and Approach

- Refactor MPI communication to coalesce messages
- Refactor hierarchical parallelism to permit more teams
- Profile particles example, target slowest kernels

Technical Accomplishments and Impact

- ~3x speedup on weak scaling for ~30 meshblocks/node with improved messaging
- ~25% speedup for single meshblock/node case with more teams
- ~15% speedup for particle buffer loading with sorting rather than atomics

Project Summary

Please summarize your team's achievements during the Open Hackathon (100 words).

During the TACC Open Hackathon the Parthenon team focused on improving the parallel efficiency of fields and the serial efficiency of particles. From previous scaling and kernel timing data, we refactored our boundary communication to dramatically reduce the number of MPI messages (~3x improvement with 27 meshblocks per rank) and increase the number of teams used during boundary buffer unpacking (up to 25% improvement for single large meshblocks per rank). We used Nsight tools to profile a representative particles example application and identified particle buffer packing as the most expensive kernel, due to atomic operations; we refactored this single kernel to use sorting rather than atomics to speed up the whole application by 15%.

PROMOTING YOUR WORK: AVAILABLE OPPORTUNITIES

- **Papers and Talks:** Please acknowledge the Open Hackathons program and OpenACC Organization in any planned or upcoming papers, presentations, or talks.
  "This work was completed in part at the TACC Open Hackathon, part of the Open Hackathons program. The authors would like to acknowledge OpenACC-Standard.org for their support."
- **Social Media Support:** Please feel free to promote your participation across your social media channels. Tag **@OpenACCorg** and **#OpenHackathons** and we are happy to amplify.
- **Blogs and Technical Write-ups**: Create a blog post or technical article that highlights the work being done and results achieved.
- **Quotes and Testimonials:** Highlight your quote or feedback on our channels (i.e. social, website, etc.).

***Please reach out to Izumi Barker (ibarker@nvidia.com) to discuss marketing options and opportunities.**

Other Resources

Continue your learning journey. Keep engaged after the event. www.openhackathons.org

2024 OAC Summit

Call for Speakers is open for the Open Accelerated Computing Summit, December 10-12, 2024.

Additional Events

Continue to advance your project and build your skills at another of our scheduled events.

NVIDIA DLI

Resources for diverse training. Explore AI, accelerated computing, data science, graphics, and more.

Open Hackathons GitHub

Explore additional Bootcamp materials and resources