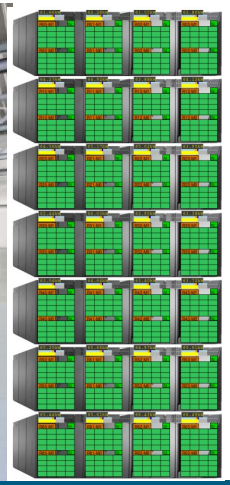
Mitglied der Helmholtz-Gemeinschaft



JÜLICH
FORSCHUNGSZENTRUM

Extreme-scaling applications *en route* to exascale

2016-04-26 |

Dirk Brömmel, Wolfgang Frings & **Brian J. N. Wylie**
Jülich Supercomputing Centre

b.wylie @ fz-juelich.de

EASC2016
Exascale Applications & Software Conference
April 25-29 | Stockholm

Outline

Motivation

Jülich Blue Gene Extreme Scaling Workshops & *High-Q Club*

- 2016 JUQUEEN workshop

Extreme-scaling lessons

- Performance issues
- Tools
- Coaching & training

Exascale?

Diversity of current leadership HPC computer systems

Cores	Name	System	Processor	(cores)	Accelerator	(cores)	Site
3 120 000	Tianhe-2	NUDT IVB-FEP	Xeon(12C)	384 000	31S1P(57M)	2 736 000	NSCC-GZ, China
1 572 864	Sequoia	IBM Blue Gene/Q	PowerPC(16C)	1 572 864			LLNL, USA
786 432	Mira	IBM Blue Gene/Q	PowerPC(16C)	786 432			ANL, USA
705 024	Kei	K computer	SPARC64(8C)	705 024			RIKEN AICS, Japan
560 640	Titan	Cray XK7	Opteron(16C)	299 008	K20x(14S)	261 632	ORNL, USA
462 462	Stampede	Dell PowerEdge	Xeon(16C)	102 400	SE10P(61M)	390 400	TACC, USA
458 752	***JUQUEEN***	IBM Blue Gene/Q	PowerPC(16C)	458 752			JSC, Germany
452 400	Blue Waters	Cray XE6+XK7	Opteron(16C)	393 600	K20x(14S)	58 800	NCSA, USA
185 088	Hazel Hen	Cray XC40	Xeon(12C)	185 088			HLRS, Germany
147 456	SuperMUC.1	IBM iDataPlex	Xeon(8C)	147 456			LRZ, Germany
115 984	Piz Daint	Cray XC30	Xeon(8C)	42 176	K20x(14S)	73 808	CSCS, Switzerland
86 016	SuperMUC.2	IBM NeXtScale	Xeon(14C)	86 016			LRZ, Germany

Extreme scaling

Image: Chronicle / Michael Macor

Image: Forschungszentrum Jülich

Background

Technology trends

- energy-efficient processors with many relatively weak cores
- organised in compute nodes with restricted shared memory
- combined with co-processors and accelerators with separate address space

Current generation of leadership supercomputers have many thousands of processors/cores

- and exascale computer systems are expected to have many more

Applications need to be extremely scalable and adaptable to effectively exploit such systems

- hybrid parallelisation combining message-passing, multi-threading, vectorisation
- careful management of communication/synchronisation, memory and file I/O

Motivation

Extreme-scaling applications is highly challenging

- correctness and performance issues only seen at large scale
  - *new issues encountered with each doubling of scale (or perhaps factor of ten)*

Application code teams need good support from supercomputing centres

- expert advice and training regarding most suitable algorithms, compilers, libraries, tools
  - *often specific to a particular code, but sometimes more generally applicable*
- access to large system configurations for testing and validation
  - *often only for short periods, but potentially disruptive*

Supercomputing centres have a variety of support approaches

- favouring capability versus capacity usage
- organising dedicated workshops and documenting successes

History of JSC extreme scaling workshops

First workshops with JUBL Blue Gene/L (2006) and JUGENE Blue Gene/P (2008)

Three JUGENE Blue Gene/P Extreme Scaling Workshop instances (2009, 2010, 2011)

- 72 racks, 288k cores, #1 in world
- from 27 international teams, 23 successfully ran 26 codes on full system
- 3 ACM Gordon Bell Prize finalists + George Michael Memorial PhD Fellowship awardee
- wide spectrum of application fields represented
- results in technical reports [juser=8924,9600,15866]

Last year's appendix to Porting & Tuning workshop with JUQUEEN Blue Gene/Q (Feb 2015)

- 7 international teams, all successfully ran their codes on full system within 24 hours
- results in technical report FZJ-JSC-IB-2015-01 [http://juser.fz-juelich.de/record/188191]

Workshop goals

Provide opportunity to try to scale application execution to full JUQUEEN

- 28 racks, 28k processors/nodes, 448k cores, 1.75M threads
- adapt to available node memory

Investigate application strong/weak scalability

- identify scaling efficiency or scaling limits

Tune application/configuration to optimise performance (at large scale)

- examine/remedy execution performance and scaling limiters

Qualify for High-Q Club membership!

Prepare for extreme/large-scale production campaign on JUQUEEN?

Scaling

Common variants

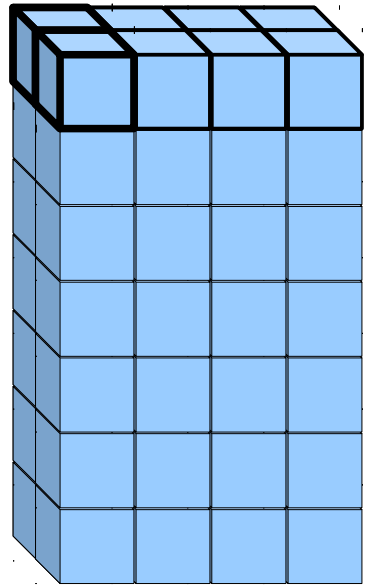
- **strong**: fixed total problem size, part per compute node diminishes with scale
  - *time to solution expected to decrease proportionally,*
  - *however, computation/communication increasingly unfavourable*
- **weak**: fixed problem size per compute node, total problem size increases with scale
  - *time to solution expected to remain constant,*
  - *since constant local computation and neighbour communication,*
  - *however, collective/global communication costs still get more expensive*

Either (or both) are legitimate, depending on the goal

JUQUEEN Blue Gene/Q

28 racks arranged as 7 rows of 4

- a rack has two mid-planes each with 512 compute nodes
  - *7x4x2 topology*
- job allocations/partitions are 3D sub-arrays of mid-planes
  - *no torus connectivity for odd-sized dimensions*
  - *square partition minimise largest dimension*
  - *16 racks therefore often optimal*
- each partition is physically isolated from others
  - *only GPFS filesystem is shared*



Compute node configurations

Each Blue Gene/Q compute node has a PowerPC A2 processor and 16 GB of memory

- 16 cores available for applications, each with 4 hardware threads
  - *full architecture performance requires use of at least 2 hardware threads*
- available memory is partitioned equally to MPI processes, and shared by OMP threads

64 MPI ranks per node can scale to full JUQUEEN, but is constraining

- *memory required by MPI on each node increases with total number of ranks*
- *time for collective MPI operations also increases with scale*

Hybrid MPI+OMP is therefore recommended, and provides considerable flexibility

- single MPI ranks per node each with up to 64 OpenMP threads optimise memory use
- 16 MPI ranks each with 4 OpenMP threads is a popular compromise
- other combinations are also worth investigating: 1p64t, 2p32t, 4p16t, 8p8t, 16p4t, 32p2t

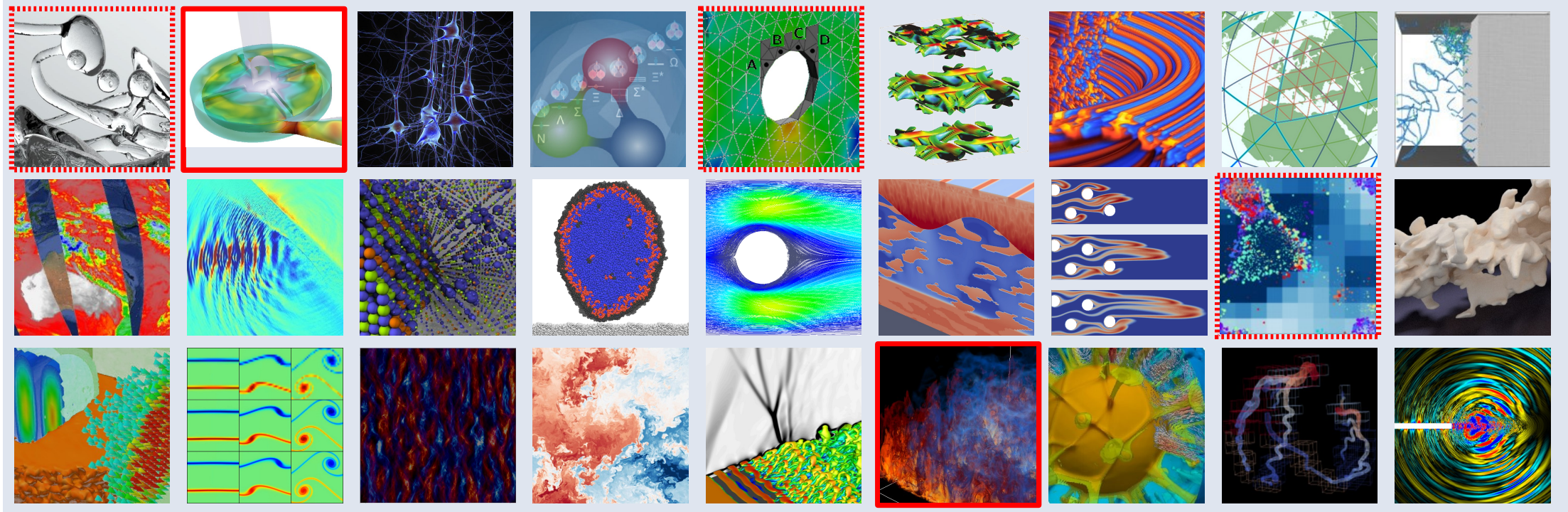
Qualifying for High-Q Club membership

Wide spectrum of applications, therefore no common set of criteria

Membership criteria are flexible (open for discussion!)

- based on discussion with developers and within JSC
  - *run a non-trivial example, ideally very close to production runs*
  - *file I/O is highly desirable, but optional*
  - *vectorisation/SIMDization for quad-FPU desirable, but optional*
  - *use all available cores (and preferably hardware threads)*
  - *evidence of strong and/or weak scalability, demonstrating benefit of additional racks*
  - *compare to peak performance characteristics, if possible*
- may be relaxed or graduated in future
  - *scaling to >75% of system, exploitation of HWTs vs. memory?*

High-Q Club members



CIAO, ***Code_Saturne***, CoreNeuron, dynQCD, *FE2TI*, FEMPAR, Gysela, ICON, IMD, JURASSIC, JuSPIC, KKRnano, LAMMPS(DCM), MP2C, muPhi, Musubi, *NEST*, OpenTBL, PEPC, PMG+PFASST, PP-Code, psOpen, SHOCK, ***SLH***, Terra-Neo, waLBerla, ZFS

Diverse set of 27 codes (incl. ***two new members*** from 2016 Extreme Scaling Workshop) from engineering, molecular dynamics, neuroscience, plasma physics, and climate and Earth science

XSW16 participating code-teams

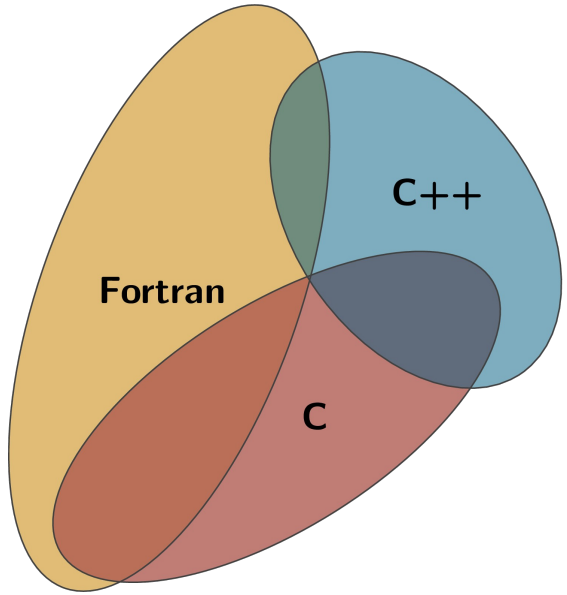
Code	Application area	Institution
CIAO	multiphysics, multiscale Navier-Stokes solver for turbulent reacting flows in complex geometries	RWTH-ITV, Germany
Code_Saturne	finite volume method CFD to solve Navier-Stokes equations	STFC Daresbury Lab, UK
ICI	implicit finite-element formulation including anisotropic mesh adaptation	École Centrale de Nantes, France
iFETI	implicit solvers for finite-element problems in nonlinear hyperelasticity & plasticity	U. Koln & TUB Freiberg, Germany
NEST-import	module to load neuron and synapse information into the NEST neural simulation tool	Blue Brain Project, Switzerland
p4est	library for parallel adaptive mesh refinement and coarsening	U. Bonn, Germany
PFLOTRAN	subsurface flow and reactive transport in heterogeneous rock	Amphos[21], Spain & FZJ-IEK6, Germany
Seven-League Hydro (SLH)	astrophysical hydrodynamics with focus on stellar evolution	Heidelberger ITS, Germany

XSW16 code characteristics

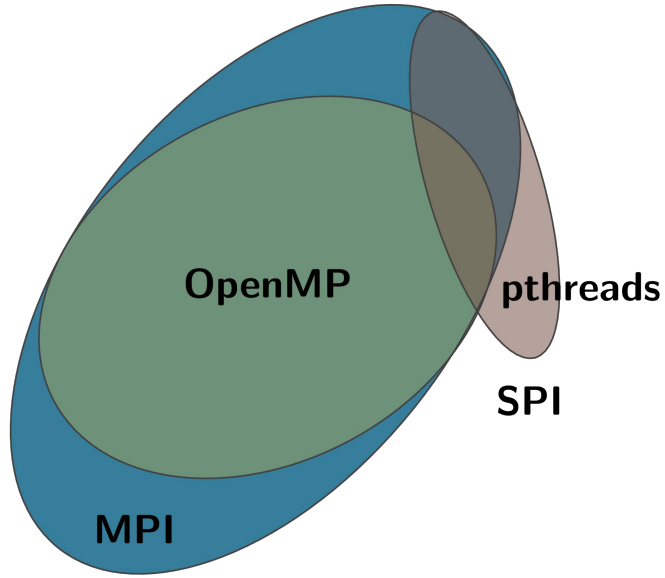
Code	Language(s)			MPI	OMP	Concurrency	File I/O
CIAO			Ftn	16		16: 458 752	MPI-IO, HDF5
Code_Saturne	C		Ftn	16	4	64: 1 835 008	MPI-IO
ICI		C++		16		16: 458 752	MPI-IO
iFETI	C	C++		32		32: 917 504	
NEST-import		C++		1	16	16: 458 752	HDF5 (MPI-IO)
p4est	C			32		32: 917 504	(MPI-IO)
PFLOTRAN			F03	16		16: 131 072	HDF5 (SCORPIO)
SLH			F95	16	4	64: 1 835 008	MPI-IO

Main application programming languages (excluding external libraries), parallelisation including maximal process/thread concurrency (per compute node and overall), and file I/O implementation (in parenthesis if not used for scaling runs)

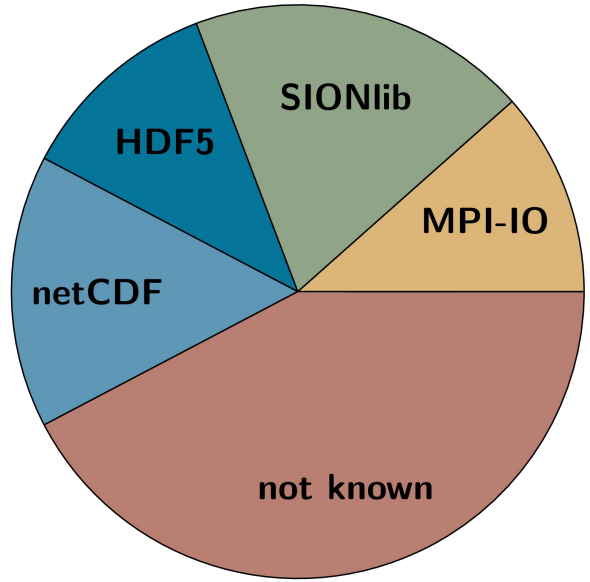
High-Q Club member code implementation analysis



Programming language



Parallelisation model



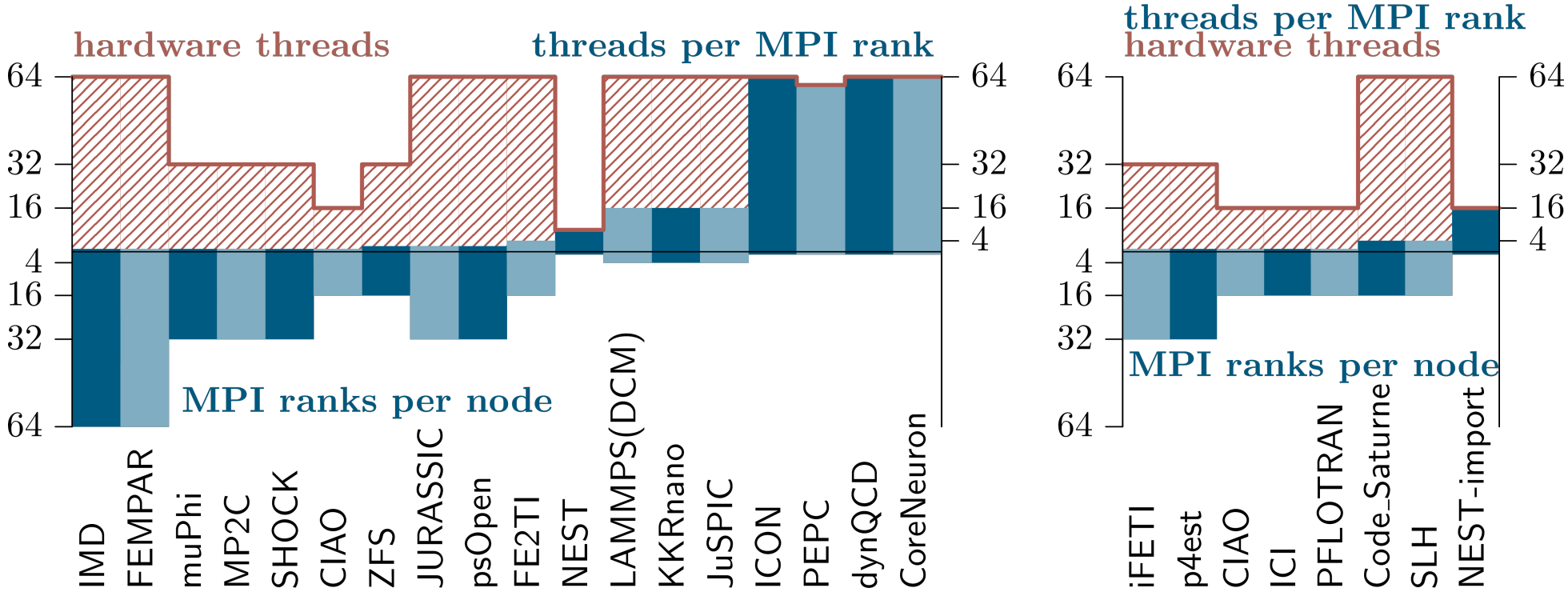
File I/O

Standard programming languages, parallelisation models and file I/O libraries dominate, supporting application portability

- typically codes run on clusters and other leadership systems

Custom variants only used when performance benefits are demonstrable

Compute node concurrency



MPI is almost ubiquitous (with only dynQCD instead using proprietary SPI) either used on its own or in conjunction with OpenMP or POSIX threading.

Node over-subscription to exploit hardware threading often more than doubles performance, but requires efficient use/sharing of compute node memory.

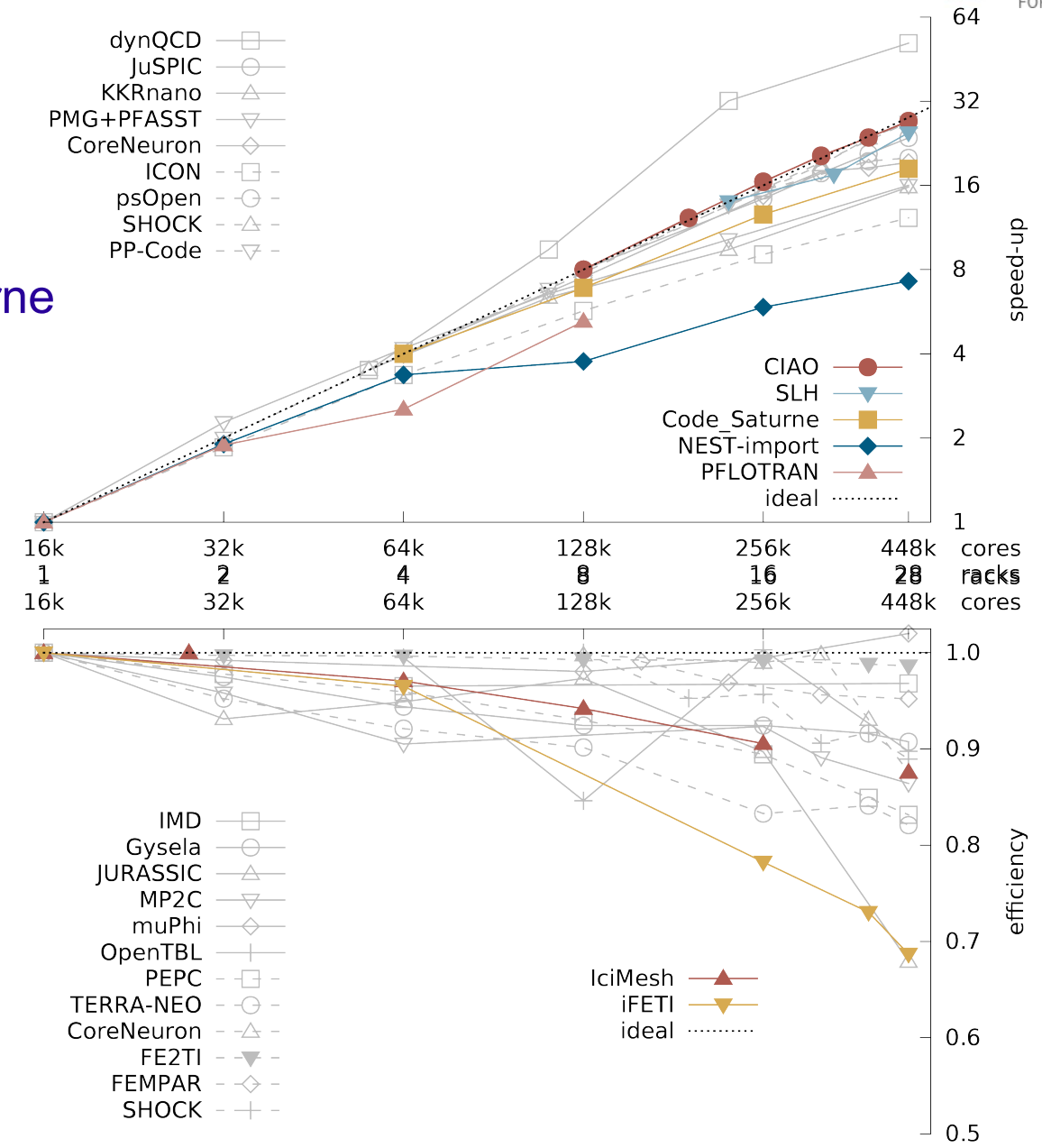
XSW16 scaling results

Excellent *strong scaling* for CIAO & SLH,
and acceptable 19x (68%) for Code_Saturne

NEST-import & PFLOTRAN inhibited by
inappropriate parallel file I/O

[p4est execution times only a
few seconds, so not included]

Good *weak scaling* for ICI-mesh,
and still acceptable for iFETI
(but poorer than its FE2TI predecessor)



Disruptive technology for exascale

Blue Gene/Q is highly reliable (also for repeatable performance) and relatively power efficient

- 2011 technology can no longer be considered state-of-the-art

Applications are expected to need to be more than simply scalable to exploit exascale systems

- adaptive, dynamic, heterogeneous, malleable, resilient, variability-tolerant, …

Potential alternatives to MPI message-passing and OpenMP (/POSIX) multithreading?

- ARMCI, CHARM++, (Coarray)Fortran, HPX, ompSs, RAJA, UPC, XMP, X10, …
- actively researched at JSC, promoted and some available on JUQUEEN, but ...
  - *not ready for scale of JUQUEEN?*
  - *don't deliver performance?*
  - *not sufficiently usable?*

High-Q Club only documents what has been used successfully, not failures or work in progress!

File I/O – too much, too many, too slow

File I/O remains the most common impediment to application scalability

- Code_Saturne: MPI collective file read of 618 GiB mesh input data
  - *however, writing simulation output was disabled as a known scalability impediment*
- SLH: 264 GiB of output written to single file using MPI-I/O
  - *separate files for each process is impractical due to creation (metadata) bottleneck and expensive post-processing*
- ICI: problems encountered reading 1.7 TiB mesh files (>64k ranks)
- PFLOTRAN: HDF5 file read/write impractical (>10k ranks)
  - *SCORPIO library was unavailable for two-stage aggregator I/O*
- CIAO: wrote 9 TiB to single file with MPI file I/O
  - *single shared files limit available filesystem bandwidth*
  - *no benefit observed from MPI-I/O (ROMIO) hints*

Tools are available to assist

SIONlib library for scalable native I/O to task-local files: http://www.fz-juelich.de/jsc/sionlib

- 129 GiB/s (63%) *reading* and 180 GiB/s (88%) *writing* of 205 GiB/s theory, attained with separate files per IONode (288 on JUQUEEN)
- already used effectively by five (of 27) HiQ member codes and currently being adopted by others, including NEST and SLH

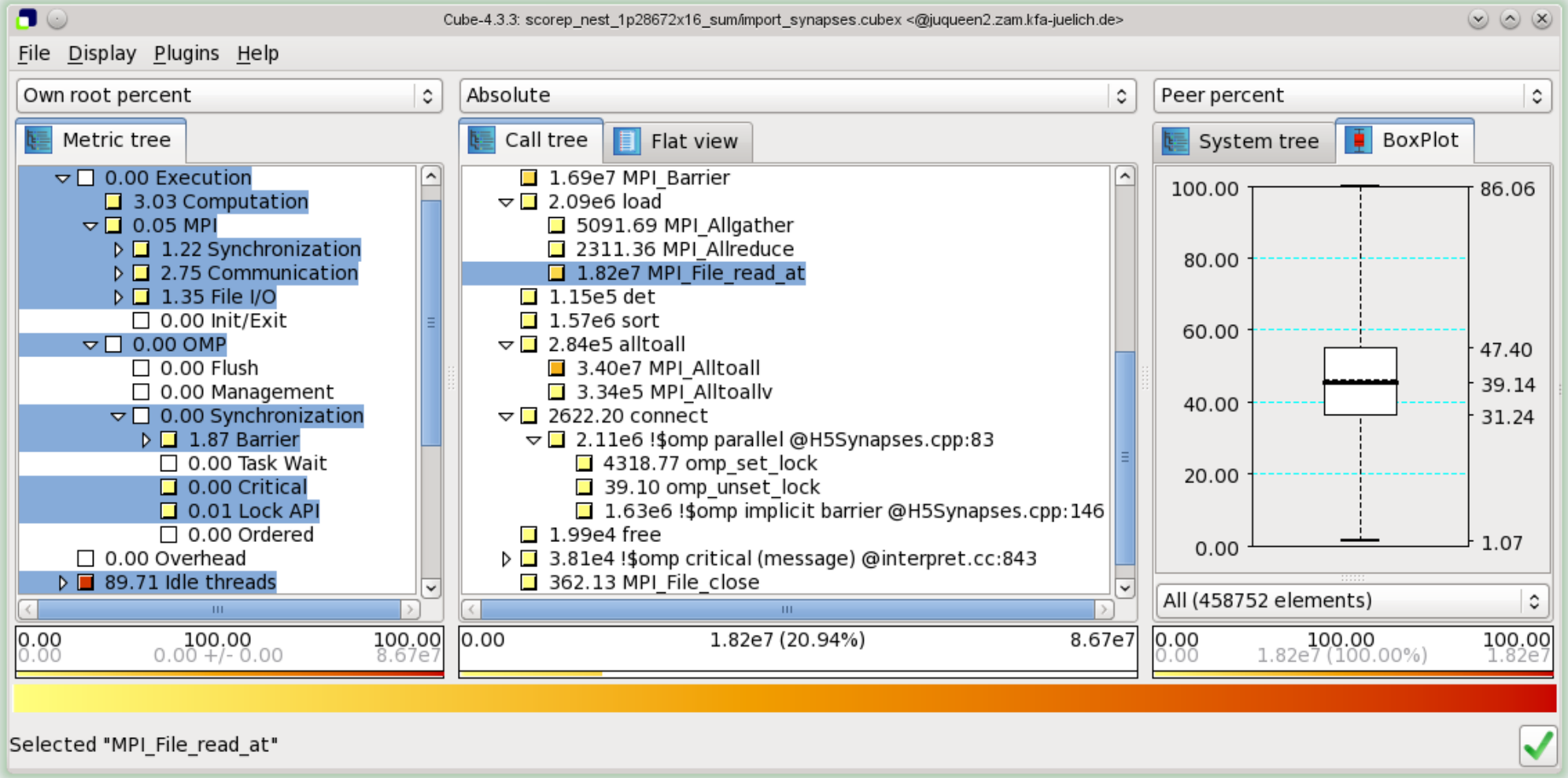
Score-P instrumentation & measurement infrastructure used by Scalasca profile & trace analyses (as well as Periscope, TAU & Vampir): http://www.scalasca.org/

- NEST-import: 1.9 TiB of neuron and synapse information
  - *HDF5 file format used for portability, uses MPI-I/O internally*
  - *only achieved a small fraction of GPFS filesystem bandwidth*
  - *reverts to individual file reads when collective not suitable*
  - *profiling identified structural mismatch between HDF5 file & internal data objects, which is now being resolved*

scalasca

JÜLICH FORSCHUNGSZENTRUM

Scalasca summary profile of NEST-import



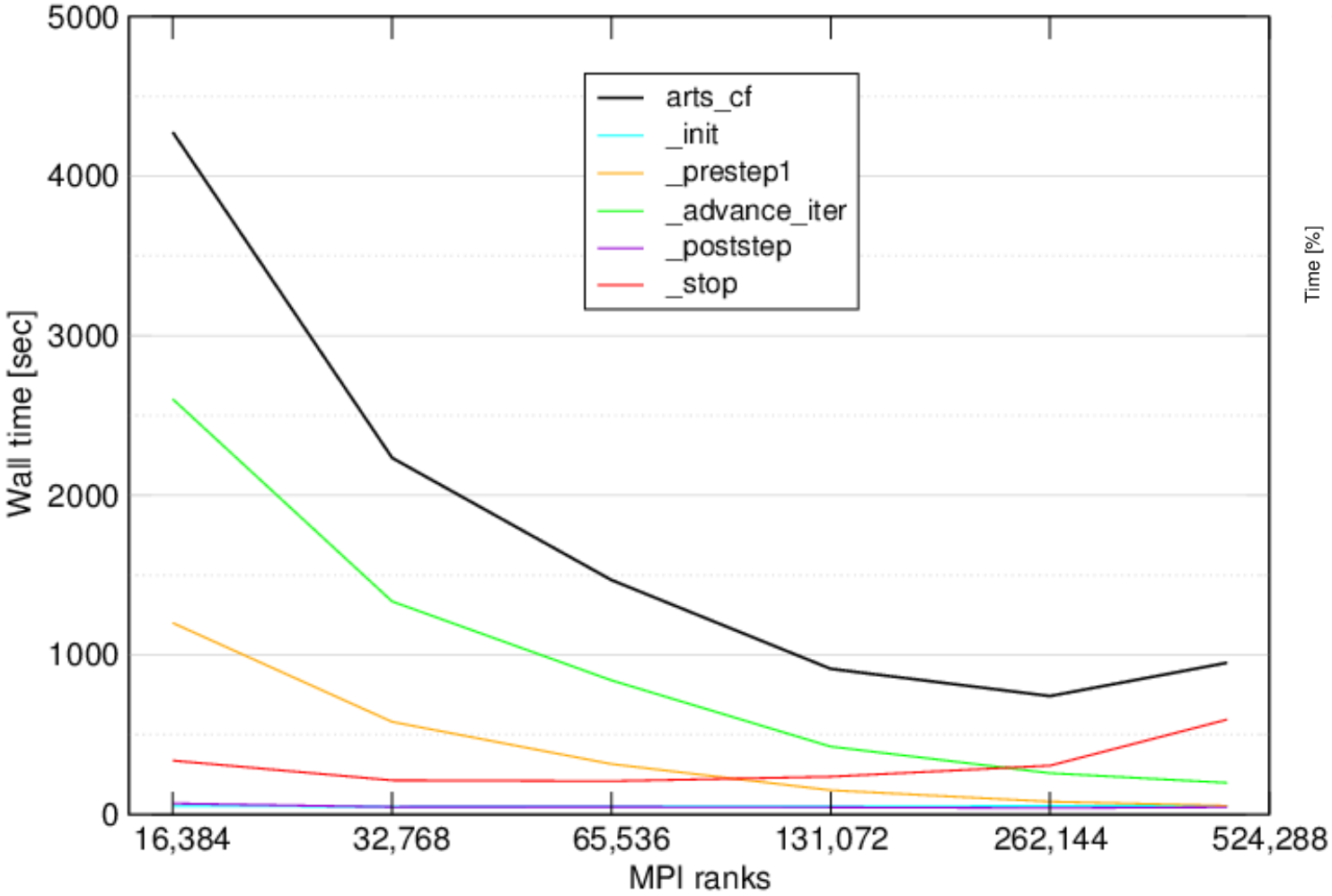
`import_synapses` extract of Scalasca/Score-P runtime summary profile of execution with 28 672 MPI ranks each with 16 OpenMP threads (28 racks) showing 21% of time in **`MPI_File_read_at`** when loading input data

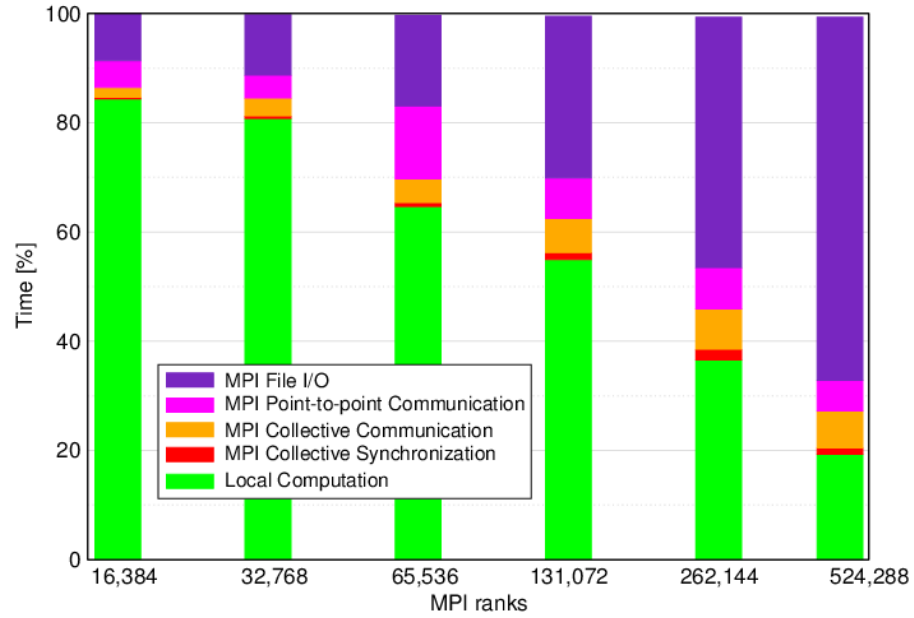
LLview monitoring of JUQUEEN during XSW

CIAO execution on 24 racks with 393 216 cores

Charts of recent power usage, job-size mix, file-system use and per-job file I/O bandwidth

CIAO scalability on JUQUEEN (1024³ grid, 100 iterations, 10 scalars)





- Excellent scaling of solver [_advance_iter]
- File writing dumping output [_stop] is bottleneck at scale

Parallel performance audit of CIAO on JUQUEEN

CIAO execution scalability breakdown at large scale

- while solver scales well, dump of 9 TiB simulation output does not
- inefficient MPI-I/O to single shared file

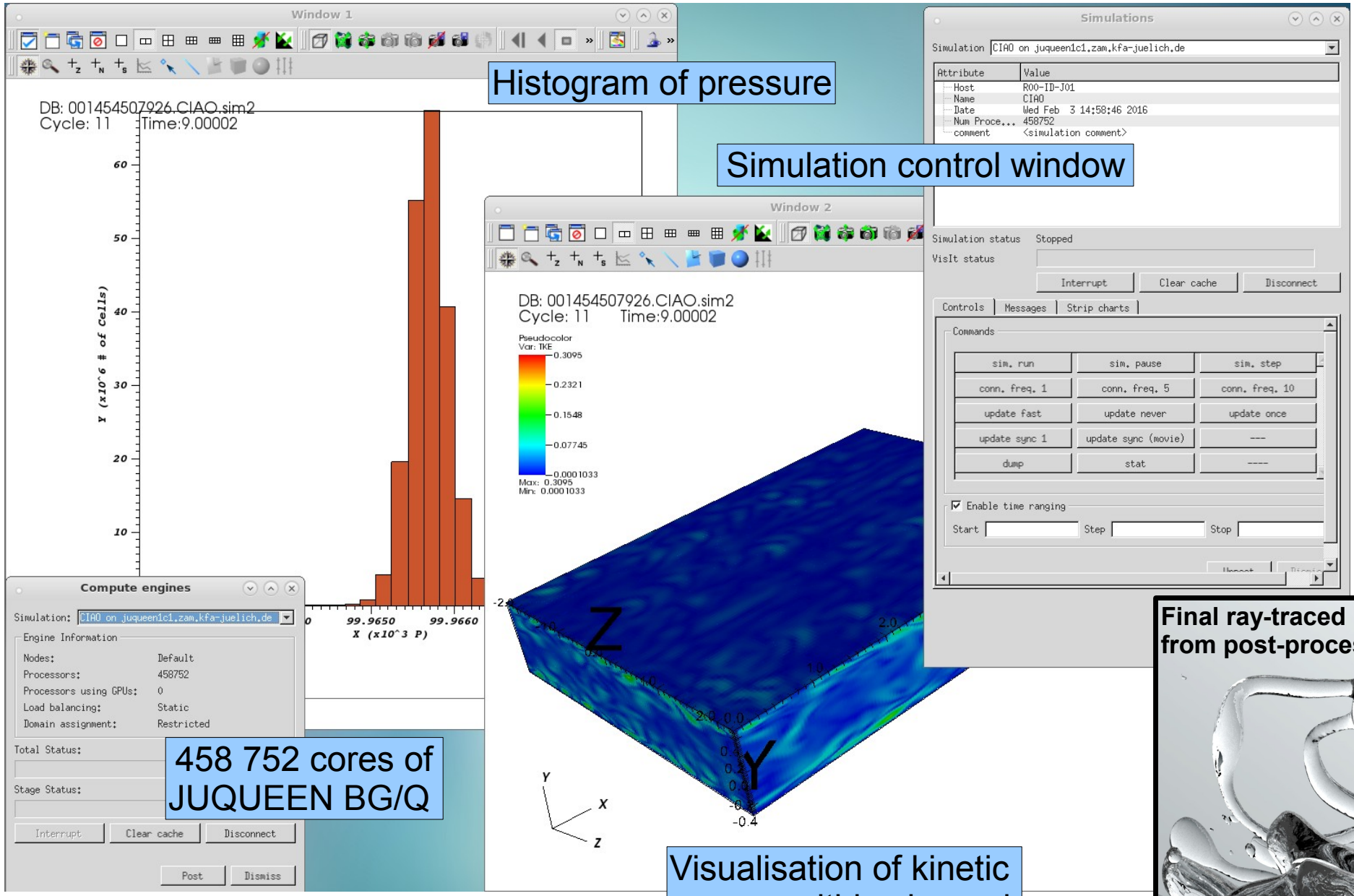
Investigated via Scalasca/Score-P measurements as part of performance audit offered by POP

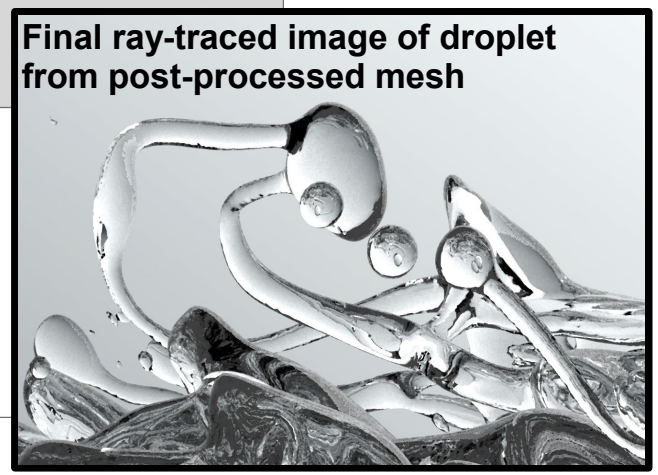
- *Performance Optimisation & Productivity* Centre of Excellence in Computing Applications
- http://www.pop-coe.eu/

Frequent dumping of simulation output (every 100 time steps) can be avoided with appropriate in-situ visualisation

- JUSITU developed to couple applications to VisIt visualisation client
- https://trac.version.fz-juelich.de/vis/wiki/VisIt

VisIt client visualisation coupled with JUSITU to CIAO on JUQUEEN





JÜLICH FORSCHUNGSZENTRUM

Performance Optimisation and Productivity CoE – http://www.pop-coe.eu

EU Horizon 2020 Centre of Excellence in Computing Applications [Oct 2015 – Mar 2018]

- transversal across application areas, academia/industry, platforms, scales
- BSC (coordinator), HLRS, JSC, NAG, RWTH, TERATEC

Provides three levels of free services

- ? – Parallel Application Performance Audit [1 month effort]
  - *identifies performance issues of client's code (on their computer system)*
- ! – Parallel Application Performance Plan [3 month effort]
  - *identifies root causes of issues, and quantifying approaches to address them*
- ✓ – Proof-of-Concept [6 month effort]
  - *experiments and mock-up tests to show effect of proposed optimisations*

Training in conjunction with VI-HPS

VI-HPS

JÜLICH FORSCHUNGSZENTRUM

Virtual Institute – High Productivity Supercomputing: http://www.vi-hps.org

Voluntary association of 12 international institutions developing tools for HPC [established 2006]

- primarily performance analysis, debugging and correctness checking

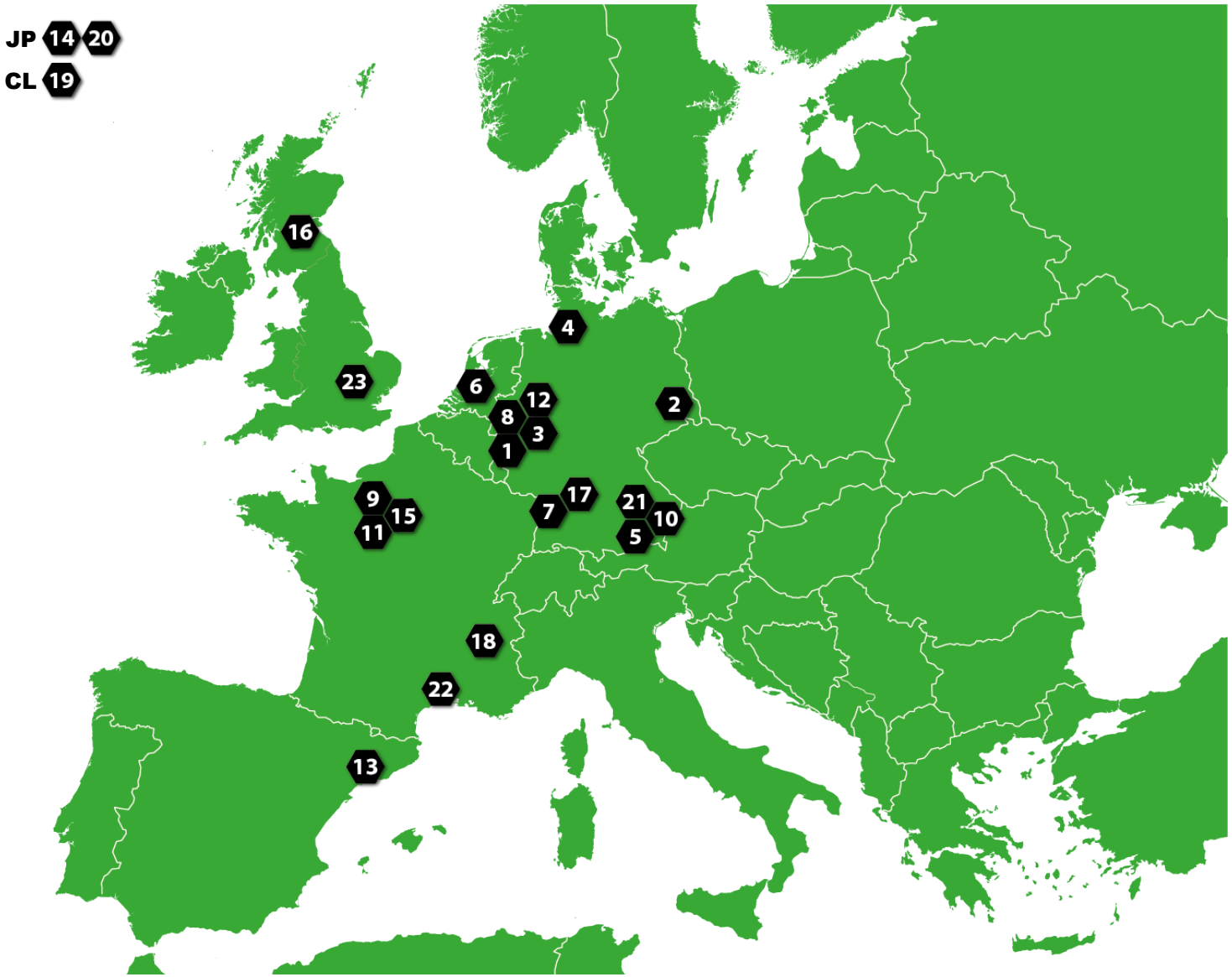
VI-HPS Tools Guide

- provides brief overview of tools capabilities and language/paradigm/system support

VI-HPS Tuning Workshops

- 3 – 5 day events focussed on coaching participants with their own application codes
- within Europe offered within PRACE Advanced Training Centre curriculum: FREE!
- next events hosted by CINES [Montpellier/France] and University of Cambridge [England] and then possibly in Berkeley/Livermore [California/USA]
- potential hosts for future training events are encouraged to get in contact

Check the VI-HPS website for latest information, or sign-up to the mailing list for news



www.vi-hps.org

PRACE

1.	2008/03/05+3	RWTH, Aachen, Germany
2.	2008/10/08+3	ZIH, Dresden, Germany
3.	2009/02/16+5	JSC, Jülich, Germany
4.	2009/09/09+3	HLRN, Bremen, Germany
5.	2010/03/08+3	TUM, Garching, Germany
6.	2010/05/26+3	SARA, Amsterdam, Netherlands
7.	2011/03/28+3	HLRS, Stuttgart, Germany
8.	2011/09/05+5	GRS, Aachen, Germany
9.	2012/04/23+5	UVSQ, St-Quentin, France
10.	2012/10/16+4	LRZ, Garching, Germany
11.	2013/04/22+4	MdS, Saclay, France
12.	2013/10/07+5	JSC, Jülich, Germany
13.	2014/02/10+5	BSC, Barcelona, Spain
14.	2014/03/25+3	RIKEN AICS, Kobe, Japan
15.	2014/04/07+4	MdS, Saclay, France
16.	2014/04/29+3	EPCC, Edinburgh, Scotland
17.	2015/02/23+5	HLRS, Stuttgart, Germany
18.	2015/05/18+5	UGA, Grenoble, France
19.	2015/10/27+3	NLHPC, Santiago, Chile
20.	2016/02/24+3	RIKEN AICS, Kobe, Japan
21.	2016/04/18+5	LRZ, Garching, Germany
22.	2016/05/23+5	CINES, Montpellier, France
23.	2016/07/06+3	Univ. of Cambridge, England

Lessons for exascale

Wide range of HPC applications have demonstrated excellent scalability on JUQUEEN BG/Q, generally with only modest tuning effort

- over-subscription of cores delivers important efficiency benefits
  - *use vectorisation/SIMDization & libraries for node performance*
- standard languages and MPI+multi-threading are sufficient
  - *MPI-only also possible but only 256MB available per rank*
  - *MPI communicator management gets increasingly costly*
- file I/O remains the most common impediment to scalability
  - *effective solutions need to be employed, such as SIONlib*
- scalable performance tools such as *Scalasca* help locate bottlenecks and identify opportunities for comm/synch optimisation

Scaling on BG/Q also delivers benefits for other HPC computer systems!

Further information

2016 Extreme Scaling Workshop technical report FZJ-JSC-IB-2016-01

- **http://juser.fz-juelich.de/record/283461**

High-Q Club webpages

- **http://www.fz-juelich.de/ias/jsc/high-q-club**

Proceedings of ParCo minisymposium (1-4 Sep 2015, Edinburgh)

- *Multi-system application extreme-scaling imperative*
- **http://www.fz-juelich.de/ias/jsc/MAXI**

Presentations of workshop at ISC-HPC (16 July 2015, Frankfurt)

- *Application extreme-scaling experience of leading supercomputing centres*
- **http://www.fz-juelich.de/ias/jsc/aXXLs**

The High-Q-Team

Dirk **Brömmel**
d.broemmel
@fz-juelich.de
Simulation Laboratory:
Plasma Physics

Wolfgang **Frings**
w.frings
@fz-juelich.de
Cross-Sectional Team:
Application Optimization

Brian **Wylie**
b.wylie
@fz-juelich.de
Cross-Sectional Team:
Performance Analysis

XSW16 application code-teams

Selected participants for 2016 JUQUEEN Extreme Scaling Workshop (1-3 Feb 2016)

- **CIAO**: Mathis Bode & Abhishek Deshmukh (RWTH-ITV/D)
- **Code_Saturne**: Charles Moulinec (STFC Daresbury/UK)
- **ICI-mesh**: Hugues Digonnet (ECNantes/F)
- **iFETI:** Axel Klawonn & Martin Lanser (UKöln/D) & Oliver Rheinbach (TUB Freiberg/D)
- **NEST-import**: Till Schumann & Fabien Delalondre (EPFL BBP/CH)
- **p4est**: Johannes Holke (UBonn/D)
- **PFLOTRAN**: Hedieh Ebrahimi (Amphos^{21}/E) & Guido Deissmann (FZJ-IEK6/D)
- **SLH**: Philip Edelmann (Heidelberger ITS/D)