Bridging the Gap: Parallel Storage, Monitoring Tools, and the Quest for Reproducibility?

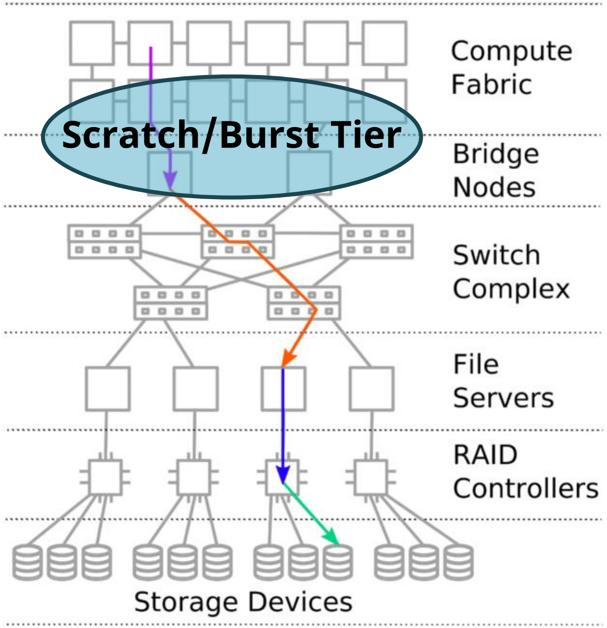
Sarah M. Neuwirth
Johannes Gutenberg University Mainz, Germany
neuwirth@uni-mainz.de

SOS26 Workshop, Cocoa Beach, FL, March 2024

Motivation

Heterogeneous and Complex HPC Infrastructures

- HPC infrastructure *too complex*, humans are *overwhelmed*
- Complexity and scope increase the *urgency*
  - *New computational paradigms* (AI/ML apps vs. BSP-style HPC)
  - *New architectural directions* (e.g., IPU, RISC-V, data flow)
  - *Heterogeneity overall*: node architectures, within the system, storage and parallel file system during application design (e.g., ML within HPC applications)
  - *New operations paradigms* (e.g., cloud, container)
  - Simplistic approaches to increasing compute demand result in *unacceptable power costs*
- Difficult for humans to optimally adapt applications to systems and to detect and diagnose vulnerabilities



Carns, P., 2023. *HPC Storage: Adapting to Change*. Keynote at REX-IO’23 Workshop.

Ciorba, F., 2023. *Revolutionizing HPC Operations and Research*. Keynote at HPCMASPA’23 Workshop.

B. Settlemyer, G. Amvrosiadis, P. Carns and R. Ross, 2021. *It’s Time to Talk About HPC Storage: Perspectives on the Past and Future*, in Computing in Science & Engineering, vol. 23, no. 6, pp. 63-68.

Motivation

Importance of Reproducibility in Scientific Research

Verification and Trust

- Allows other researchers to verify the original findings
- If findings cannot be reproduced, it casts doubt on the entire study and its conclusions

Building on Knowledge

- Science is a collaborative effort. By reproducing research, scientists can build upon existing knowledge. They can confirm findings, explore them further, or even identify inconsistencies that lead to new discoveries.

Reducing Errors

- Not all irreproducible research is due to misconduct. Mistakes happen.
- Reproducibility helps identify these errors in methodology, data collection, or analysis

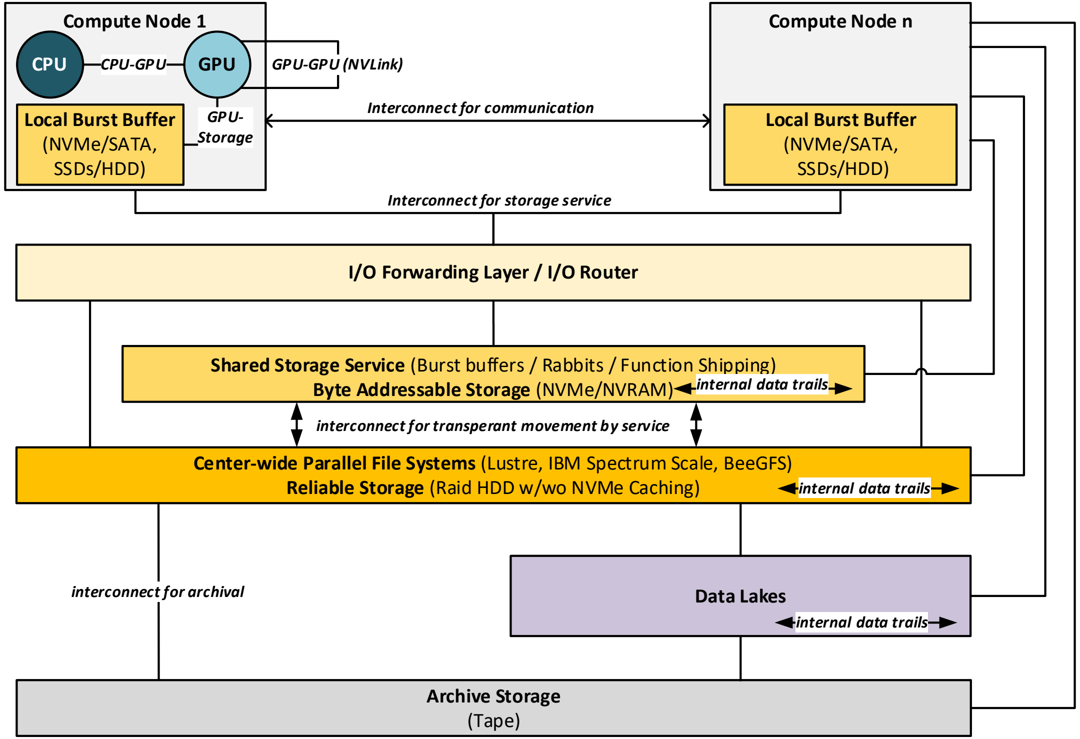
Transparency and Openness

- Reproducibility enables transparency in research. When researchers make their data and methods openly available, it allows others to see how they reached their conclusions. This openness is essential for scientific integrity.

Parallel I/O and Storage

Parallel I/O and Storage

Tracking the Data Trail in HPC Systems



Compute Node 1

CPU —CPU-GPU— GPU

GPU-GPU (NVLink)

GPU-Storage

Local Burst Buffer (NVMe/SATA, SSDs/HDD)

Interconnect for communication

Compute Node n

Local Burst Buffer (NVMe/SATA, SSDs/HDD)

Interconnect for storage service

I/O Forwarding Layer / I/O Router

Shared Storage Service (Burst buffers / Rabbits / Function Shipping)
Byte Addressable Storage (NVMe/NVRAM) *internal data trails*

interconnect for transperant movement by service

Center-wide Parallel File Systems (Lustre, IBM Spectrum Scale, BeeGFS)
Reliable Storage (Raid HDD w/wo NVMe Caching) *internal data trails*

interconnect for archival

Data Lakes
internal data trails

Archive Storage
(Tape)

Joint work-in-progress with Hariharan Devarajan (LLNL) and Jay Lofstead (SNL)

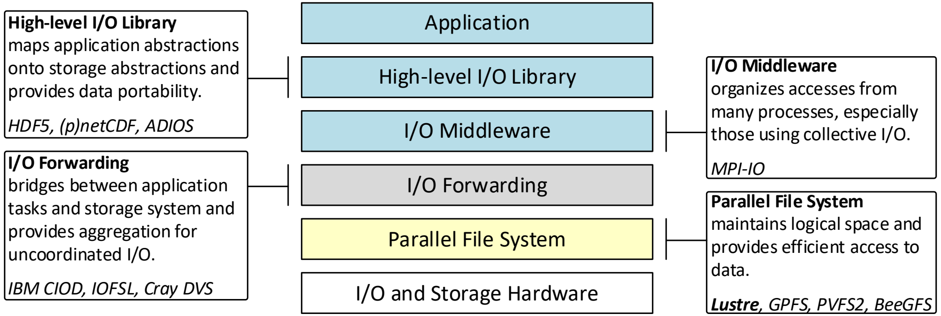
IO500

TOP500 The List.

Parallel I/O and Storage

Software Architectures for Parallel I/O

- Characterizing and understanding I/O behavior is critical => *increasingly complex I/O stack*
  - More diverse applications, computational frameworks, etc.
  - Emerging hardware and storage paradigms
- Understanding and re-envisioning I/O stands to benefit numerous HPC stakeholders:
  - Application scientists: Improved I/O performance ⇒ decreased time to scientific discovery
  - Admins: Inform decisions related to procuring new systems
  - Researchers: Optimizing storage system and I/O library designs



Application

High-level I/O Library

I/O Middleware

I/O Forwarding

Parallel File System

I/O and Storage Hardware

High-level I/O Library maps application abstractions onto storage abstractions and provides data portability.

HDF5, (p)netCDF, ADIOS

I/O Forwarding bridges between application tasks and storage system and provides aggregation for uncoordinated I/O.

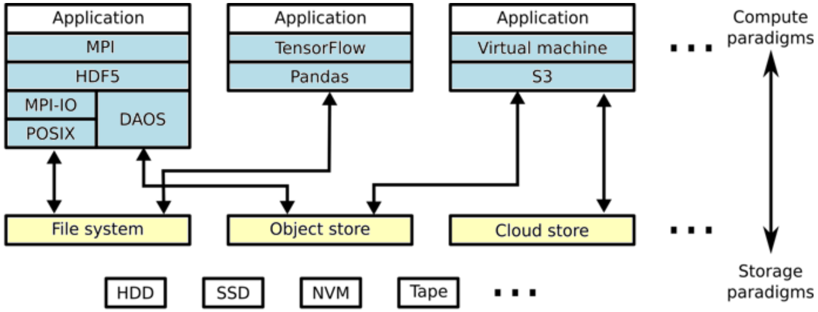
IBM CIOD, IOFSL, Cray DVS

I/O Middleware organizes accesses from many processes, especially those using collective I/O.

MPI-IO

Parallel File System maintains logical space and provides efficient access to data.

***Lustre**, GPFS, PVFS2, BeeGFS*

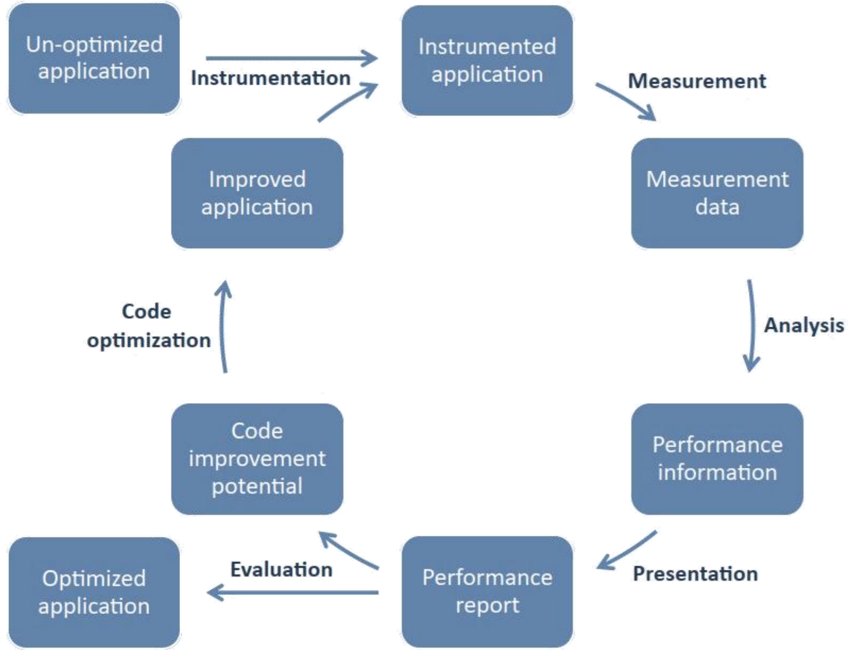


Monitoring Tools and I/O

Monitoring Tools and I/O
Performance Optimization Cycle

Performance engineering typically is a cyclic process:

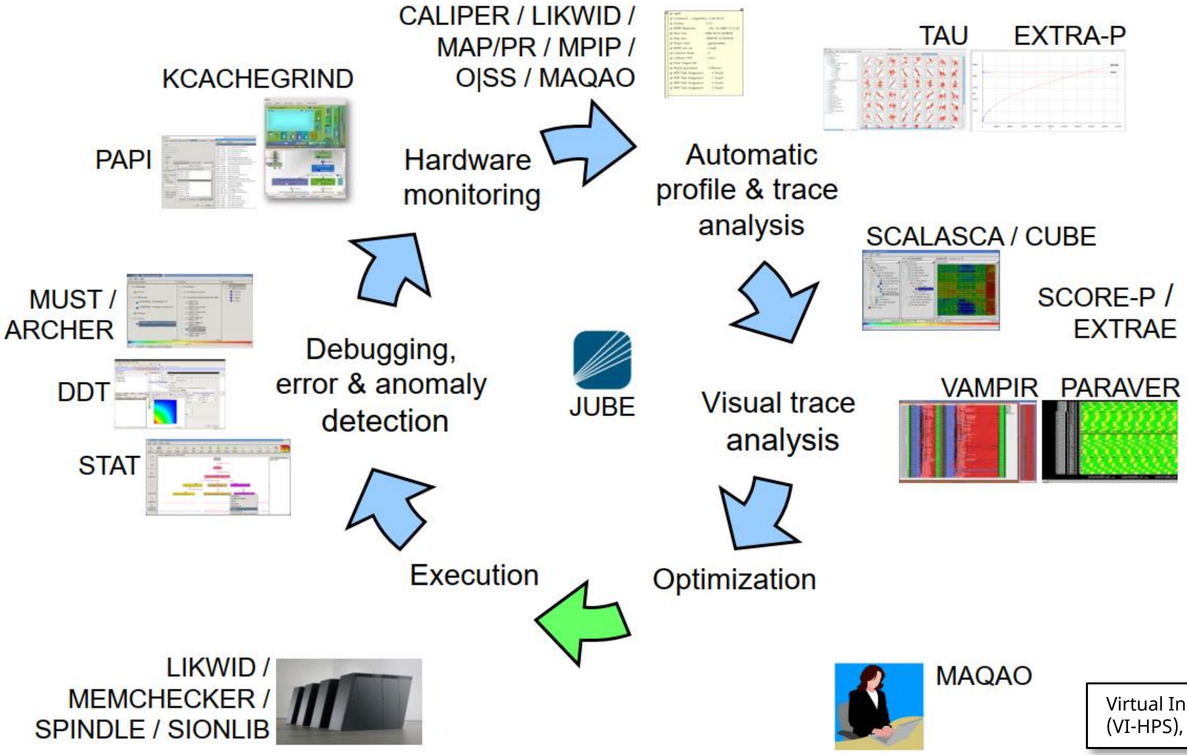
- ***Instrumentation:*** common term for preparing the performance measurement
- ***Measurement:*** During measurement, raw performance data is collected
  - **Profiles:** hold aggregated data (e.g. total time spent in function foo())
  - **Traces:** consist of a sorted list of timed application events/samples (e.g. enter function foo() at 0.11 s)
- ***Analysis:*** Well defined performance metrics are derived from raw performance data during analysis
- ***Presentation:*** Presenting performance metrics graphically fosters human intuition
- ***Evaluation (and Code Optimization):*** Requires tools and lots of thinking



Performance Engineering Overview, https://doc.zih.tu-dresden.de/software/performance_engineering_overview/

Monitoring Tools and I/O

Overview of the VI-HPS Tools



Hardware monitoring

KCACHEGRIND

PAPI

CALIPER / LIKWID / MAP/PR / MPIP / O|SS / MAQAO

Automatic profile & trace analysis

TAU

EXTRA-P

SCALASCA / CUBE

SCORE-P / EXTRAE

Visual trace analysis

VAMPIR

PARAVER

Optimization

MAQAO

Execution

LIKWID / MEMCHECKER / SPINDLE / SIONLIB

Debugging, error & anomaly detection

MUST / ARCHER

DDT

STAT

JUBE

Virtual Institute – High Productivity Supercomputing (VI-HPS), https://www.vi-hps.org/tools/tools.html

Monitoring Tools and I/O

I/O Performance Factors and Metrics

Factors Potentially Affecting Reproducibility of I/O Performance:

Application

- Number of processes
- Request sizes
- Access patterns
- I/O operation
- Data volume

Network

- Message sizes
- Network topology
- Network paths
- Network type

File System

- Type of file system
- Disk types
- Stripe sizes
- File hierarchy
- Shared access

Multiple Tools for I/O Performance Analysis:

- May be a problem when users need to change the tool and want to ensure the measurement continuity and comparability
- There is no easy way to verify metrics consistency between tools

=> *Mango-IO first attempt to provides tools-agnostic metrics calculation*

Liem, Radita, Sebastian Oeste, Jay Lofstead, and Julian Kunkel. *Mango-IO: I/O Metrics Consistency Analysis*. In 2023 IEEE International Conference on Cluster Computing Workshops (CLUSTER Workshops), pp. 18-24. IEEE, 2023.

Quest for Reproducibility

Quest for Reproducibility

Understanding the Terminology

Reproducibility

- ***Definition (Oxford Dictionary):*** The extent to which measurements made under one set of conditions (or by one observer) can be repeated under different conditions (or by another observer).
- ***In computational sciences***[1,2]***:*** The results should be documented by making all data and code available in such a way that the computations can be executed again with identical results.

Replicability

- ***Definition (Cambridge Dictionary) – replicable:*** that can be done in exactly the same way as before, or produced again to be exactly the same as before
- ***In science:*** The ability of a study's findings to be reproduced by independent researchers using the same question but potentially new data or methods.

Repeatability

- ***Definition***[3]***:*** Closeness of the agreement between the results of successive measurements of the same measure, when carried out under the same conditions
- ***Conditions that need to be met***[4] ***:*** the same experimental tools, the same observer, the same measuring instrument, used under the same conditions, the same location, repetition over a short period of time.

Interpretability

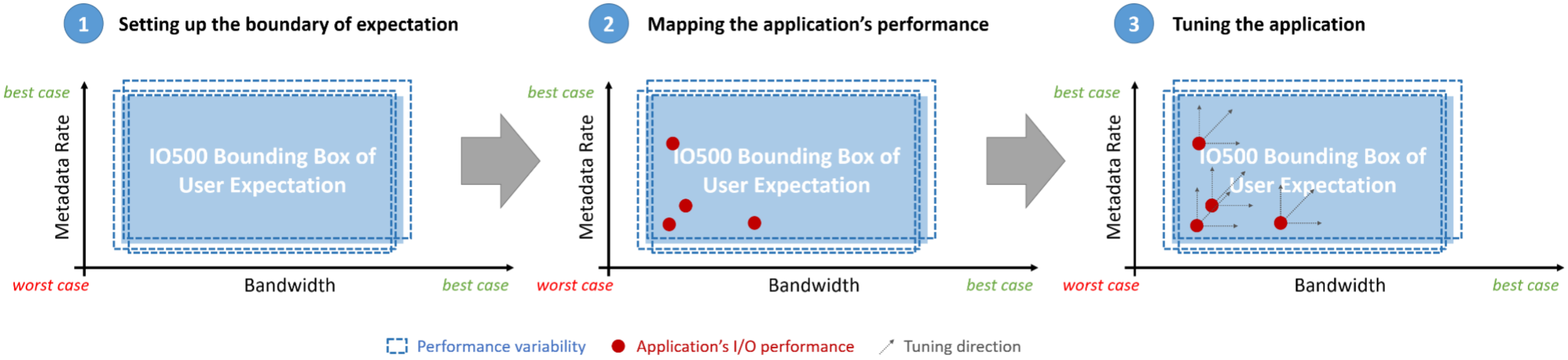
- ***Machine Learning***[5]***:*** Interpretable ML is a useful umbrella term that captures the "extraction of relevant knowledge from a machine-learning model concerning relationships either contained in data or learned by the model".
- ***Thorsten Hoefler***[6]***:*** An experiment is interpretable if it provides enough information to allow scientists to understand the experiment, draw own conclusions, assess their certainty, and possibly generalize results

Quest for Reproducibility

Example: Reproducibility using the IO500 Benchmark

General idea:

Use IO500 benchmark's mdtest and IOR to form a *bounding box of user expectations*



1 Setting up the boundary of expectation
2 Mapping the application's performance
3 Tuning the application
best case
Metadata Rate
IO500 Bounding Box of User Expectation
worst case
Bandwidth
best case
Performance variability
Application's I/O performance
Tuning direction

Worst case scenario is from IOR and mdtest 'hard' scenario

Best case scenario is from IOR and mdtest 'easy' scenario

Liem, Radita, et al. "*User-centric system fault identification using IO500 benchmark*." 2021 IEEE/ACM Sixth International Parallel Data Systems Workshop (PDSW). IEEE, 2021.

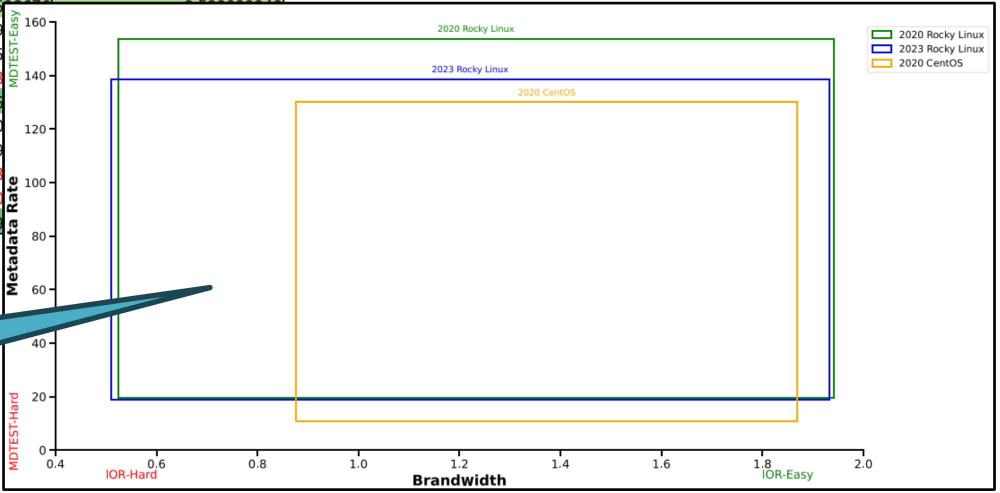
Quest for Reproducibility

Example: Reproducibility using the IO500 Benchmark

Same system: CLAIX18 – 4 nodes BeeOND with the same config but different OS

	ISC 2020 benchmark	ISC 2023 benchmark		ISC 2020 benchmark
	CentOS - mpiexec	Rocky Linux - mpiexec	Rocky Linux - srun	Rocky Linux - srun
find	1468.114386	75.1189045	361.4813317	560.52
ior-easy-read	2.019125	1.598536625	2.1384143	2.14
ior-easy-write	1.731133778	1.438956	1.7465284	1.761
IOR-EASY	1.869592332	1.516648894	1.932563403	1.941272778
ior-hard-read	1.409629778	0.16210825	0.3387456	0.341
ior-hard-write	0.544298222	0.1616045	0.7664143	0.805
IOR-HARD	0.875933206	0.161856179	0.5095	[illegible]
mdtest-easy-delete	99.55603122	7.256035125	86.9	
mdtest-easy-stat	332.9714572	14.05152075	312.35	
mdtest-easy-write	66.35817467	3.66257475	98.18	
MDTEST-EASY	130.0537875	7.201170065	138.6	
mdtest-hard-delete	9.040069222	0.695508	8.80	
mdtest-hard-read	22.66211533	2.2294245	20.99	
mdtest-hard-stat	26.73728556	18.01108238	89.18	
mdtest-hard-write	2.601157333	1.7526735	7.90	
MDTEST-HARD	10.92544247	2.645050286	18.9	

Liem, Radita. "*I/O Performance Reproducibility using IO500 Benchmark*." EOFS Workshop @ TU Dresden, 2024.



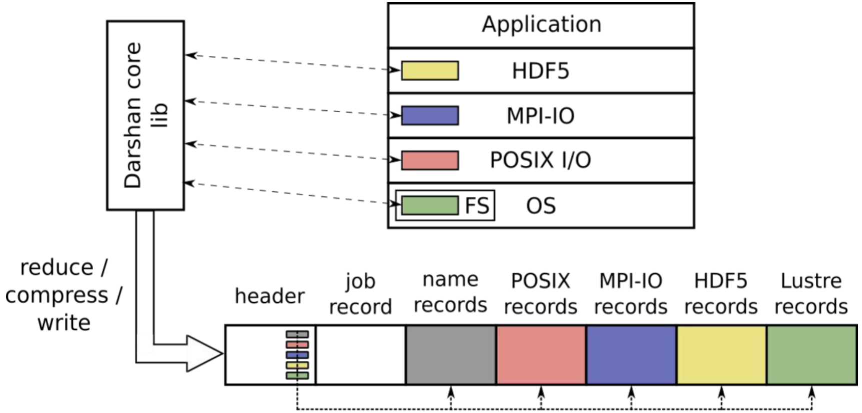
Impact of the OS change or just hardware degradation?

Quest for Reproducibility

Example: Darshan I/O Characterization Tool

- **Blue Waters, Mira, and Theta popular Darshan log sources used for research:**
  - https://bluewaters.ncsa.illinois.edu/data-sets
  - https://reports.alcf.anl.gov/data/
  - ftp://ftp.mcs.anl.gov/pub/darshan/data
- **Some open questions:**
  - How relevant are the logs to current systems?
  - How do we know the integrity of the logs?
- **Community statements:**
  - Darshan is one of the first tools to be deactivated in the event of I/O problems.
  - Darshan cannot grasp the complexity of state-of-the-art parallel storage systems.

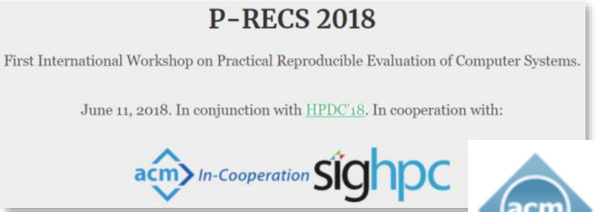
Darshan I/O Characterization Tool



Snyder, S., 2022. *Darshan: Enabling Insights into HPC I/O Behavior*. ECP Community BoF Days.

What are the implications of these questions and observations?

Quest for Reproducibility
Community Efforts



P-RECS 2018

First International Workshop on Practical Reproducible Evaluation of Computer Systems.

June 11, 2018. In conjunction with HPDC'18. In cooperation with:

ACM REP

The International Conference for High Performance Computing, Networking, Storage, and Analysis

INTERNATIONAL CONFERENCE ON PARALLEL PROCESSING

NDSS

Artifact Review and Badging:
https://www.acm.org/publications/policies/artifact-review-and-badging-current

How well are parallel storage and I/O being considered?

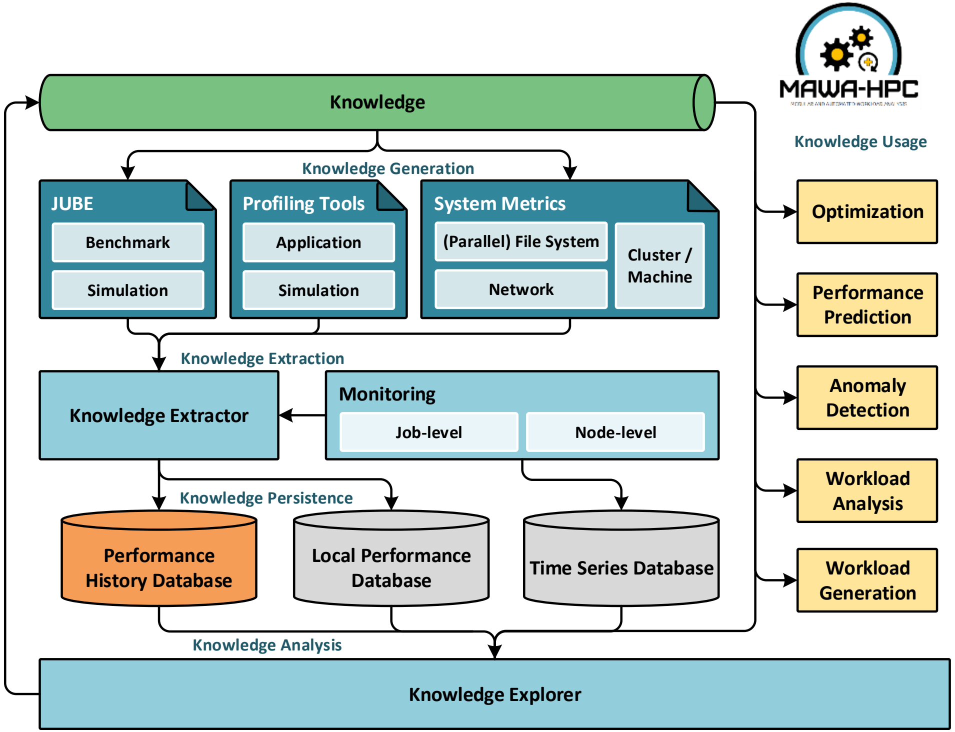
HPCIO ANALYSIS

HPC IO Analysis –I/O Trace Initiative:
https://hpcioanalysis.zdv.uni-mainz.de/

Quest for Reproducibility

Holistic Performance Engineering and Analysis

- ***Idea:*** Design and implement standardized and tool-independent approach for HPC workload and application analysis
- Support and integration of various community tools, increasing the compatibility and coverage of different use cases
- Intuitive performance modeling and visualization so that users without prior knowledge can understand the results
- ***Goal:*** Establish a *performance history database* to categorize systems, workload behaviors, and characteristic patterns for different science domains



Quest for Reproducibility

Reproducible Evaluation – Design Discussion

Repeatability

Runs should be able to be repeated with little configuration effort

Schifrin, A., 2023. *Automated Performance Characterization of HPC Systems.* Bachelor thesis, Goethe University Frankfurt.

Comparability

A simplified structure amplifies comparability of results between runs or even machines

Bartelheimer, N. and Neuwirth, S., 2023. *Toward Reproducible Benchmarking of PGAS and MPI Communication Schemes.* ICPADS'23.

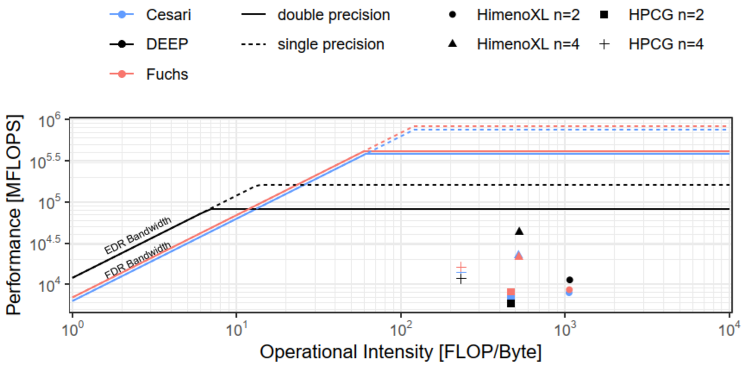
Modularity

Enhanced modularity increases possiblities for further extensions

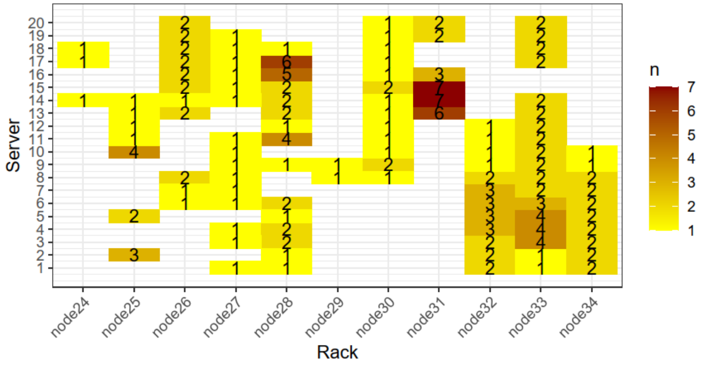
Bartelheimer, N., Zhu, Z., and Neuwirth, S., 2024. *Automated Network Performance Characterization for HPC Systems*. International Journal of Networking and Computing.

Quest for Reproducibility

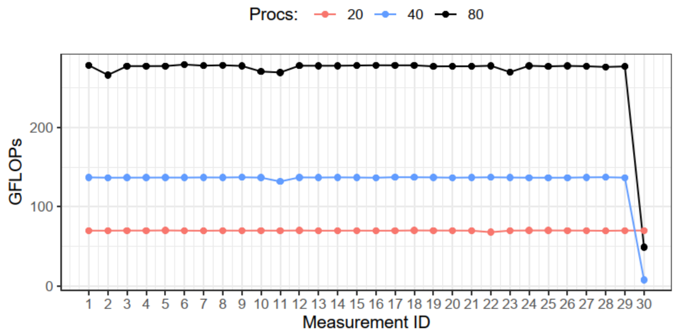
Reproducible Evaluation – Example Results



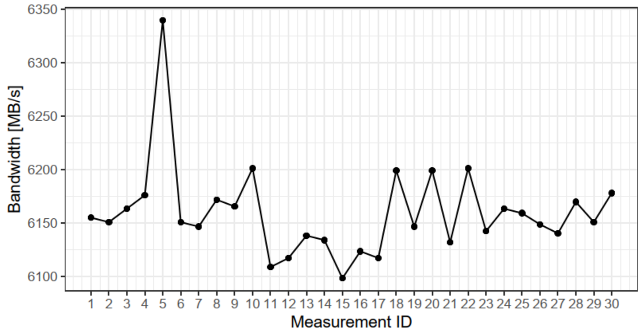
Roofline model with Himeno and HPCG results.



Heat map of the allocated nodes (overall benchmark runs).



Himeno benchmark over 15 days / 2 measurements per day.



RDMA point-to-point performance over 15 days / 2 measurements per day.

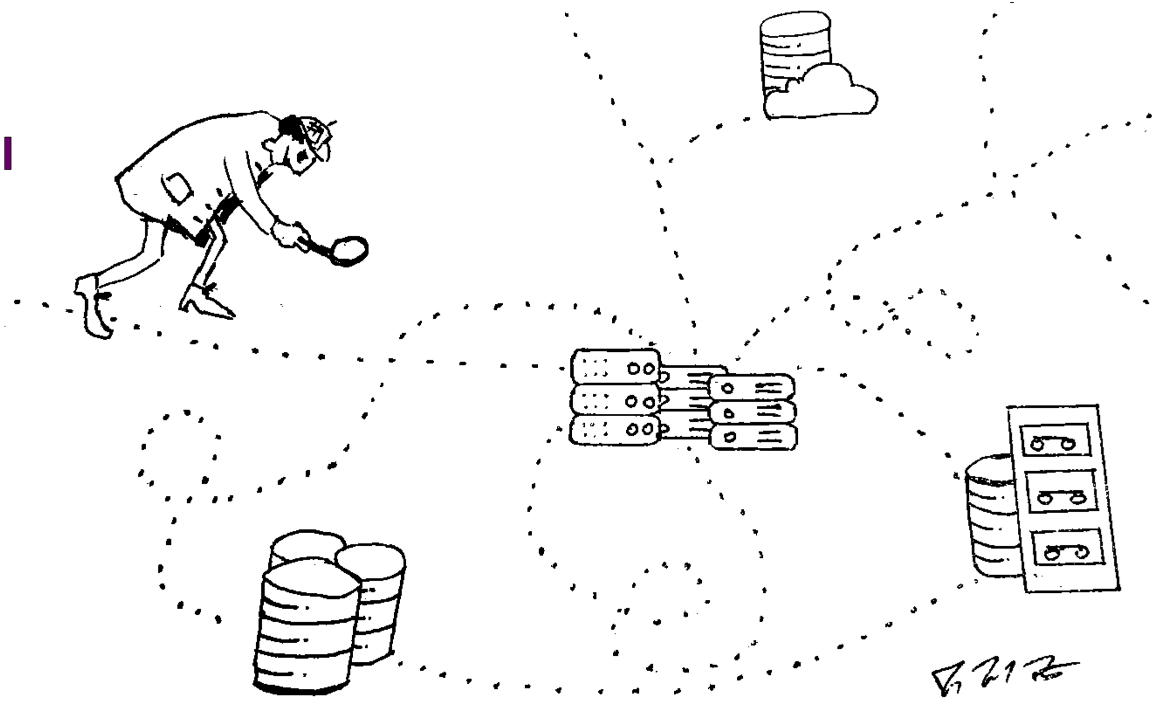
Quest for Reproducibility

Toward traceroute for Data Trails

How can the differences between modern monitoring infrastructures and the actual data trails be reconciled?

↓

What if there would be *traceroute* for parallel storage and I/O architectures?

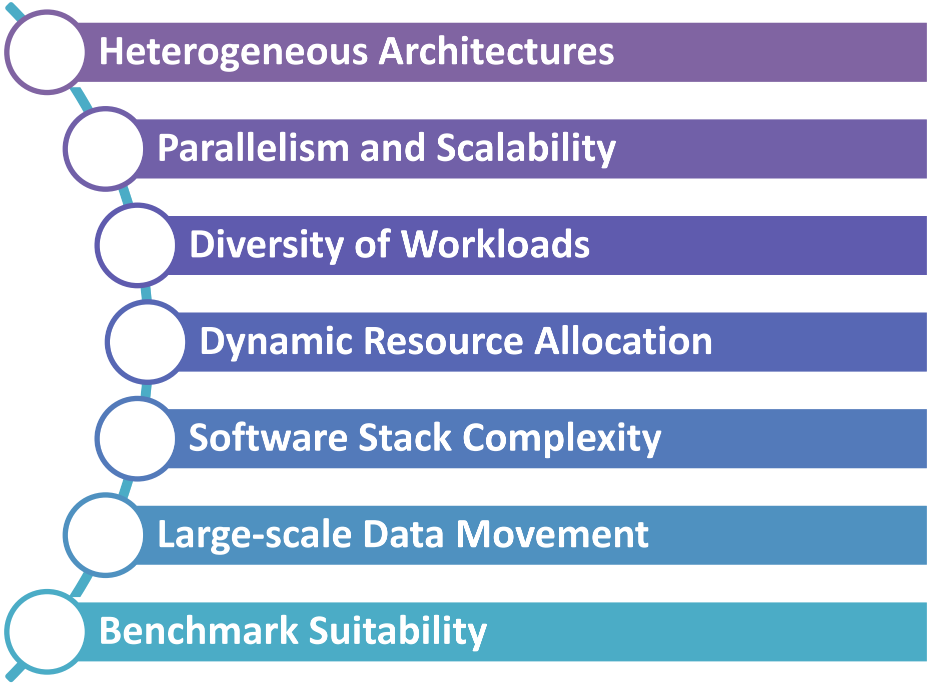


© Joey White-Swift

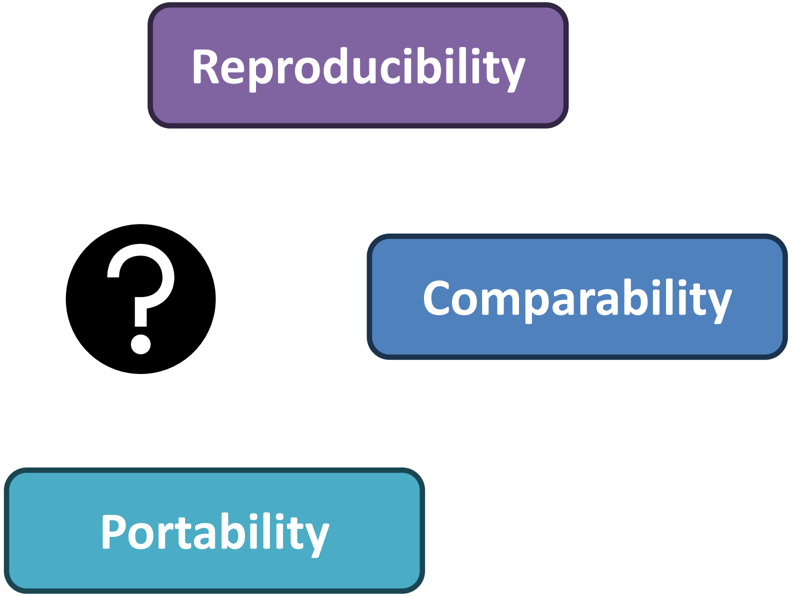
Conclusion

Conclusion

Reproducibility through Holistic Performance Engineering



- Heterogeneous Architectures
- Parallelism and Scalability
- Diversity of Workloads
- Dynamic Resource Allocation
- Software Stack Complexity
- Large-scale Data Movement
- Benchmark Suitability



Reproducibility

?

Comparability

Portability

Thank you for your Attention!