APPLICATION-DRIVEN EXASCALE: THE JUPITER BENCHMARK SUITE SC24 ATLANTA

19 November 2024 | Andreas Herten and colleagues | Forschungszentrum Jülich, Jülich Supercomputing Centre

Member of the Helmholtz Association

JÜLICH Forschungszentrum

Content

- JUPITER
  - System Overview
  - Procurement
  - Framework
- Benchmark Suite
  - Creation Process
  - Benchmarks Overview
  - Infrastructure
  - Benchmark Details
    - The Grand Table
    - TCO Results
    - High-Scaling Results
  - Lessons Learned
- Concluding
  - Availability
  - Summary

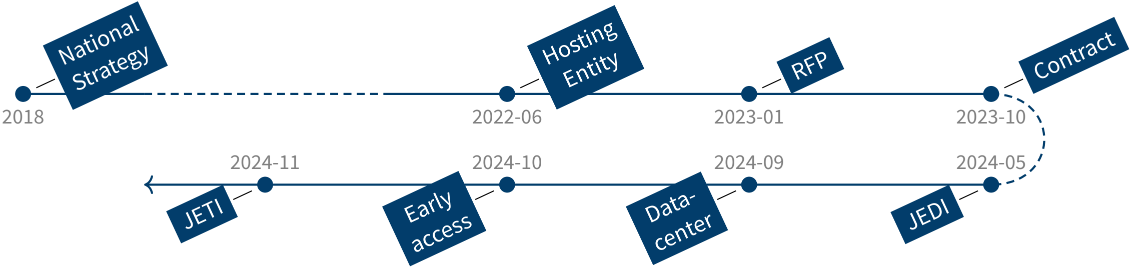
JUPITER

About JUPITER

- **JUPITER**: First European exascale supercomputer (HPL: 1 EFLOP/s)
  - Procured by EuroHPC JU, BMBF (Federal Ministry of Education and Research), MKW (NRW Ministry of Culture and Science)
  - Hosting entity: Forschungszentrum Jülich / Jülich Supercomputing Centre for Gauß Center for Supercomputing (GCS)

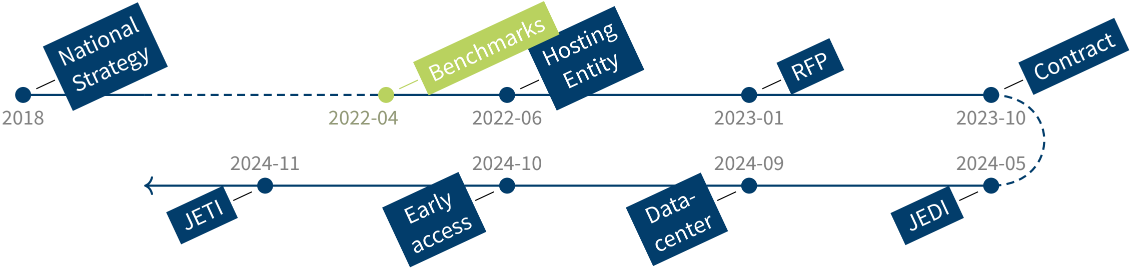
About JUPITER

- **JUPITER**: First European exascale supercomputer (HPL: 1 EFLOP/s)
  - Procured by EuroHPC JU, BMBF (Federal Ministry of Education and Research), 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 (HPL: 1 EFLOP/s)
  - Procured by EuroHPC JU, BMBF (Federal Ministry of Education and Research), MKW (NRW Ministry of Culture and Science)
  - Hosting entity: Forschungszentrum Jülich / Jülich Supercomputing Centre for Gauß Center for Supercomputing (GCS)


2018 National Strategy

2022-04 Benchmarks

2022-06 Hosting Entity

2023-01 RFP

2023-10 Contract

2024-05 JEDI

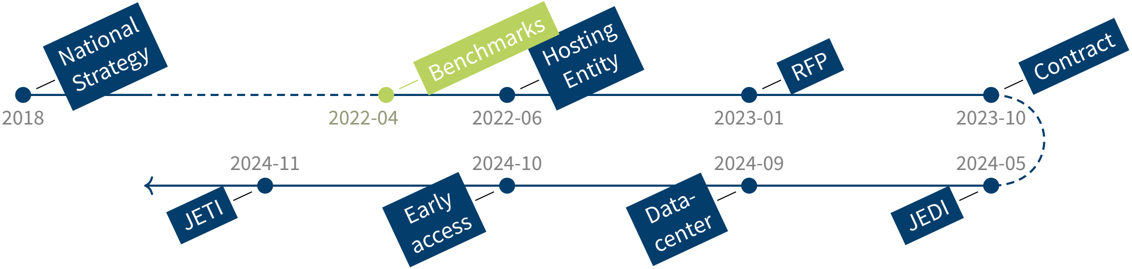
2024-09 Data-center

2024-10 Early access

2024-11 JETI

About JUPITER

- **JUPITER**: First European exascale supercomputer (HPL: 1 EFLOP/s)
  - Procured by EuroHPC JU, BMBF (Federal Ministry of Education and Research), MKW (NRW Ministry of Culture and Science)
  - Hosting entity: Forschungszentrum Jülich / Jülich Supercomputing Centre for Gauß Center for Supercomputing (GCS)



⇒ Paper: Background, methods, benchmarks, results, insights, release of software

JUPITER System Overview

- ParTec/Eviden consortium
- Implementing Modular Supercomputing Architecture

BullSequana XH3000

JUPITER System Overview

- ParTec/Eviden consortium
- Implementing Modular Supercomputing Architecture
- JUPITER Booster: High scalability, 1 EFLOP/s HPL, $>$ 35 EFLOP/s FP8
  $\approx$ 6000 nodes: 4 × Grace-Hopper superchip, 4 × network
- JUPITER Cluster: High versatility, 0.5 B/FLOP balance
  $\approx$ 1300 nodes: 2 × SiPearl Rhea1 (HBM), 1 × network

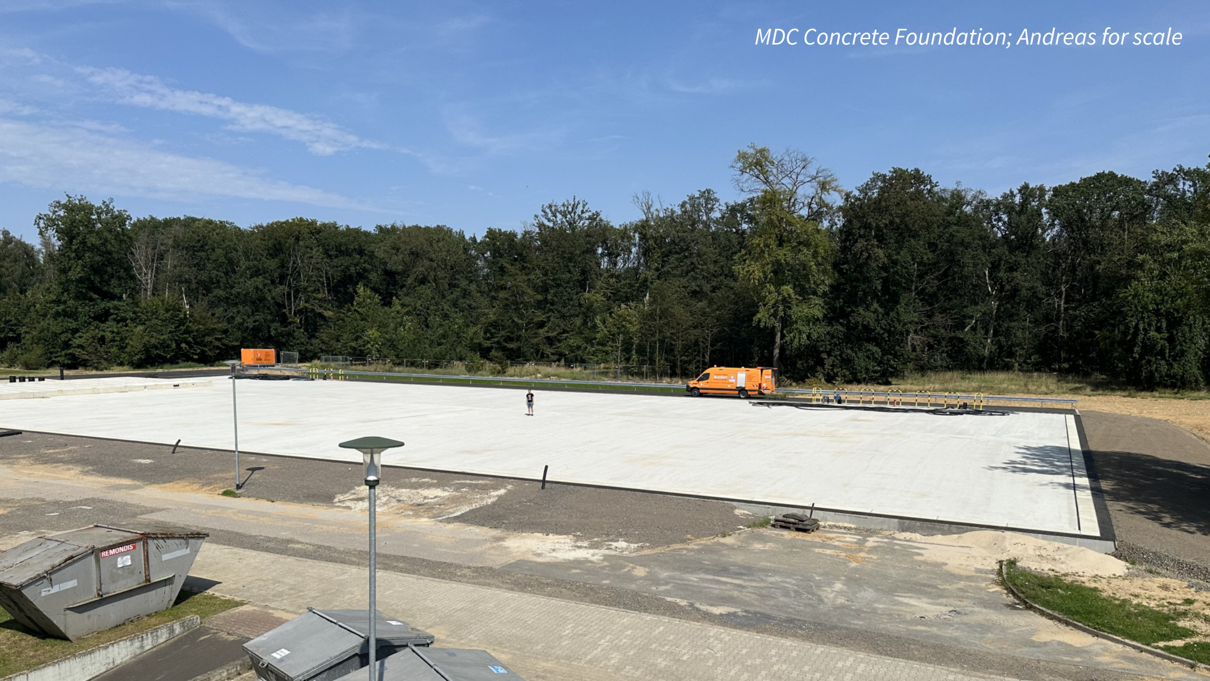
ParTec MODULAR SUPERCOMPUTING EVIDEN NVIDIA SIPEARL

JUPITER System Overview

- ParTec/Eviden consortium
- Implementing Modular Supercomputing Architecture
- JUPITER Booster: High scalability, 1 EFLOP/s HPL, > 35 EFLOP/s FP8
  ≈ 6000 nodes: 4× Grace-Hopper superchip, 4× network
- JUPITER Cluster: High versatility, 0.5 B/FLOP balance
  ≈ 1300 nodes: 2× SiPearl Rhea1 (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 🟥 🟩 🟥

BullSequana XH3000

ParTec MODULAR SUPERCOMPUTING EVIDEN NVIDIA SIPEARL



MDC Concrete Foundation; Andreas for scale

Delivery of first entry hall containers

Delivery of data hall

Photos by Herwig Zilken / FZJ

Preparations

JEDI: Preparation system

- 48 nodes (⅕ DragonFly group)
- May 2024: #1 Green500 (72.7 GFLOP/(s W))

JETI: Staging system

- 480 nodes (2 DragonFly groups)
- Nov 2024: #18 Top500 (83 PFLOP/s)

JUREAP: Research & Early Access Program

- $>$ 100 participants
- Currently: Selection stage (Lighthouse)

By Forschungszentrum Jülich

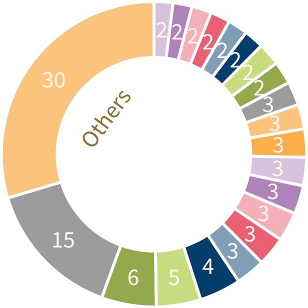
By Forschungszentrum Jülich/Eviden

By Robert Wiedemann on Unsplash

But how did we get there?

JSC Workload

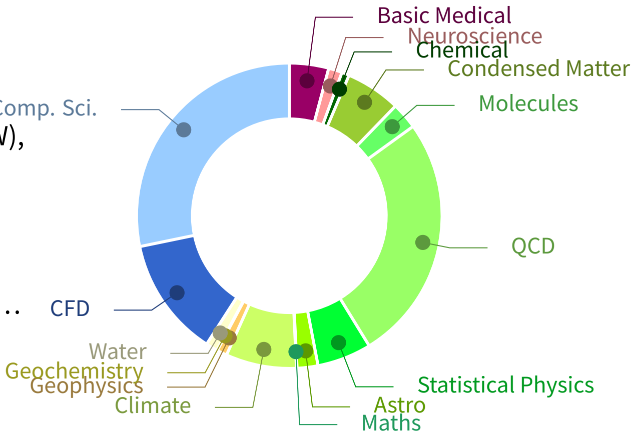
- JSC: HPC resources for Forschungszentrum campus, state (NRW), Germany, Europe
- Compute time through peer-review
- Heterogeneous workload
- Physics, climate, biology, chemistry, AI, …
- **Goal:** Respect current & anticipated workload in procurements of new systems; incl. domains, methods, programming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



Programs / GPUh (2020)
JUWELS Booster

JSC Workload

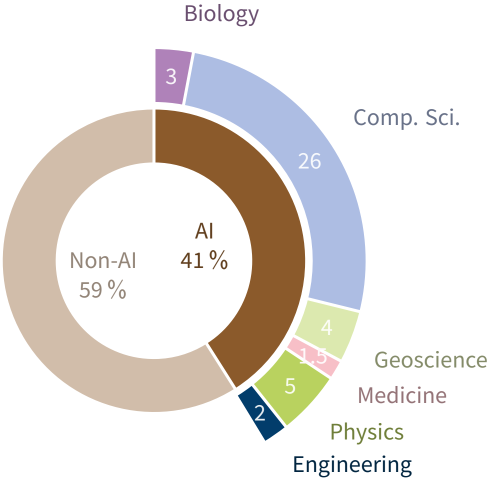
- JSC: HPC resources for Forschungszentrum campus, state (NRW), Germany, Europe
- Compute time through peer-review
- Heterogeneous workload
- Physics, climate, biology, chemistry, AI, …
- **Goal:** Respect current & anticipated workload in procurements of new systems; incl. domains, methods, programming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



Compute Time per Domain (2024-1)
JUWELS Booster

JSC 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

Framework

- Procuring entity: EuroHPC JU
- Hosting entity: JSC
- Procurement: JSC, with support from EuroHPC, national ministries
- Compute time allocations: GCS (Germany), JU (Europe)
- 500 M€ project budget

- Strong investment from all sides →

replicability, reproducibility, reusability $_{R}R^{R}$

- Benchmarks
  - **Applications** (Base TCO, High-Scaling, MSA)
  - Synthetic
- JUPITER Cluster **and** Booster

Framework

- Procuring entity: EuroHPC JU
- Hosting entity: JSC
- Procurement: JSC, with support from EuroHPC, national ministries
- Compute time allocations: GCS (Germany), JU (Europe)
- 500 M€ project budget

- Strong investment from all sides →

replicability, reproducibility, reusability $_{\mathrm{R}}\mathrm{R}^{\mathrm{R}}$

- Benchmarks
  - **Applications** (Base TCO, High-Scaling, MSA)
  - Synthetic
- JUPITER Cluster **and** Booster

Evaluation Target

Mainly: Total Cost of Ownership (TCO)

- Proposals ranked by *workload intensity* (*how much workload over system lifespan*)
- Also energy consumption respected (simplified)
- Master formula: calculate value for ranking
- Normalized metric/FOM: runtime
- Applications-based

Evaluation Target

Mainly: Total Cost of Ownership (TCO)

- Proposals ranked by *workload intensity* (*how much workload over system lifespan*)
- Also energy consumption respected (simplified)
- Master formula: calculate value for ranking
- Normalized metric/FOM: runtime
- Applications-based

New: High-Scaling Benchmarks

- For Exascale procurement → respect large-scaleness of system
- Run dedicated workload on entire system
- Applications-based

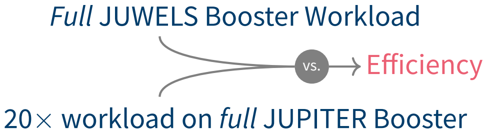
Evaluation Target

Mainly: Total Cost of Ownership (TCO)

- Proposals ranked by *workload intensity* (*how much workload over system lifespan*)
- Also energy consumption respected (simplified)
- Master formula: calculate value for ranking
- Normalized metric/FOM: <u>runtime</u>
- Applications-based

New: High-Scaling Benchmarks

- For Exascale procurement → respect large-scaleness of system
- Run dedicated workload on entire system
- Applications-based
- Method:



Full JUWELS Booster Workload

vs. → Efficiency

20× workload on *full* JUPITER Booster

  - *Full*: 50 PFLOP/s vs. 1000 PFLOP/s th. peak
  - Instructions/rules to determine workload
  - Memory variants: (tiny,) small, medium, large

S M L

Evaluation Target

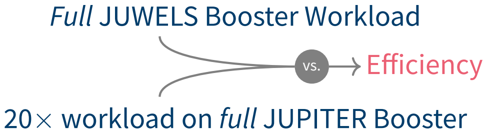
Mainly: Total Cost of Ownership (TCO)

- Proposals ranked by *workload intensity* (*how much workload over system lifespan*)
- Also energy consumption respected (simplified)
- Master formula: calculate value for ranking
- Normalized metric/FOM: <u>runtime</u>
- Applications-based

Also: Synthetic benchmarks

New: High-Scaling Benchmarks

- For Exascale procurement → respect large-scaleness of system
- Run dedicated workload on entire system
- Applications-based
- Method:



Full JUWELS Booster Workload

vs. → Efficiency

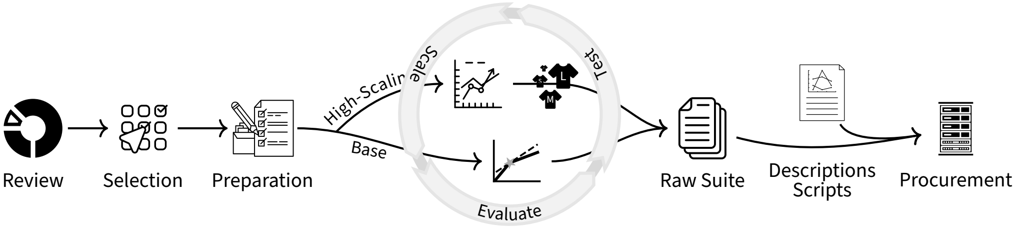
20× workload on *full* JUPITER Booster

- *Full*: 50 PFLOP/s vs. 1000 PFLOP/s th. peak
- Instructions/rules to determine workload
- Memory variants: (tiny,) small, medium, large

S M L

Benchmark Suite

Creation Process



Review

Selection

Preparation

High-Scalin

Base

Scale

Test

Evaluate

Raw Suite

Descriptions
Scripts

Procurement

Benchmarks Overview

- 16 application benchmarks (4 de-selected for actual procurement)
  Cross-section of domains and methods, 3× AI
- 7 synthetic benchmarks
- Extensive descriptions
- Right: Patterns of 7 Dwarfs

		Dense LA ●	Sparse LA ●	Spectral ●	Particle ●	Structured Grid ●	Unstructured Grid ●	Monte Carlo ●
Benchmark	**Domain**	**Dwarfs**						
Amber*	MD			●	●			
Arbor	Neurosci.	●	●				●	
Chroma-QCD	QCD		●			●		●
GROMACS	MD			●	●			
ICON	Climate		●			●		
JUQCS	QC		●					
nekRS	CFD		●	●			●	
ParFlow*	Earth Sys.		●			●		
PIConGPU	Plasma				●		●	
Quantum Espresso	Materials Sci.	●		●	●			
SOMA*	Polymer Sys.				●			●
MMoCLIP	AI (MM)	●						
Megatron-LM	AI (LLM)	●						
ResNet*	AI (Vision)	●						
DynQCD	QCD		●			●		●
NAStJA	Biology					●		●
Graph500	Graph	*Graph Traversal (D. 9)*						
HPCG	CG		●					
HPL	LA	●						
IOR	Filesys.	*Input/Output*						
LinkTest	Network	*P2P, Topology*						
OSU	Network	*Message Exchange, DMA*						
STREAM	Memory	*Regular Access*						

Infrastructure

- Preparation system: JUWELS Booster (73 PFLOP/s peak, 44 PFLOP/s HPL), JURECA-DC (15 PFLOP/s, 9 PFLOP/s)
  Dependencies: EasyBuild; versioned environment modules
- JUBE workflow environment for every benchmark, similar structure; implicit documentation; platform-independence through inheritance
  → Continuous Benchmarking with *exaCB*
- Extensive description (incl. guidelines, rules), similar structure
- Git, Git submodules for sources (if possible)
- Management: Teams with captains and domain scientists, meetings every 2 weeks, hackathons, scale days, *11 step program*, Gitlab issues

```
- name: systemParameter
  init_with: platform.xml
  parameter:
    - name: preprocess
      _: $modules
    - name: executable
      _: myapp
    - name: args_exec
      _: input.json
    - name: queue
      tag: "baseline|scaling_base|scaling
      _: booster
    - name: queue
      tag: "exa_tiny|exa_small|exa_medium
      _: largebooster
```

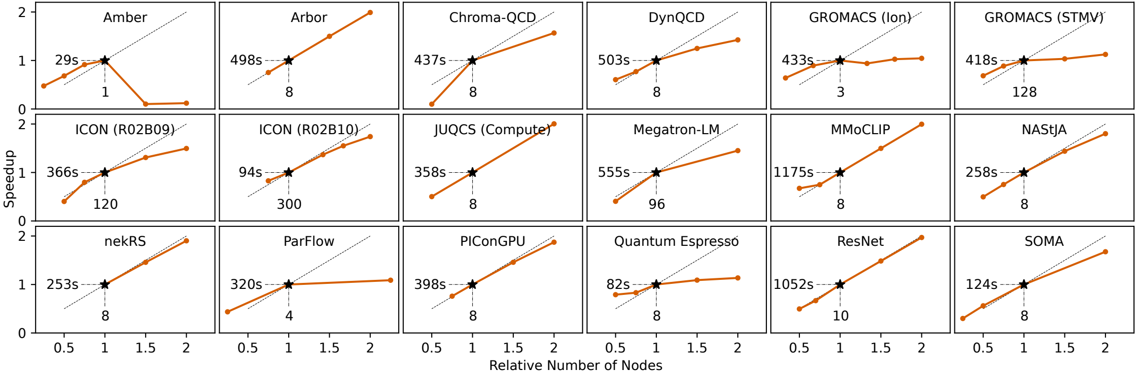
JUBE example

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 G	B C	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		✓		✓	

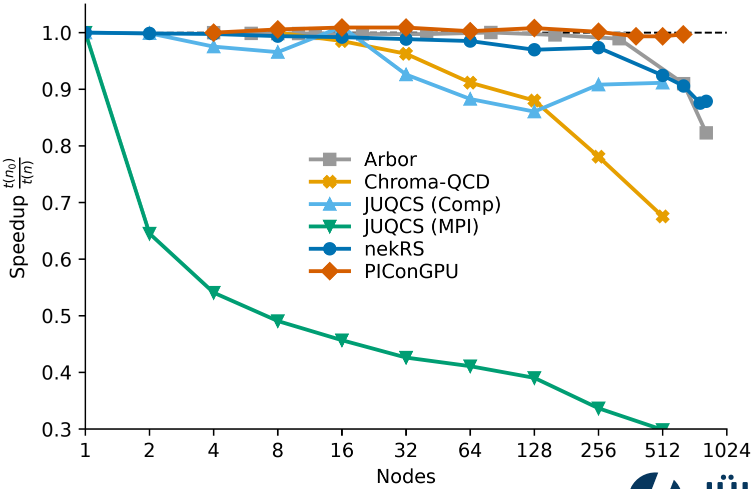
TCO Base Result Grid



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

High-Scaling Results

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



Lessons Learned

Performance models useful, even if simple

Domain decomposition scripts/rules for unknown system makeup

Intensive feedback for app devs

Verification is hard

Applications

Suite: resource-intensive → aim for short turn-around times during dev

Artificially limit benchmarks on prep system to mimic future system

Extensive, balanced execution rules/guidelines

Benchmarks

Multi-system procurement → benchmark balance 😰

Collaboration, tools

Bias towards incremental update

Limiter: time, on *all* sides → reuse!

Procurement

Concluding

Availability

- All benchmark workflows, descriptions, data released online
- Code GitHub `jubench` meta-repository, Zenodo meta-record

  Individual repos: Arbor Amber Chroma-LQCD GROMACS ICON JUQCS Megatron-LM MMoCLIP nekRS ParFlow PIConGPU Quantum ESPRESSO ResNet SOMA DynQCD (CPU) NAStJA (GPU) Graph500 HPCG HPL IOR LinkTest OSU Micro-Benchmarks STREAM STREAM (GPU)
- Paper SC24 Proceedings, arXiv:2408.17211

  Including extensive SC reproducibility appendix

GitHub

Proceedings

Conclusions

- JUPITER: First European exascale system (EuroHPC JU, BMBF, MKW; hosted at JSC); currently being built
  - **JUPITER Benchmark Suite**: Benchmark suite for JUPITER procurement *and beyond*
  - Application workloads from variety of domains, balanced profiles
  - Reference results, lessons learned provided
  - All benchmarks published as open source software: `github.com/FZJ-JSC/jubench`
  - Next: Continuous benchmarking (*exaCB*), housekeeping, extension

Conclusions

- JUPITER: First European exascale system (EuroHPC JU, BMBF, MKW; hosted at JSC); currently being built
- **JUPITER Benchmark Suite**: Benchmark suite for JUPITER procurement *and beyond*
- Application workloads from variety of domains, balanced profiles
- Reference results, lessons learned provided
- All benchmarks published as open source software: `github.com/FZJ-JSC/jubench`
- Next: Continuous benchmarking (*exaCB*), housekeeping, extension

Huge team effort at JSC

Andreas Herten, Sebastian Achilles, Damian Alvarez, Jayesh Badwaik, Eric Behle, Mathis Bode, Thomas Breuer, Daniel Caviedes-Voullième, Mehdi Cherti, Adel Dabah, Jan Ebert, Thomas Eickermann, Salem El Sayed, Wolfgang Frings, Ana Gonzalez-Nicolas, Eric B. Gregory, Kaveh Haghighi Mood, Thorsten Hater, Jenia Jitsev, Chelsea John, Stefan Kesselheim, Jan H. Meinke, Catrin I. Meyer, Pavel Mezentsev, Jan-Oliver Mirus, Stepan Nassyr, Carolin Penke, Manoel Römmer, Ujjwal Sinha, Benedikt von St. Vieth, Olaf Stein, Estela Suarez, Dennis Willsch, Ilya Zhukov

Conclusions

- JUPITER: First European exascale system (EuroHPC JU, BMBF, MKW; hosted at JSC); currently being built
- **JUPITER Benchmark Suite**: Benchmark suite for JUPITER procurement *and beyond*
- Application workloads from variety of domains, balanced profiles
- Reference results, lessons learn
- All benchmarks publis vare: `github.com/FZJ-JSC/jubench`
- Next: Continuous benc ekeeping, extension

Thank you for your attention!

a.herten@fz-juelich.de

Huge team effort at JSC

Andreas Herten, Sebastian Achilles, Damian Alvarez, Jayesh Badwaik, Eric Behle, Mathis Bode, Thomas Breuer, Daniel Caviedes-Voullième, Mehdi Cherti, Adel Dabah, Jan Ebert, Thomas Eickermann, Salem El Sayed, Wolfgang Frings, Ana Gonzalez-Nicolas, Eric B. Gregory, Kaveh Haghighi Mood, Thorsten Hater, Jenia Jitsev, Chelsea John, Stefan Kesselheim, Jan H. Meinke, Catrin I. Meyer, Pavel Mezentsev, Jan-Oliver Mirus, Stepan Nassyr, Carolin Penke, Manoel Römmer, Ujjwal Sinha, Benedikt von St. Vieth, Olaf Stein, Estela Suarez, Dennis Willsch, Ilya Zhukov

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

EuroHPC
Joint Undertaking

Bundesministerium
für Bildung
und Forschung

Ministerium für
Kultur und Wissenschaft
des Landes Nordrhein-Westfalen

GCS
Gauss Centre for Supercomputing

JÜLICH
Forschungszentrum

ParTec
MODULAR SUPERCOMPUTING

EVIDEN

NVIDIA

SIPEARL

IBM

fz-juelich.de/jupiter

Appendix

Application Descriptions

TCO

Amber	Molecular dynamics; STMV use-case; single node
ParFlow	Hydrology; ClayL use-case
SOMA	Polymer simulation; Monte-Carlo
ResNet	AI: Computer vision; TensorFlow
DynQCD	Particle physics; CPU-only
GROMACS	Molecular dynamics: GluCL, STMV use-cases; multi-node
ICON	Climate simulation; atmosphere with R02B09, R02B10; many nodes
Megatron-LM	AI: LLM; PyTorch
MMoCLIP	AI: Mixed-Modal (text, image); PyTorch
Quantum ESPRESSO	Electronic structure; Car-Parrinello test-case
NAStJA	Biology; CPU-only, MPI-only

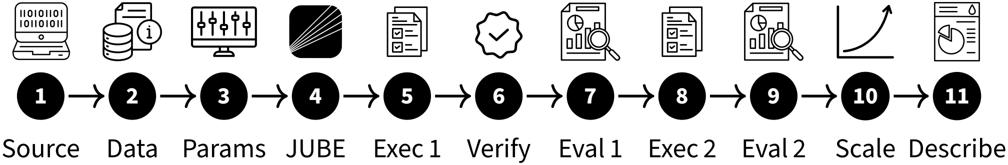
TCO & High-Scaling

Arbor	Neuroscience; busyring benchmark
Chroma	Particle physics; hybrid-Monte-Carlo test; QUDA with JIT; max 512 nodes
JUQCS	Quantum computer simulator; gate-based simulation; communication-heavy; max 512 nodes; Cluster-Booster version (MSA)
nekRS	Fluid dynamics; Rayleigh-Bénard convection use-case
PIConGPU	Plasma physics; Kelvin-Helmholtz instability use-case

JUPITER Application Benchmarks

- JUPITER: Largest procurement to date
- >18 months of work
- >30 people involved
- 1(-3) associated people (*captains*) per benchmark
- Meetings every two weeks
- Gitlab issue tracker, status tracker (11 points)

1. Source code available
2. Input data available
4. JUBE integration
11. Description, documentation


1 Source
2 Data
3 Params
4 JUBE
5 Exec 1
6 Verify
7 Eval 1
8 Exec 2
9 Eval 2
10 Scale
11 Describe