

FROM JUPITER TO THE STARS

EXPERIENCES AND RESULTS FROM EUROPE'S FIRST EXASCALE SYSTEM

7 October 2026 | Andreas Herten | Forschungszentrum Jülich, Jülich Supercomputing Centre

Content

- JUPITER
  - System Overview
  - Pictures
  - Preparation
    - Build-Up
    - Inauguration
  - Exascale!
  - Procurement
- Benchmark Suite
  - Benchmarks Overview
  - Benchmark Details
    - TCO Results
    - High-Scaling Results
  - exaCB
- Application Results
  - Overview
  - ICON
  - BEAST
  - JUQCS
- MPI/OpenMP Results
  - OSU at Scale
  - NCCL at Scale
  - LinkTest
  - MPI Init
  - LLVM OpenMP
  - Optimizations
- Conclusion

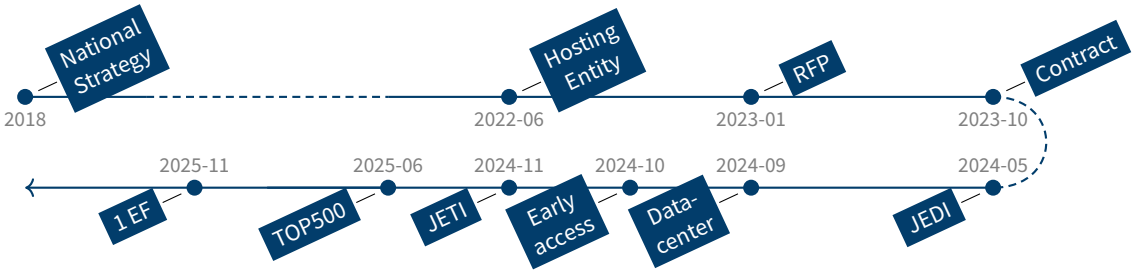
JÜLICH Forschungszentrum

About JUPITER

- **JUPITER**: First European exascale supercomputer
  - Procured by EuroHPC JU, BMFTR (Federal Ministry of Research, Technology, and Space), MKW (NRW Ministry of Culture and Science)
  - Hosting entity: Forschungszentrum Jülich / Jülich Supercomputing Centre for Gauß Center for Supercomputing (GCS)

About JUPITER

- **JUPITER**: First European exascale supercomputer
  - Procured by EuroHPC JU, BMFTR (Federal Ministry of Research, Technology, and Space), MKW (NRW Ministry of Culture and Science)
  - Hosting entity: Forschungszentrum Jülich / Jülich Supercomputing Centre for Gauß Center for Supercomputing (GCS)



JÜLICH Forschungszentrum

JUPITER System Overview

- ParTec/Eviden consortium
- Implementing Modular Supercomputing Architecture



JUPITER System Overview

- ParTec/Eviden consortium
- Implementing Modular Supercomputing Architecture
- JUPITER Booster: High scalability, 1 EFLOP/s HPL, $>$ 40 EFLOP/s FP8
  5884 nodes: 4$\times$ Grace-Hopper superchip, 4$\times$ network
- JUPITER Cluster: High versatility, high memory-to-compute ratio
  582 nodes: 2$\times$ AMD Turin 9655 (2 $\times$ 96 cores; DDR5; 1$\times$ network)
  162 nodes: 2$\times$ SiPearl Rhea1 (2 $\times$ 80 cores; HBM; 1$\times$ network)



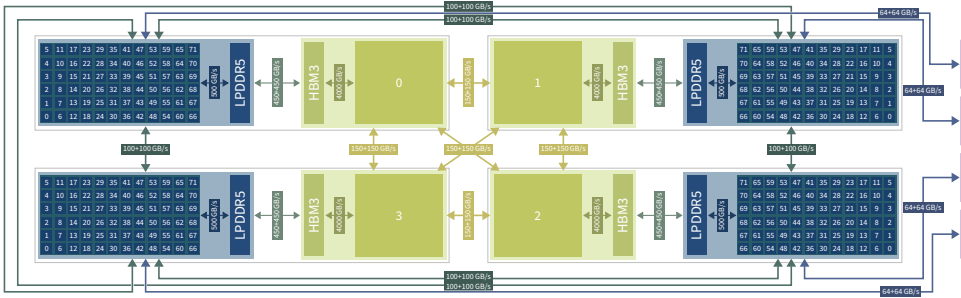
JUPITER System Overview

- ParTec/Eviden consortium
- Implementing Modular Supercomputing Architecture
- JUPITER Booster: High scalability, 1 EFLOP/s HPL, > 40 EFLOP/s FP8
  5884 nodes: 4× Grace-Hopper superchip, 4× network
- JUPITER Cluster: High versatility, high memory-to-compute ratio
  582 nodes: 2× AMD Turin 9655 (2 × 96 cores; DDR5; 1× network)
  162 nodes: 2× SiPearl Rhea1 (2 × 80 cores; HBM; 1× network)
- Network: 200/400 Gbit/s NVIDIA InfiniBand NDR (Dragonfly+)
- Storage: 29 PB flash, 310 PB HDD, 370 PB tape
- Energy: 17 MW limit (HPL); direct liquid-cooled, energy re-use
- Modular data center of containers



ParTec MODULAR SUPERCOMPUTING EVIDEN NVIDIA SIPEARL JÜLICH Forschungszentrum

JUPITER Booster Node Design

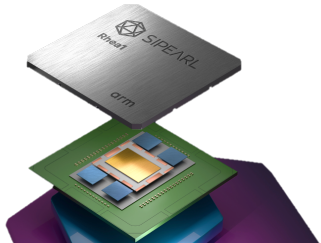


- 4× GH200 superchip (*CG4*)
  - Hopper: 132 SMs
  - Grace: 72 Arm Neoverse-V2 cores
  - 900 GB/s NVLink C2C
  - 120 GB LPDRR5X memory, 500 GB/s (CPU)
  - 96 GB HBM3 memory, 4000 GB/s (GPU)
- TDP superchip: 680 W
- GPU-to-GPU: NVLink 4, 300 GB/s per pair
- CPU-to-CPU: cNVLink, 200 GB/s per pair
- Cache coherence between all GPUs and CPUs
- 4× NDR200 InfiniBand (200 Gbit/s)

JUPITER Cluster Node Design

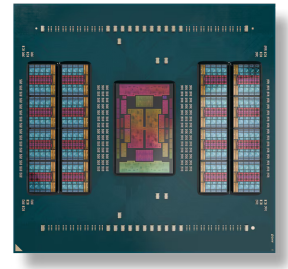
SiPearl Partition

- 162 nodes with 2×SiPearl Rhea1 CPU, each
  - 80 Neoverse-V1 cores (2× 256 bit SVE)
  - 256 GB DDR5 memory
  - 64 GB HBM memory
- Origins in European HPC ecosystem: EPI, EUPEX, and further projects
- Currently tested in Bull facility



AMD Partition

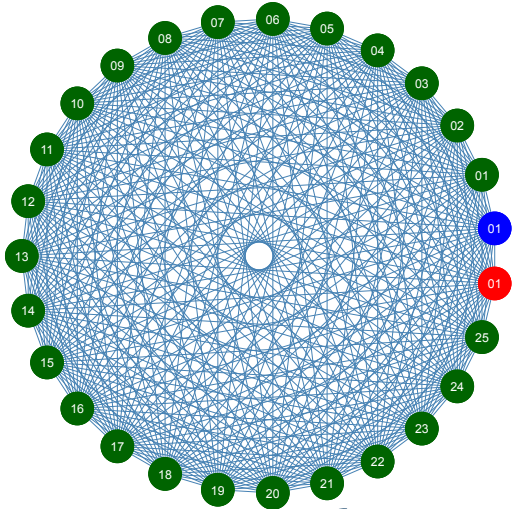
- 582 nodes with 2×AMD Turin 9655 CPU, each
  - 96 Zen5 cores (512 bit AVX-512; SMT-2)
  - 546 nodes: 768 GB DDR5 memory
  - 36 nodes: 1536 GB DDR5 memory
- Operational *very soon*



Rhea1 image by SiPearl, Zen5 image by AMD

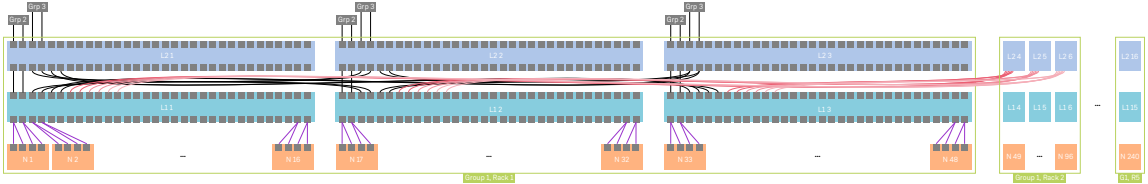
JUPITER Network

- NVIDIA Mellanox InfiniBand NDR
- Quantum-2 switches, ConnectX-7 HCAs
- Nodes connected to switches with split cables
- Dragonfly+ topology
  - 27 Dragonfly+ groups (25 for Booster, 1 for Cluster, 1 for management/storage)
  - Inside each group: full fat tree (two levels)
- Adaptive routing, SHARPv3
- 51 000 links, 25 400 endpoints, 867 switches
- 140 TB/s bisection bandwidth



JÜLICH Forschungszentrum

JUPITER Network



- In group *intra*
  - 5 racks
  - 240 nodes
  - 15 L1 (lower), 16 L2 (upper) switches
  - Per rack: 48 nodes, 192 GH200
  - Link bandwidth from HCA to L1: 200 Gbit/s; from L1 to L2: 400 Gbit/s
- Beyond group *inter*
  - 16 links to each other group
  - Link bandwidth 400 Gbit/s

JÜLICH Forschungszentrum

Site Preparations

- New 110 kV power lines drawn to campus transformer substation
- Upgrade of substations:
  2×40 MVA →2×80 MVA
- New lines to exascale site:
  2×35 kV







JÜLICH Forschungszentrum



MDC Concrete Foundation; Andreas for scale





Photos by Herwig Zilken / FZJ


Delivery of
IT Module 2
COLONIA
Photo by Herwig Zilken (GZ)



Installation of Compute Blades in IT Module 1

Photo by Herwig Zilken / FZJ


MDC bird's eye view
hotography Schmidt Kleklau / FZJ


Inside of IT module
Photo: Sascha Kreklau / FZJ



Inauguration Tent

Photo by Jenö Gellinek / FZJ



Thomas Lippert and the press

Photo by Jenö Gellinek / FZJ



Friedrich Merz in speech

Photo by Kurt Steinhausen / FZJ

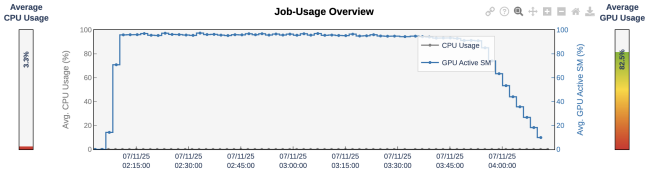


Buttons! So many Buttons!

Photo by F

Exascale Liftoff!

- TOP500 Nov 2025: #4 world, #1 EU
- 1000.184 PFLOP/s HPL
  1 JUGENE system (2008) over 1 EFLOP/s



LLview job report of HPL job

JOINING FORCES

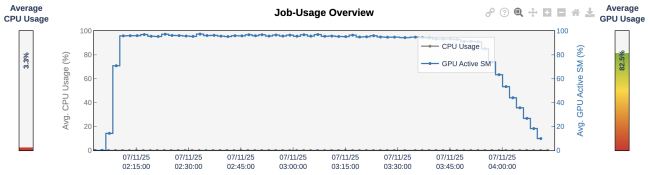
jupiter.fz-juelich.de



Certificate from first listing in June 2025

Exascale Liftoff!

- TOP500 Nov 2025: #4 world, #1 EU
- 1000.184 PFLOP/s HPL
  1 JUGENE system (2008) over 1 EFLOP/s
- Green500 Nov. 2025: JUPITER #14
  (63.3 GFLOP/(s W) no dedicated run)
  Jun. 2025: JEDI #1 (72.7 GFLOP/(s W), many tricks)



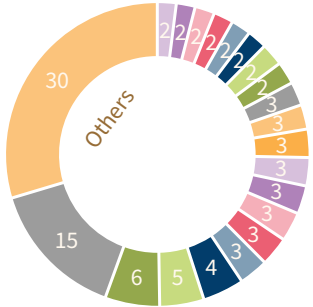
LLview job report of HPL job



Certificate from first listing in June 2025

Planning JUPITER: Workload

- JSC: HPC resources for Forschungszentrum campus, state (NRW), Germany, Europe
- Compute time through peer-review
- Heterogeneous workload
- Physics, climate, biology, chemistry, AI, …
- **Goal:** Respect current & anticipated workload in procurements of new systems; incl. domains, methods, programming languages, profiles

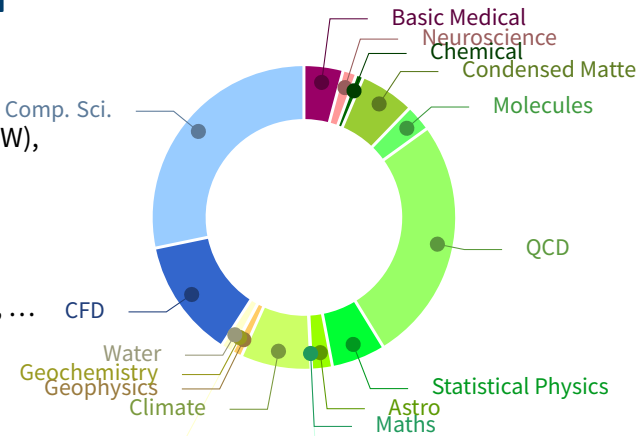


Programs / GPUh (2020)
JUWELS Booster

JÜLICH Forschungszentrum

Planning JUPITER: Workload

- JSC: HPC resources for Forschungszentrum campus, state (NRW), Germany, Europe
- Compute time through peer-review
- Heterogeneous workload
- Physics, climate, biology, chemistry, AI, …
- **Goal:** Respect current & anticipated workload in procurements of new systems; incl. domains, methods, programming languages, profiles

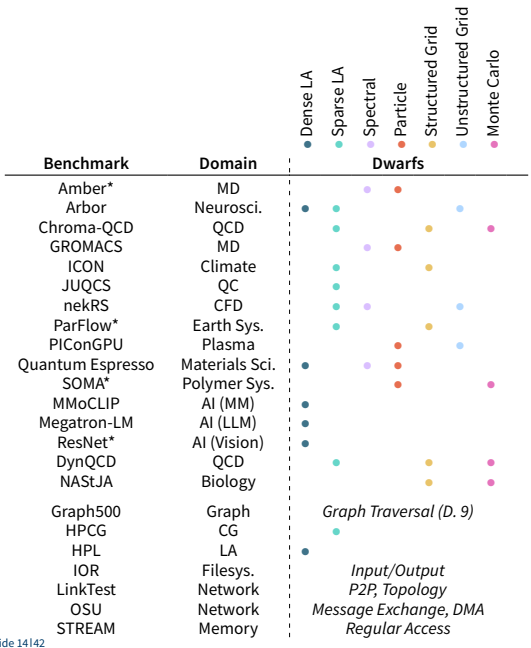


Compute Time per Domain (2024-1)
JUWELS Booster

JÜLICH Forschungszentrum

Benchmarks Overview

- Mixture of TCO application, synthetic, high-scaling (1000 PFLOP/sec (th.) scale) benchmarks
- 16 application benchmarks (4 de-selected for actual procurement; 5 for high-scaling) Cross-section of domains and methods, 3× AI
- 7 synthetic benchmarks
- Extensive descriptions
- Right: Patterns of 7 Dwarfs

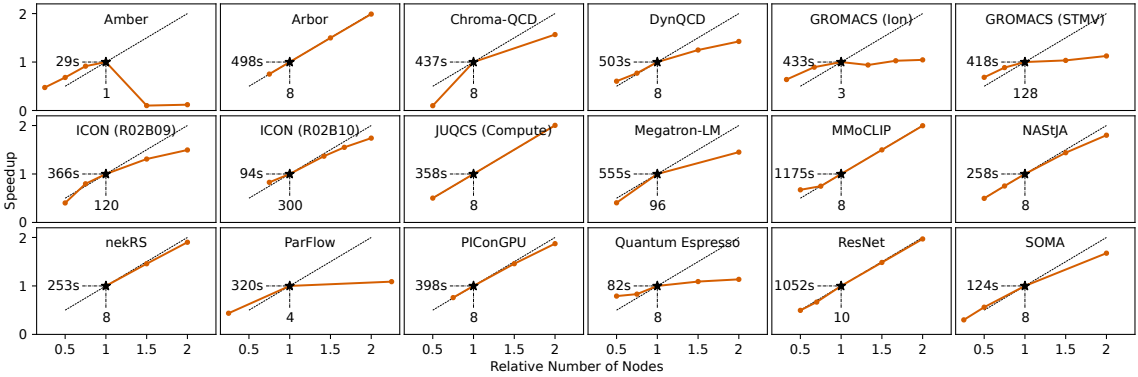


Grand Table Benchmark Details

- Languages, models, libraries
- Licenses
- References nodes Base, High-Scaling
- Memory variants
- Execution targets

	Application Features			Execution Targets		
	Benchmark Name	Progr. Language, [Libraries,]Prog. Models	Licence	Nodes Base	Nodes High-Scale $N^{\text{Mem Vars}}$	Module/ Device B B A C C M GC
Application	Amber*	Fortran, CUDA	*Custom*	1		✓
	Arbor	C++, CUDA/HIP	BSD-3-Clause	8	$642^{T,S,M,L}$	✓
	Chroma-QCD	C++, QUDA, CUDA/HIP	JLab	8	$512^{S,M,L}$	✓
	GROMACS	C++, CUDA/SYCL	LGPLv2.1	3/128		✓
	ICON	Fortran/C, OpenACC/CUDA/HIP	BSD-3-Clause	120/300		✓
	JUQCS	Fortran, CUDA/OpenMP	*None*	8	$512^{S,L}$	✓ ✓
	nekRS	C++/C, OCCA, CUDA/HIP/SYCL	BSD-3-Clause	8	$642^{S,M,L}$	✓
	ParFlow*	C, Hypre, CUDA/HIP	LGPL	4		✓
	PIConGPU	C++, Alpaka, CUDA/HIP	GPLv3+	8	$640^{S,M,L}$	✓
	Quantum Espresso	Fortran, ELPA, OpenACC/CUF	GPL	8		✓
	SOMA*	C, OpenACC	LGPL	8		✓
	MMoCLIP	Python, PyTorch, CUDA/ROCm	MIT	8		✓
	Megatron-LM	Python, PyTorch/Apex, CUDA/ROCm	BSD-3-Clause	96		✓
	ResNet*	Python, TensorFlow, CUDA/ROCm	Apache-2.0	10		✓
	DynQCD	C, OpenMP	*None*	8		✓
	NAStJA	C++, MPI	MPL-2.0	8		✓
Synthetic	Graph500	C, MPI	MIT	4/16/all		✓
	HPCG	C++, OpenMP, CUDA/HIP	BSD-3-Clause	1/4/all		✓ ✓
	HPL	C, BLAS, OpenMP, CUDA/HIP	BSD-4-Clause	1/16/all		✓ ✓
	IOR	C, MPI	GPLv2	-/> 64		✓ ✓
	LinkTest	C++, MPI/SIONlib	BSD-4-Clause+	all		✓ ✓ ✓
	OSU	C, MPI, CUDA	BSD	1/2		✓ ✓ ✓
	STREAM	C, CUDA/ROCm/OpenACC	*Custom*	1		✓ ✓

TCO Base Result Grid

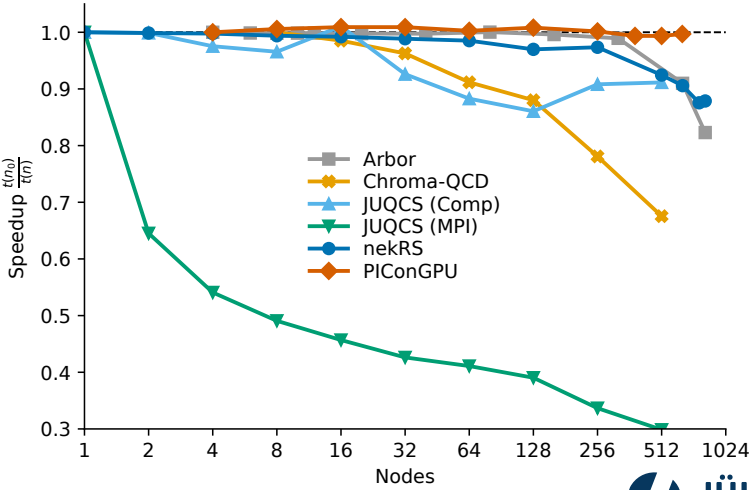


Execution on reference number of nodes ($x = 1$) resulting in reference timing ($y = 1$); absolute numbers super-imposed; strong scaling to $0.5\times$ - $2\times$

JÜLICH Forschungszentrum

High-Scaling Results

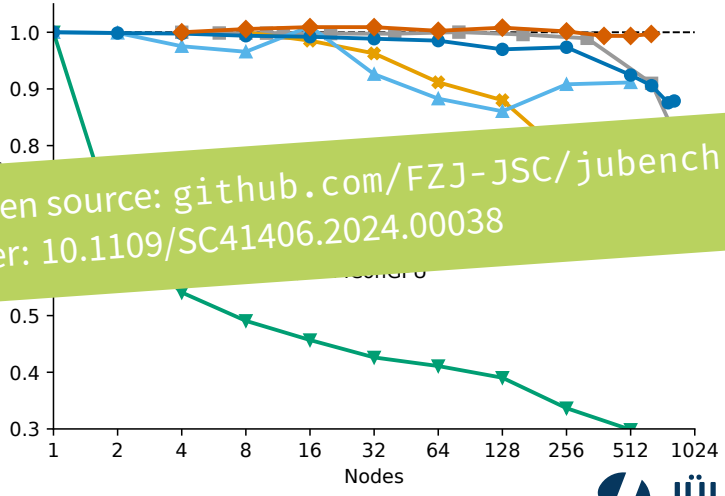
Weak-scaling relative to reference up to 642 nodes (50 PFLOP/s th.)



JÜLICH Forschungszentrum

High-Scaling Results

Weak-scaling
rel
ref
642
(50



All benchmarks open source: github.com/FZJ-JSC/jubench
Paper: 10.1109/SC41406.2024.00038



exacb.pages.jsc.fz-juelich.de

Features

- Continuous Benchmarking for HPC
- Template-based declarative syntax
- Reduced barrier to entry for CI/CB
- Reproducible benchmarks (JUBE, …)
- Ease of sharing workflows
- Pre-defined runs, experiments

→ Currently extending, publishing

JUREAP default example:

```yaml
include:
 - component: jureap/jube@v3.2
   inputs:
    prefix: "jedi.strong.tiny"
    variant: "strong.tiny"
    machine: "jedi"
    queue: "all"
    project: "cjsc"
    budget: "zam"
    jube_file: "simple.yaml"
```



exacb.pages.jsc.fz-juelich.de

Features

- Continuous Benchmarking for HPC
- Template-based declarative syntax
- Reduced barrier to entry for CI/CB
- Reproducible benchmarks (JUBE, …)
- Ease of sharing workflows
- Pre-defined runs, experiments

→ Currently extending, publishing

Energy measurement example:

```yaml
include:
- component: jureap/energy@v3.2
  inputs:
    prefix: "jedi.strong.tiny"
    variant: "strong.tiny"
    machine: "jedi"
    queue: "all"
    project: "cjsc"
    budget: "zam"
    jube_file: "simple.yaml"
```


JÜLICH Forschungszentrum

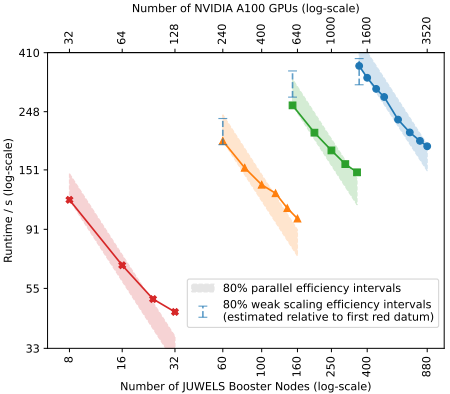
exaCB

exacb.pages.jsc.fz-juelich.de

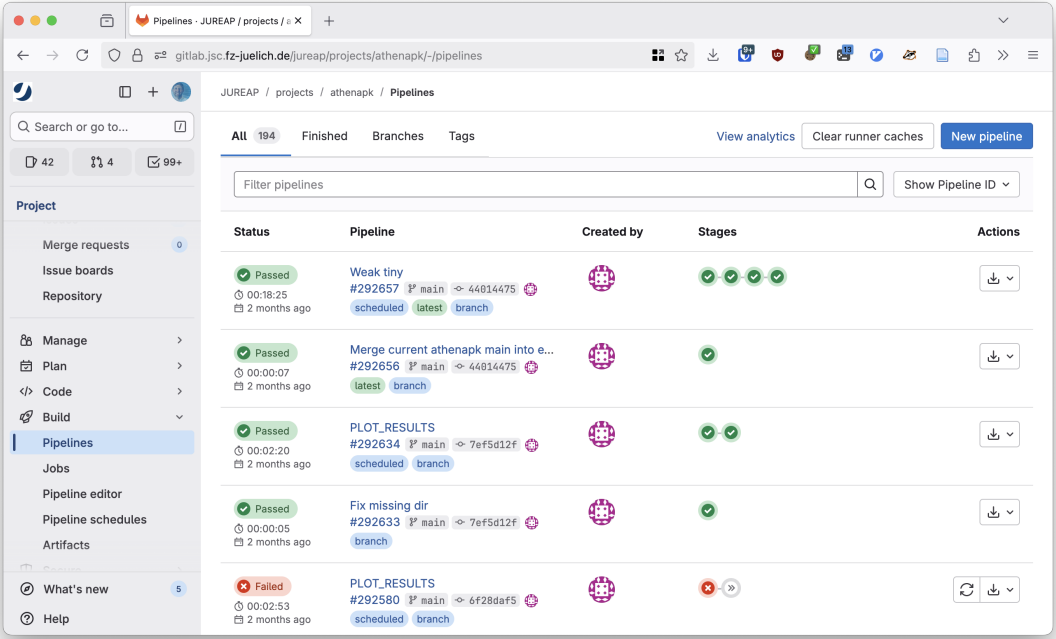
Features

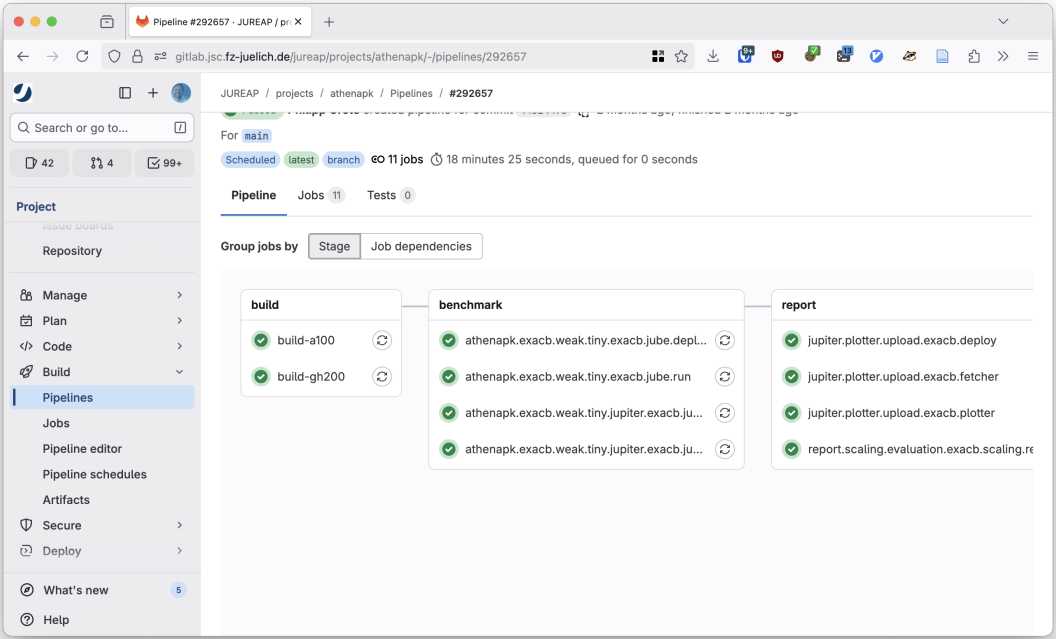
- Continuous Benchmarking for HPC
- Template-based declarative syntax
- Reduced barrier to entry for CI/CB
- Reproducible benchmarks (JUBE, …)
- Ease of sharing workflows
- Pre-defined runs, experiments

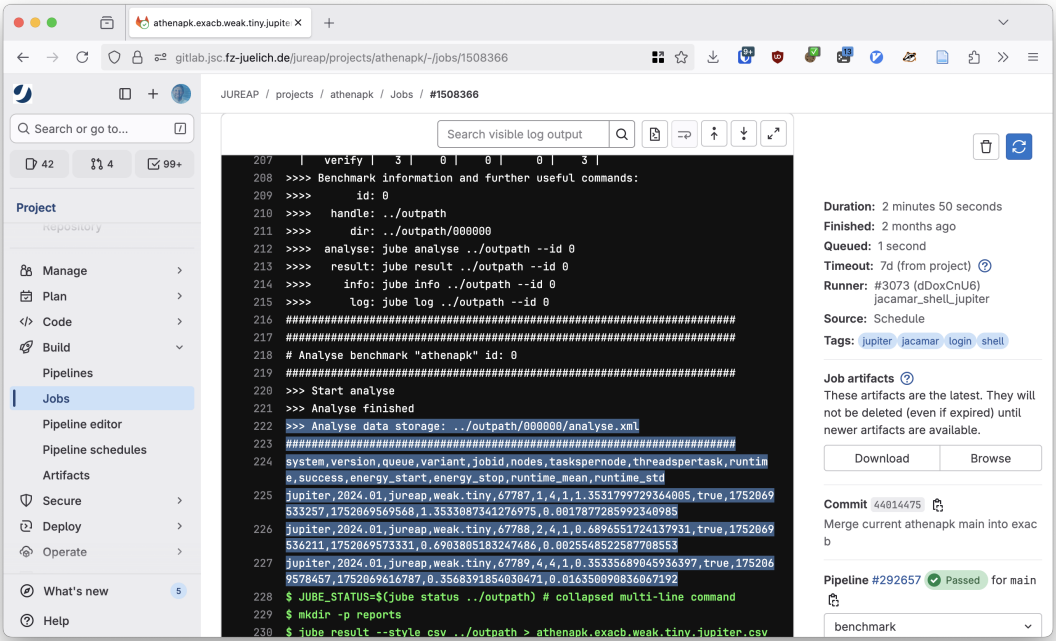
→ Currently extending, publishing



JÜLICH Forschungszentrum


Pipelines · JUREAP / projects / athenapk
gitlab.jsc.fz-juelich.de/jureap/projects/athenapk/-/pipelines
Search or go to...
42
4
99+
Project
Merge requests 0
Issue boards
Repository
Manage
Plan
Code
Build
Pipelines
Jobs
Pipeline editor
Pipeline schedules
Artifacts
What's new 5
Help
JUREAP / projects / athenapk / Pipelines
All 194
Finished
Branches
Tags
View analytics
Clear runner caches
New pipeline
Filter pipelines
Show Pipeline ID
Status
Pipeline
Created by
Stages
Actions
Passed
00:18:25
2 months ago
Weak tiny
#292657 main 44014475
scheduled latest branch
Passed
00:00:07
2 months ago
Merge current athenapk main into e...
#292656 main 44014475
latest branch
Passed
00:02:20
2 months ago
PLOT_RESULTS
#292634 main 7ef5d12f
scheduled branch
Passed
00:00:05
2 months ago
Fix missing dir
#292633 main 7ef5d12f
branch
Failed
00:02:53
2 months ago
PLOT_RESULTS
#292580 main 6f28daf5
scheduled branch

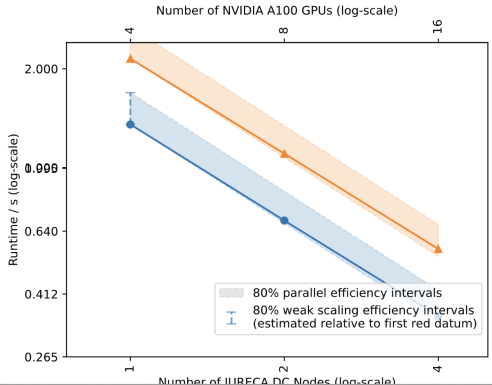

Pipeline #292657 · JUREAP / pr
gitlab.jsc.fz-juelich.de/jureap/projects/athenapk/-/pipelines/292657
Search or go to...
42
4
99+
Project
Repository
Manage
Plan
Code
Build
Pipelines
Jobs
Pipeline editor
Pipeline schedules
Artifacts
Secure
Deploy
What's new
5
Help
JUREAP / projects / athenapk / Pipelines / #292657
For main
Scheduled
latest
branch
11 jobs
18 minutes 25 seconds, queued for 0 seconds
Pipeline
Jobs 11
Tests 0
Group jobs by
Stage
Job dependencies
build
build-a100
build-gh200
benchmark
athenapk.exacb.weak.tiny.exacb.jube.depl...
athenapk.exacb.weak.tiny.exacb.jube.run
athenapk.exacb.weak.tiny.jupiter.exacb.ju...
athenapk.exacb.weak.tiny.jupiter.exacb.ju...
report
jupiter.plotter.upload.exacb.deploy
jupiter.plotter.upload.exacb.fetcher
jupiter.plotter.upload.exacb.plotter
report.scaling.evaluation.exacb.scaling.re


athenapk.exacb.weak.tiny.jupite
gitlab.jsc.fz-juelich.de/jureap/projects/athenapk/-/jobs/1508366
Search or go to...
42
4
99+
Project
Manage
Plan
Code
Build
Pipelines
Jobs
Pipeline editor
Pipeline schedules
Artifacts
Secure
Deploy
Operate
What's new
Help
JUREAP / projects / athenapk / Jobs / #1508366
Search visible log output
| verify | 3 | 0 | 0 | 0 | 3 |
>>>> Benchmark information and further useful commands:
>>>> id: 0
>>>> handle: ../outpath
>>>> dir: ../outpath/000000
>>>> analyse: jube analyse ../outpath --id 0
>>>> result: jube result ../outpath --id 0
>>>> info: jube info ../outpath --id 0
>>>> log: jube log ../outpath --id 0
>>> Start analyse
>>> Analyse finished
>>> Analyse data storage: ../outpath/000000/analyse.xml
system,version,queue,variant,jobid,nodes,taskspernode,threadspertask,runtime,success,energy_start,energy_stop,runtime_mean,runtime_std
jupiter,2024.01,jureap,weak.tiny,67787,1,4,1,1,1.3531799729364005,true,1752069533257,1752069569568,1.3533087341276975,0.0017877285992340985
jupiter,2024.01,jureap,weak.tiny,67788,2,4,1,0,0.6896551724137931,true,1752069536211,1752069573331,0.6903805183247486,0.0025548525287708553
jupiter,2024.01,jureap,weak.tiny,67789,4,4,1,0,0.35335689045936397,true,1752069578457,1752069616787,0.3568391854030471,0.016350090836067192
$ JUBE_STATUS=$(jube status ../outpath) # collapsed multi-line command
$ mkdir -p reports
$ jube result --style csv ../outpath > athenapk.exacb.weak.tiny.jupiter.csv
Duration: 2 minutes 50 seconds
Finished: 2 months ago
Queued: 1 second
Timeout: 7d (from project)
Runner: #3073 (dDoxCnU6) jacamar_shell_jupiter
Source: Schedule
Tags: jupiter jacamar login shell
Job artifacts
These artifacts are the latest. They will not be deleted (even if expired) until newer artifacts are available.
Download
Browse
Commit 44014475
Merge current athenapk main into exacb
Pipeline #292657 Passed for main
benchmark

Search or go to...
42 4 99+
Project
Repository
Manage
Plan
Code
Merge requests 0
Repository
Branches
Commits
Tags
Repository graph
Compare revisions
Snippets
What's new 5
Help

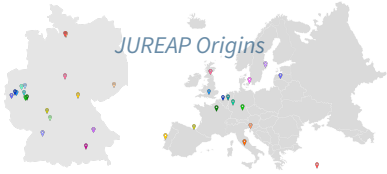
July 9, 2025

1 Evaluation Plot



Application Results

Results Overview



- JUPITER currently in **pre-access**
- **JUREAP**: Competitive early access program since 2024
  - Open phase: 130 proposals
  - **Reproducibility, continuous integration** (***exaCB***)
  - Lighthouse application: peer-reviewed selection, 15🇪🇺+18🇩🇪 apps
- National: 150 regular and large-scale projects and *AI Competition*
- EU: 50 *normal* projects, 50 *test* projects
- Next, some highlights from JUREAP and tests

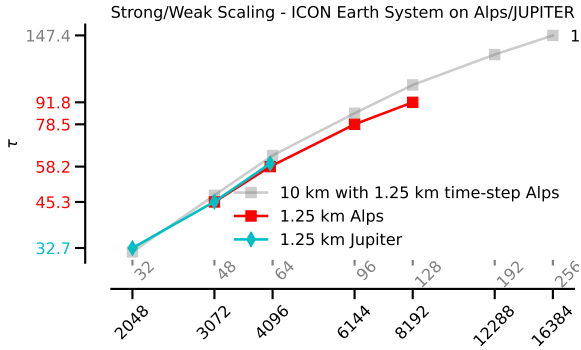
JÜLICH Forschungszentrum

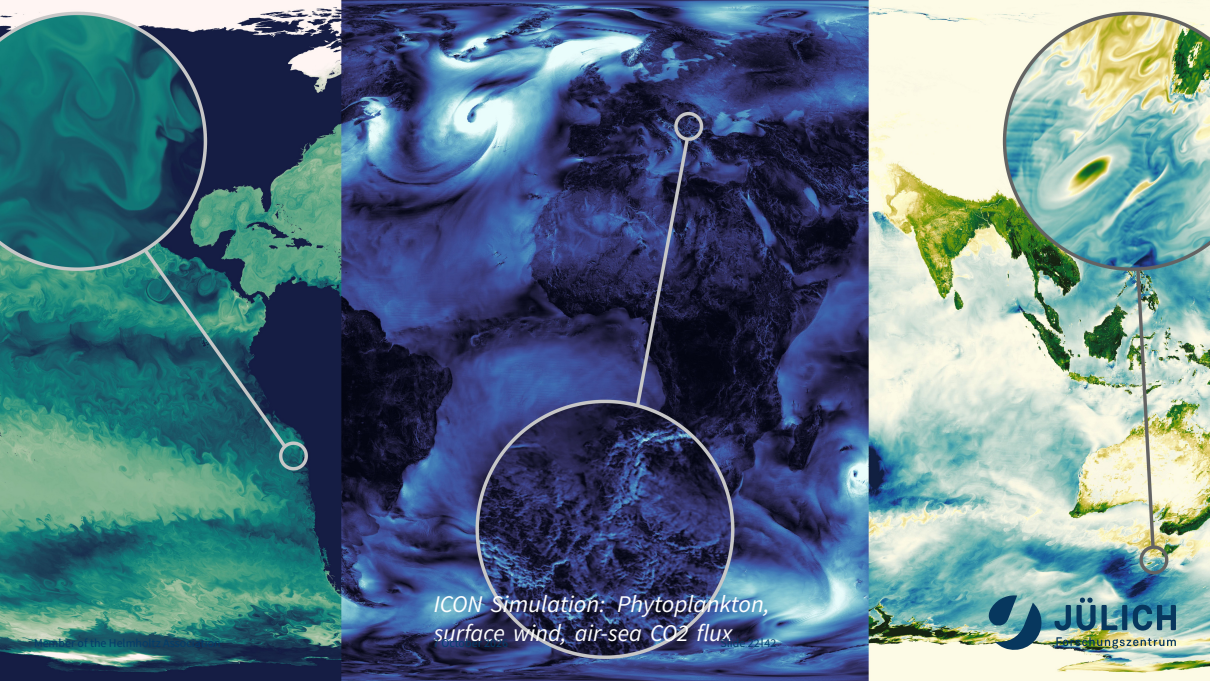
ICON: Full Earth System at 1 km

MAX-PLANCK-INSTITUT FÜR METEOROLOGIE

50 YEARS OF CLIMATE RESEARCH

- **ICON**: European weather/climate model by Max Planck Institute for Meteorology, DKRZ, and others
- Components for simulation of atmosphere, ocean, land, …
- Mostly Fortran code, MPI/OpenMP (CPU) and OpenACC (GPU); but also DaCe
- Simulation of full Earth System at 1.25 km resolution
  - 20 480 GH200 of JUPITER
  - Full superchip: CPU (ocean, land) and GPU (atmosphere)
  - 145 days simulated days per real day



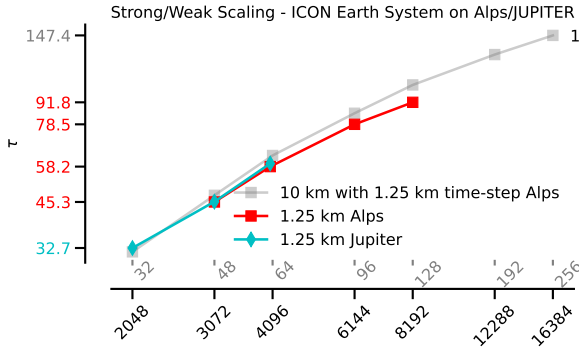

ICON Simulation: Phytoplankton,
surface wind, air-sea CO2 flux
JÜLICH
Forschungszentrum

ICON: Full Earth System at 1 km

MAX-PLANCK-INSTITUT FÜR METEOROLOGIE

50 YEARS OF CLIMATE RESEARCH

- **ICON**: European weather/climate model by Max Planck Institute for Meteorology, DKRZ, and others
- Components for simulation of atmosphere, ocean, land, …
- Mostly Fortran code, MPI/OpenMP (CPU) and OpenACC (GPU); but also DaCe
- Simulation of full Earth System at 1.25 km resolution
  - 20 480 GH200 of JUPITER
  - Full superchip: CPU (ocean, land) and GPU (atmosphere)
  - 145 days simulated days per real day



→ DOI 10.1145/3712285.3771789

JÜLICH Forschungszentrum

ICON: Full Earth System at 1 km

MAX-PLANCK-INSTITUT FÜR METEOROLOGIE

50 YEARS OF CLIMATE RESEARCH

- **ICON**: European weather/climate model by Max Planck Institute for Meteorology, DKRZ, and others
- Components for simulation of atmosphere, ocean, land, …
- Mostly Fortran code, MPI/OpenMP (CPU) and OpenACC (GPU); but also DaCe
- Simulation of full Earth System at 1.25 km resolution
  - 20 480 GH200 of JUPITER
  - Full superchip: CPU (ocean, land) and GPU (atmosphere)
  - 145 days simulated days per real day

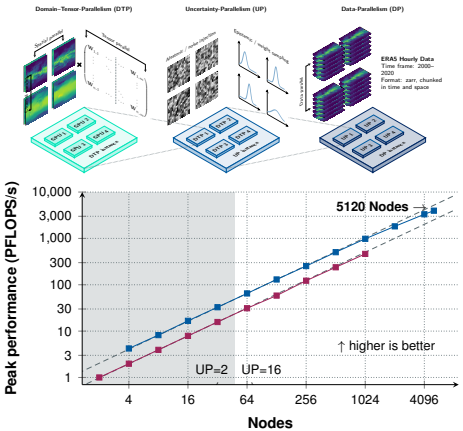
→ DOI 10.1145/3712285.3771789



Gordon Bell Prize for Climate Modeling 2025

BEAST: 4D-Parallel Bayesian Weather Model

- Bayesian Swin Transformer for atmospheric modeling
- By Karlsruhe Institute of Technology (C. Debus)
- 4D parallelism: Domain-Tensor, Uncertainty, Data → uncertainty part of the model
- PyTorch with DDP
- 2.4B model, 4 EFLOP/s performance on JUPITER
- Smaller model: predictive skill like current reference model, also predict extreme events
- Nominated for Gordon Bell Climate 🤞



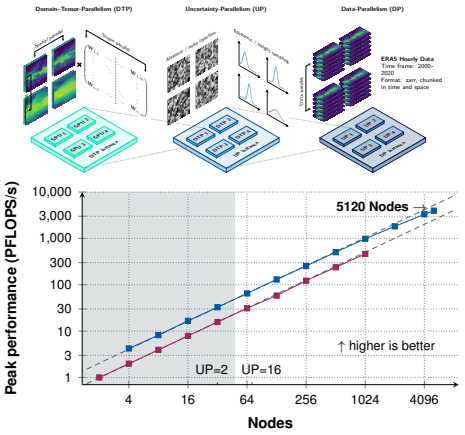
red: LUMI (smaller model), blue: JUPITER

JÜLICH Forschungszentrum

BEAST: 4D-Parallel Bayesian Weather Model

- Bayesian Swin Transformer for atmospheric modeling
- By Karlsruhe Institute of Technology (C. Debus)
- 4D parallelism: Domain-Tensor, Uncertainty, Data → uncertainty part of the model
- PyTorch with DDP
- 2.4B model, 4 EFLOP/s performance on JUPITER
- Smaller model: predictive skill like current reference model, also predict extreme events
- Nominated for Gordon Bell Climate 🤞

→ DOI 10.48550/arXiv.2609.12815



red: LUMI (smaller model), blue: JUPITER

JÜLICH Forschungszentrum

JUQCS: 50 Qubits

- Universal Quantum Computer Simulator
- 2^{N+4} memory with constant exchange → HPC network
- Fortran, MPI/OpenMP (CPU) and MPI/CUDA Fortran (GPU)
- **JUPITER: Record-breaking 50 Qubit simulation**
  - 16 384 GH200 (2^{14}); GPU for simulation, but CPU+GPU memory together
  - Advanced data encoding: 16 PB memory ⇒ 2 PB
  - Memory: use host-side memory, evaluate different approaches
  - 11× speed-up over previous 48 Qubit record (K computer)

Strategy	Runtime (s)	Speedup (*rel*)
1 `malloc()`	446.67	1
2 `numactl`	160.00	2.8
3 `numa_alloc_onnode()`	101.57	4.4
4 Data Usage Hints	71.70	6.2
5 Explicit Data Copies	53.87	8.3

JUQCS: 50 Qubits

- Universal Quantum Computer Simulator
- 2^{N+4} memory with constant exchange → HPC network
- Fortran, MPI/OpenMP (CPU) and MPI/CUDA Fortran (GPU)
- **JUPITER: Record-breaking 50 Qubit simulation**
  - 16 384 GH200 (2^{14}); GPU for simulation, but CPU+GPU memory together
  - Advanced data encoding: 16 PB memory ⇒ 2 PB
  - Memory: use host-side memory, evaluate different approaches
  - 11× speed-up over previous 48 Qubit record (K computer)

→ DOI 10.1016/j.future.2026.108592

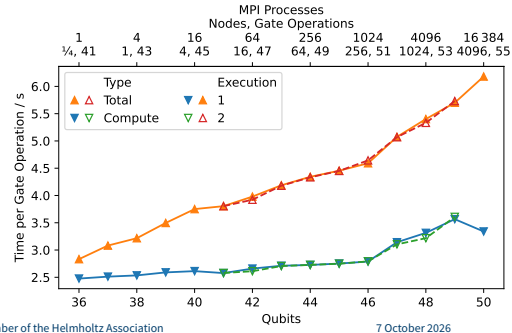
Strategy	Runtime (s)	Speedup (*rel*)
1 `malloc()`	446.67	1
2 `numactl`	160.00	2.8
3 `numa_alloc_onnode()`	101.57	4.4
4 Data Usage Hints	71.70	6.2
5 Explicit Data Copies	53.87	8.3

JÜLICH Forschungszentrum

JUQCS: Scaling

Weak Scaling

- Total and compute; time per gate operation
- Two datasets: **1** – empty JUPITER / exclusive JUQCS; **2** – full JUPITER / shared network
- Behavior $N > 46$: interruptions in global MPI communication for encoding, GPU compute variability, network congestion

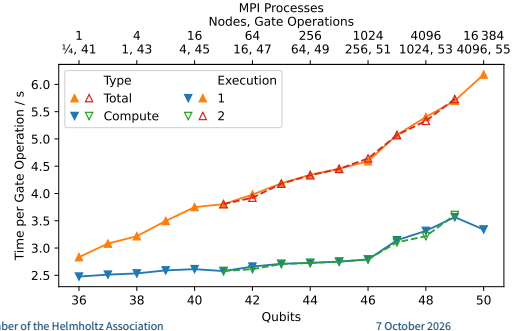


JÜLICH Forschungszentrum

JUQCS: Scaling

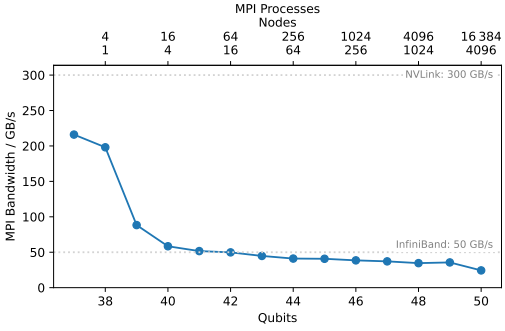
Weak Scaling

- Total and compute; time per gate operation
- Two datasets: **1** – empty JUPITER / exclusive JUQCS; **2** – full JUPITER / shared network
- Behavior $N > 46$: interruptions in global MPI communication for encoding, GPU compute variability, network congestion



Communication Bandwidth

- Transition between node-local and inter-node
- Inter-node fit: bandwidth $\mathcal{O}(1/\sqrt{N})$
- $N = 50$: 38 766 TiB through network in 100 s

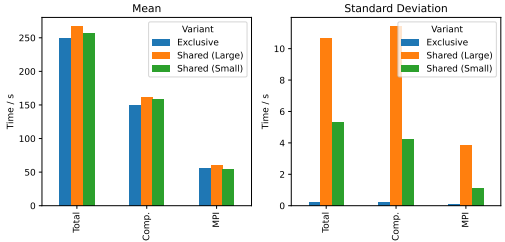


JÜLICH Forschungszentrum

JUQCS: Some Testing

Self-Pollution

- Run instances of JUQCS on free JUPITER
- 47 qubits, 512 nodes
  - **Excl:** Empty system, one job after another,
  - **Small:** Packs of 5 jobs executed in parallel
  - **Large:** Packs of 10 jobs executed in parallel
- Mean time changes 7 %; σ changes 50$\times$!

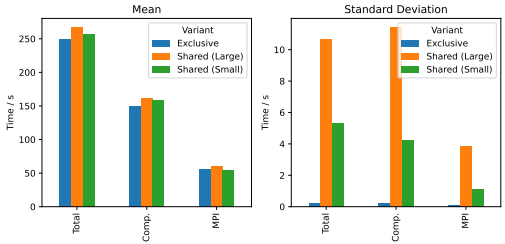


JÜLICH Forschungszentrum

JUQCS: Some Testing

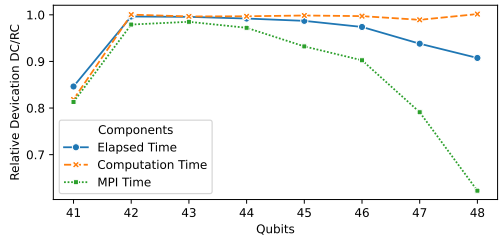
Self-Pollution

- Run instances of JUQCS on free JUPITER
- 47 qubits, 512 nodes
  - **Excl:** Empty system, one job after another,
  - **Small:** Packs of 5 jobs executed in parallel
  - **Large:** Packs of 10 jobs executed in parallel
- Mean time changes 7 %; σ changes 50$\times$!



Transport Layers

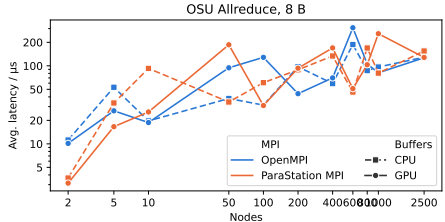
- Testing RC (*Reliable*) and DC (*Dynamically*)
- Usually application- / message-size dependent
- Here: DC better, especially at scale

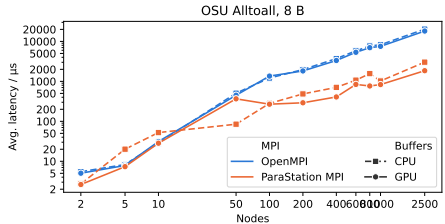


JÜLICH Forschungszentrum

OSU at Scale: Latencies

- For EuroMPI/IWOMP: Test (untuned) MPI performance at scale
- OSU Micro-Benchmarks 7.5.2, OpenMPI 5.0.5 / ParaStationMPI/5.13.0-1, with / without CUDA
- `osu_allreduce` and `osu_alltoall`
- Executed yesterday, up to 2500 nodes
- Allreduce jumps around; OpenMPI worse for large alltoall (more investigated needed); mostly only little difference between CPU and GPU



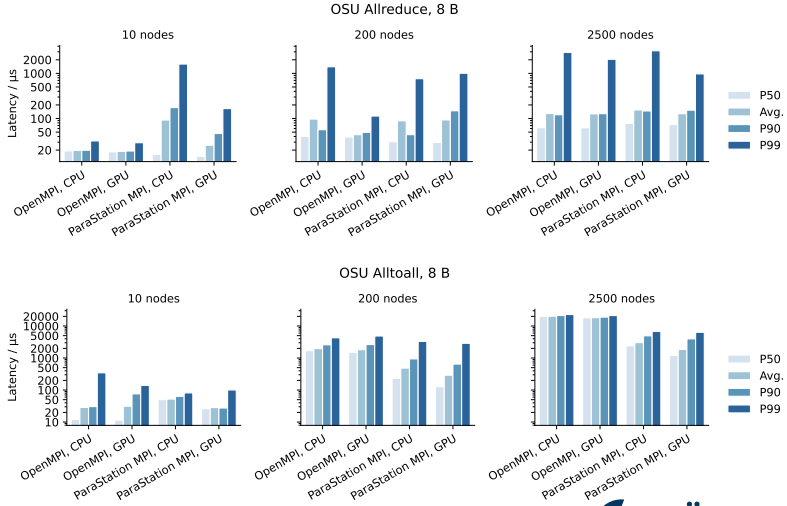


AI-supported design and evaluation

JÜLICH Forschungszentrum

OSU at Scale: Deviations

- Outliers have serious impact, especially at scale
- Different MPIs behave differently
- Tuning can certainly mitigate this
- *Not run in isolation*



Network Stability Deep Dive

Introduction

- Investigation from July
- Aim: understand outliers, scales, Dragonfly groups (DFGs)
- OMB allreduce → latency; RC transport
- Only OpenMPI for now
- Targeting DFGs with `--constraint`
- Also: patched OMB to report rank-precise values (and no accumulation)
- *Internal report available on request*

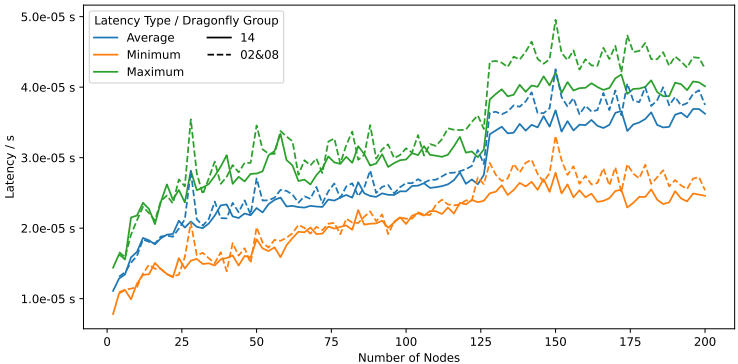
AI-supported plotting

JÜLICH Forschungszentrum

Network Stability Deep Dive

Baseline

- Jump >125 nodes
- Stronger for multi-DFG
- Plateau

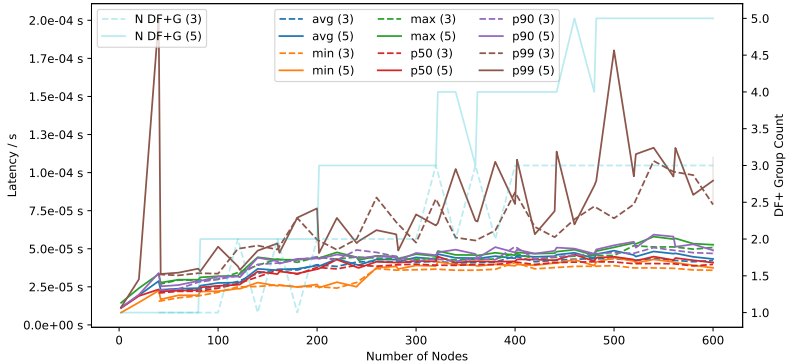


JÜLICH Forschungszentrum

Network Stability Deep Dive

Increase Scale

- Driven by P99 outliers
- Light-blue: number of DFGs job step is executed on

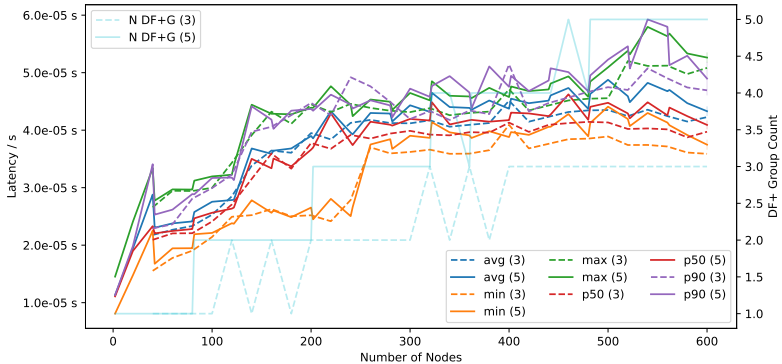


JÜLICH Forschungszentrum

Network Stability Deep Dive

Increase Scale

- Driven by P99 outliers
- Light-blue: number of DFGs job step is executed on
- -P99: Pattern from before continues

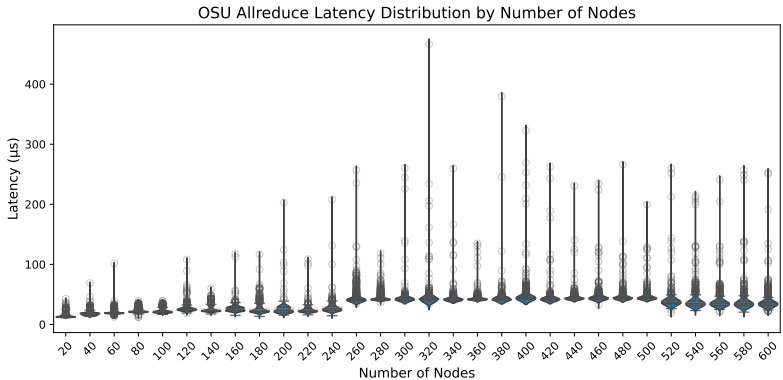


JÜLICH Forschungszentrum

Network Stability Deep Dive

Latency Violin Plot

- Violin/box plot
- So many outliers!

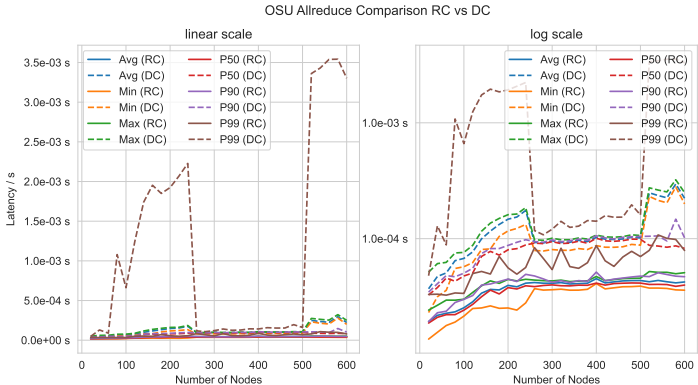


JÜLICH Forschungszentrum

Network Stability Deep Dive

RC vs. DC

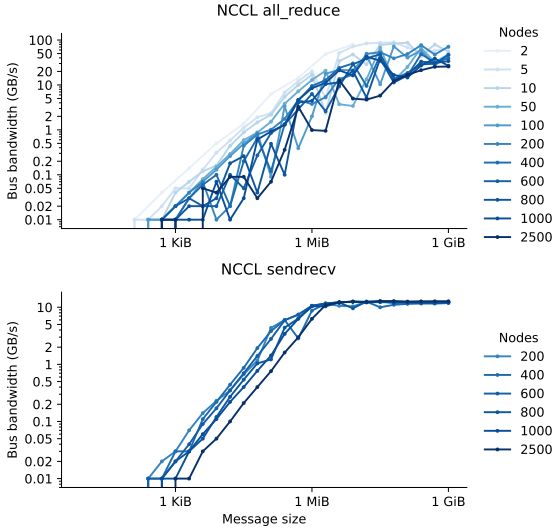
- DC no good for OMB
- Strange bathtub shape



JÜLICH Forschungszentrum

NCCL Tests

- NCCL 2.29.7
- Little difference between MPIs (shown: OpenMPI)
- Log-log plot
- 50 MiB messages needed to saturate



JÜLICH Forschungszentrum

LinkTest: Dragonfly+ Group

- LinkTest
  - Ping pong between hosts/network adapters
  - Measure latency/bandwidth
  - Developed by JSC: `https://go.fzj.de/linktest`
- Parameters of test
  - 238 hosts / 952 network adapters →905 352 pairs
  - 16 MiB message size, 8 messages; parallel execution
  - CPU-only, aligned `malloc`
- Result: nearly all at 22.89 GiB/s; few outliers (worst at ⅒ of bandwidth)



LinkTest: Dragonfly+ Group

- LinkTest
  - Ping pong between hosts/network adapters
  - Measure latency/bandwidth
  - Developed by JSC: `https://go.fzj.de/linktest`
- Parameters of test
  - 238 hosts / 952 network adapters →905 352 pairs
  - 16 MiB message size, 8 messages; parallel execution
  - CPU-only, aligned `malloc`
- Result: nearly all at 22.89 GiB/s; few outliers (worst at ⅒ of bandwidth)

→ Clip to 99.9 % of data

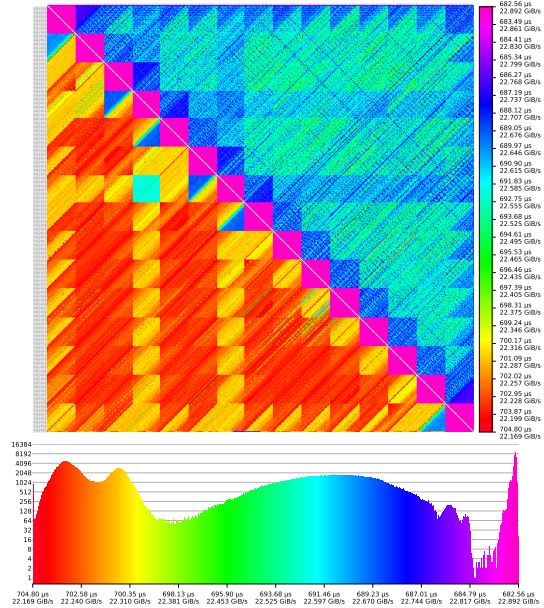


LinkTest: Dragonfly+ Group

- LinkTest
  - Ping pong between hosts/network adapters
  - Measure latency/bandwidth
  - Developed by JSC: `https://go.fzj.de/linktest`
- Parameters of test
  - 238 hosts / 952 network adapters →905 352 pairs
  - 16 MiB message size, 8 messages; parallel execution
  - CPU-only, aligned `malloc`
- Result: nearly all at 22.89 GiB/s; few outliers (worst at 1/10 of bandwidth)

→ Clip to 99.9 % of data

- Interesting patterns (L1/L2), but in small 3 % window!



LinkTest: Full JUPITER*

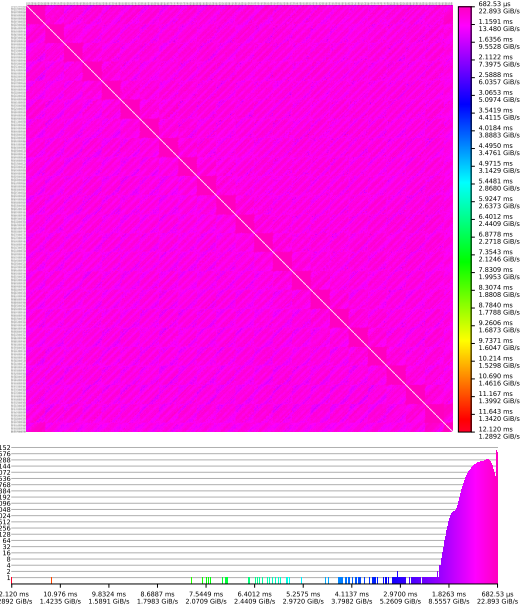
- *: 5284 nodes *rest in maintenance*
- Parameters
  - 21136 network adapters → 446 709 360 pairs
  - 16 MiB message size, 8 messages; parallel execution
  - CPU-only, aligned `malloc`
  - Adaptive routing enabled
- Results: Many at 20 GiB/s, but average at 15 GiB/s → outliers! (even 1/20 of bandwidth)



LinkTest: Full JUPITER*

- *: 5284 nodes *rest in maintenance*
- Parameters
  - 21136 network adapters → 446 709 360 pairs
  - 16 MiB message size, 8 messages; parallel execution
  - CPU-only, aligned `malloc`
  - Adaptive routing enabled
- Results: Many at 20 GiB/s, but average at 15 GiB/s → outliers! (even 1/20 of bandwidth)

→ Clip to 2 ms and better

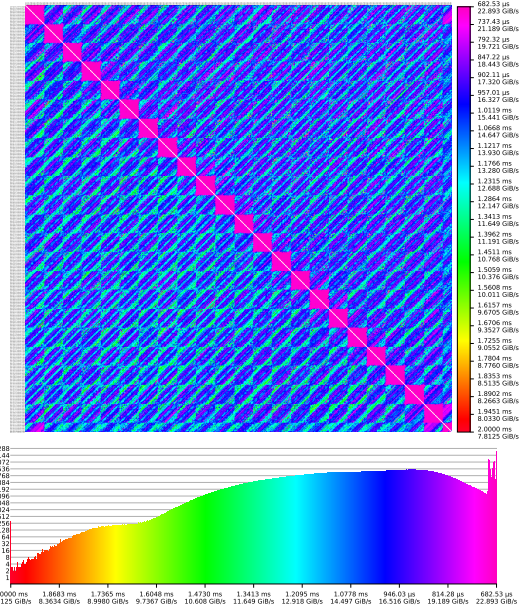


LinkTest: Full JUPITER*

- *: 5284 nodes *rest in maintenance*
- Parameters
  - 21136 network adapters → 446 709 360 pairs
  - 16 MiB message size, 8 messages; parallel execution
  - CPU-only, aligned `malloc`
  - Adaptive routing enabled
- Results: Many at 20 GiB/s, but average at 15 GiB/s → outliers! (even 1/20 of bandwidth)

→ Clip to 2 ms and better

- Interesting patterns: 22.5 groups, *anti*-diagonal elements

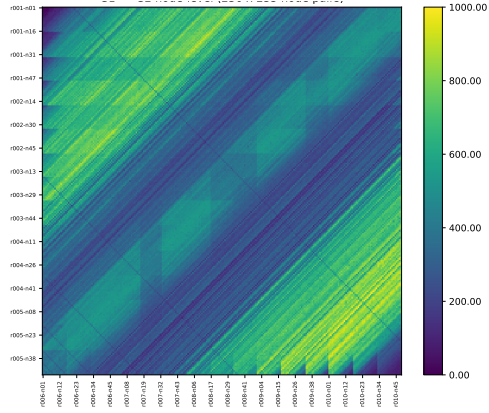


LinkTest: Full JUPITER*

- *: 5284 nodes *rest in maintenance*
- Parameters
  - 21136 network adapters → 446 709 360 pairs
  - 16 MiB message size, 8 messages; parallel execution
  - CPU-only, aligned `malloc`
  - Adaptive routing enabled
- Results: Many at 20 GiB/s, but average at 15 GiB/s → outliers! (even 1/20 of bandwidth)

→ Clip to 2 ms and better

- Interesting patterns: 22.5 groups, *anti*-diagonal elements



DFG 1 → DFG 2

MPI Startup Peculiarities

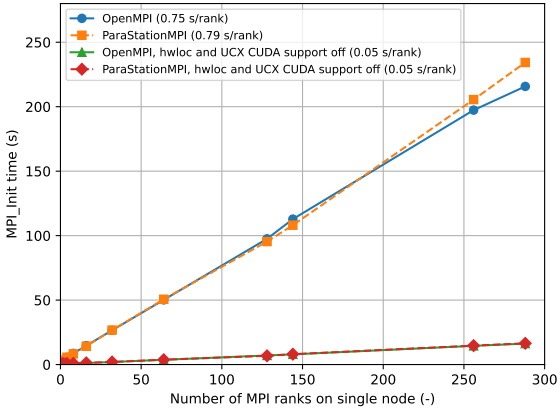
- Timing `MPI_Init()` (via OMB)
- Single node

→ Linear increase with number of tasks; nearly unit slope

- Disable hwloc CUDA lookup: much better

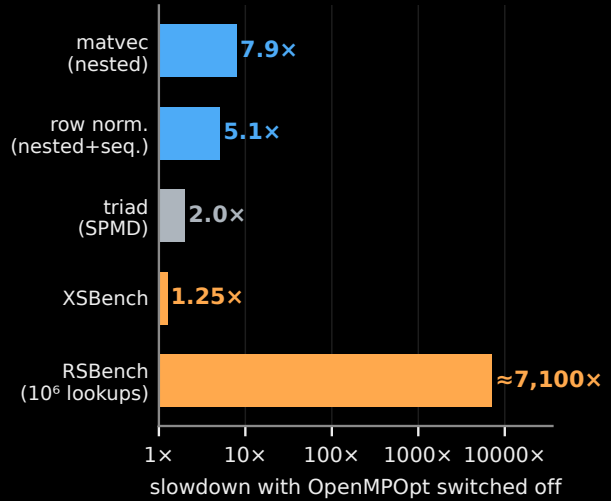
```
export HWLOC_COMPONENTS='-cuda,-nvml,-opencl'
export UCX_MODULES='^cuda'
```

- Still investigating further



JÜLICH Forschungszentrum

LLVM OpenMP Optimizations on JUPITER



- **OpenMP written naïvely** (nested `teams`/`parallel`):
  5–8× slower without optimizations
  Deglobalization, SPMDization important
- **Already SPMD code** still gains **2×**:
  Via kernel-wide analysis
- **RSBench**: 7100× slower, identical result

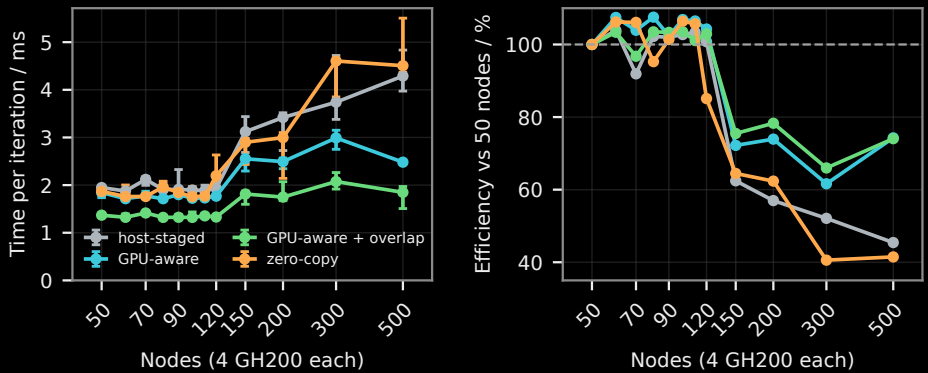
OpenMP optimization has large impact

JUPITER Booster • 1× NVIDIA GH200 • LLVM 20.1.8, upstream Clang • median of 5 runs • AI supported

For Johannes

Hybrid MPI + OpenMP Offload at Scale

3D Jacobi, weak scaling: 512^3 points, 1 MPI rank per GPU, 6-neighbour halo exchange



Overlap wins: interior via `target nowait` during the exchange: **20–26 %** faster.

GPU-aware MPI beats **host staging** by ∼9 %; **zero-copy** ≈ GPU-aware, more node-sensitive.

Beyond DFG: halo exchange $2-3\times$ slower

Enabled through LLVM & OpenMP `target`, `target nowait`, `omp_target_alloc` next to MPI, no CUDA

JUPITER Booster • 4 GH200 per node • upstream LLVM 20.1.8 (OpenMP offload) • Open MPI 5.0.8 (CUDA-aware) • median of 3 rounds • AI supported

For Johannes

Conclusion

- JUPITER: Exascale in Europe, nearly finished building up, 23 536 GPUs
- Interesting scientific results already coming out
- Network at scale is complicated

Conclusion

- JUPITER: Exascale in Europe, nearly finished building up, 23 536 GPUs
- Interesting scientific results already coming out
- Network at scale is complicated

Thank you for your attention!

a.herten@fz-juelich.de

JÜLICH Forschungszentrum

JUPITER

The Arrival of Exascale in Europe



fz-juelich.de/jupiter | #exa_jupiter

Funding Agencies:

EuroHPC Joint Undertaking

Federal Ministry of Education and Research

Ministry of Culture and Science of the State of North Rhine-Westphalia

JOINING FORCES

















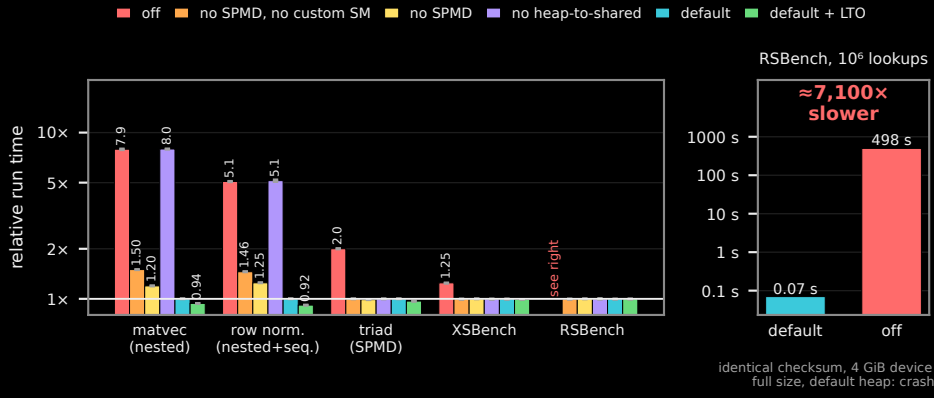






fz-juelich.de/jupiter

Optimization Details



- **Heap-to-shared off** ≈ **everything off** for nested kernels: the runtime's 512 B shared scratchpad split over 128 threads leaves 0 B per thread → device `malloc/free` per row.
- **Generic mode** costs 20–25 %, the generic state machine another 25–30 %; **device LTO** gains a further 6–8 %.
- Only in "off": the Attributor's heap-to-stack (RSBench) and the kernel-wide analysis proving "no nested parallelism" (triad, XSBench) have no separate switch.