A Full Year I/O Request Size Analysis of HPC Applications on the Intrepid Supercomputer

VALÉRIA S. GIRELLI[1], JEAN LUCA BEZ[1], FRANCIELI Z. BOITO[2], PABLO J. PAVAN[1] AND PHILIPPE O. A. NAVAUX[1]

[1]Universidade Federal do Rio Grande do Sul - Instituto de Informática, Brazil
[2]Univ. Grenoble Alpes, Inria, CNRS, Grenoble INP, LIG, 38000 Grenoble, France

WSPPD 2018

INSTITUTO DE INFORMÁTICA UFRGS

UFRGS
UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Summary

- Motivation
- Goals
- Methodology
- Request Size Analysis
- Conclusions

Motivation

Historical gap between the **processing** speed and the **data access** speed

- Hundreds of computing nodes
- Concurrent access to the storage system

Applications with different I/O behaviors

Inefficient access patterns are a problem

- Bad programming choices
- Poor use of high level interfaces

Motivation

Inefficient access patterns

Small requests generally translate into **poor I/O performance**

- Small portions of data being transferred
- High cost to access the storage system

High level interfaces assist in the data access

With POSIX, the programmer is the only responsible for **managing I/O operations**

Goals

Understand the applications I/O behavior:

- An entire year of executions - large dataset
- Most common access sizes
- Both read and write operations
- Both POSIX and MPI-IO interfaces

Identifying inefficient access patterns

Methodology

Methodology

Darshan I/O Characterization Tool

- Profile of I/O operations at application level

Supercomputer Intrepid Blue Gene/P, Argonne

91,994 captured jobs during 2012

- Application coverage rate between 20% and 80%

Anonymized information

Methodology

Separate log for each job

Application information:

- Application identifier
- Job identifier
- User identifier
- Number of processes
- Runtime

Methodology

Darshan

Statistics for each opened file

Counters:

- Individual/Collective Access
- Interface (POSIX and MPI-IO)
- Access sizes
- Number of read and write operations
- Time spent in I/O operations

Request Size Analysis

Request Size Analysis

POSIX was responsible for:

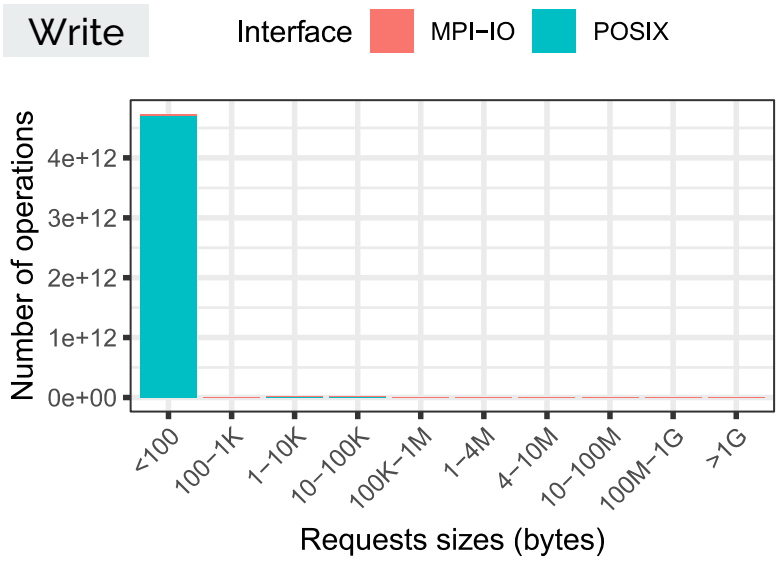
99.2% of write operations

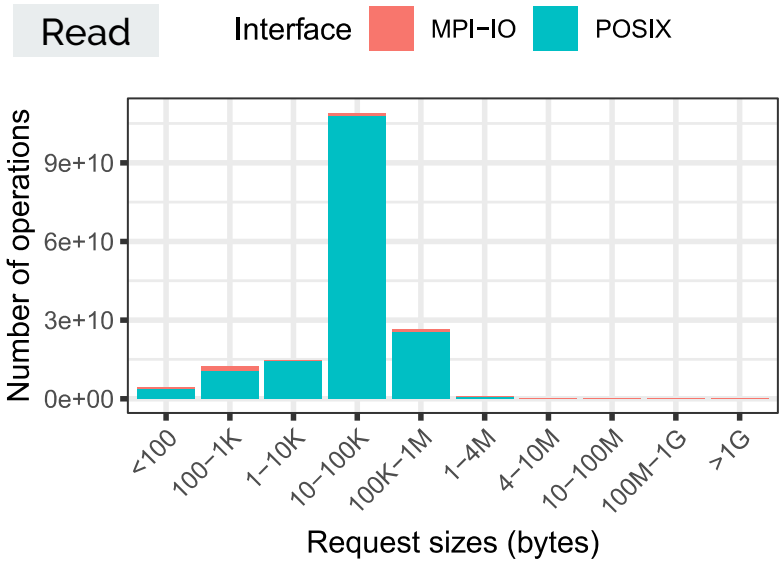
97.3% of read operations

Most common access sizes:

Write: up to 100 bytes - 98.7%

Read: 10KB and 100KB - 64.3%





Request Size Analysis

Most common access sizes:

Write: up to 100 bytes - 53.6%

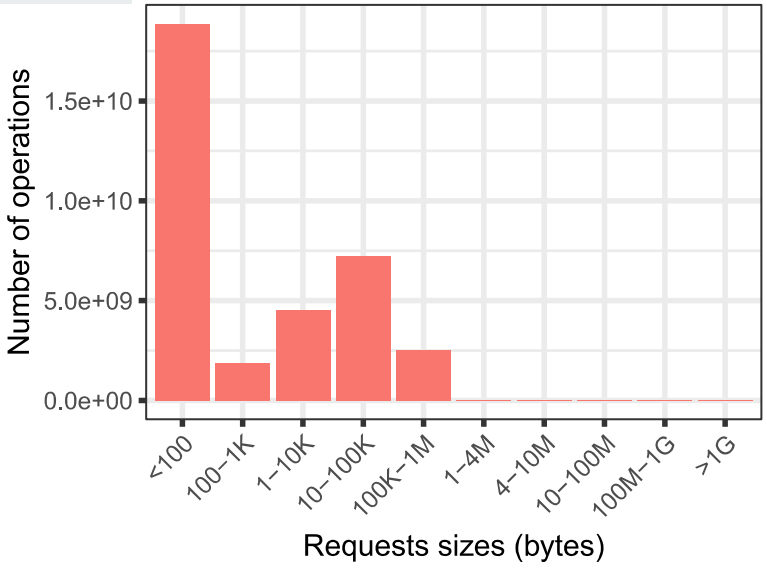
Read: 100 bytes and 1KB - 34.2%

10KB and 100KB - 23.7%

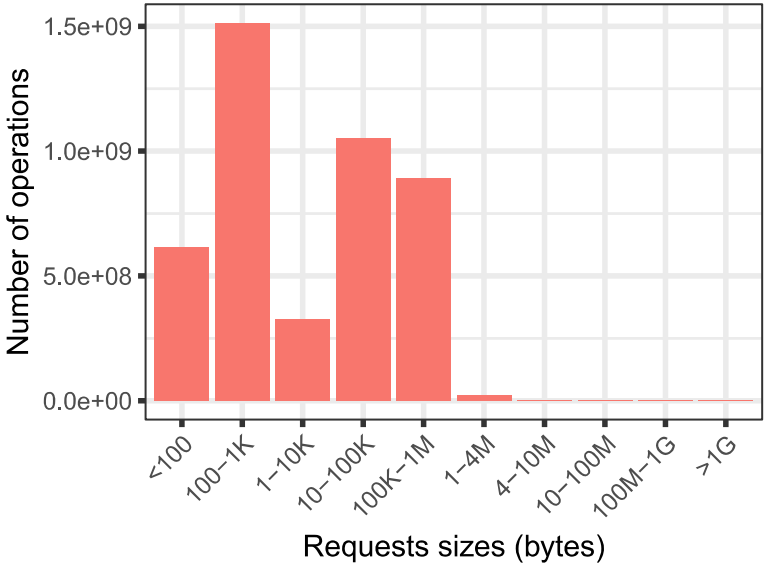
Spread distribution:

- complex datatypes with MPI-IO
- small and sparse requests aggregated into a large one

Write



Read



Request Size Analysis

Is this behavior the result of a small group of applications?

27.495 different applications

Some applications had the same number of read operations

- Same input data
- Similar I/O behavior
- **2,019 different applications**

Request Size Analysis

Analysis of the **10 more I/O intensive** applications on **number of operations**

Low impact using MPI-IO:

- 15,360 read operations
- 4,623 write operations

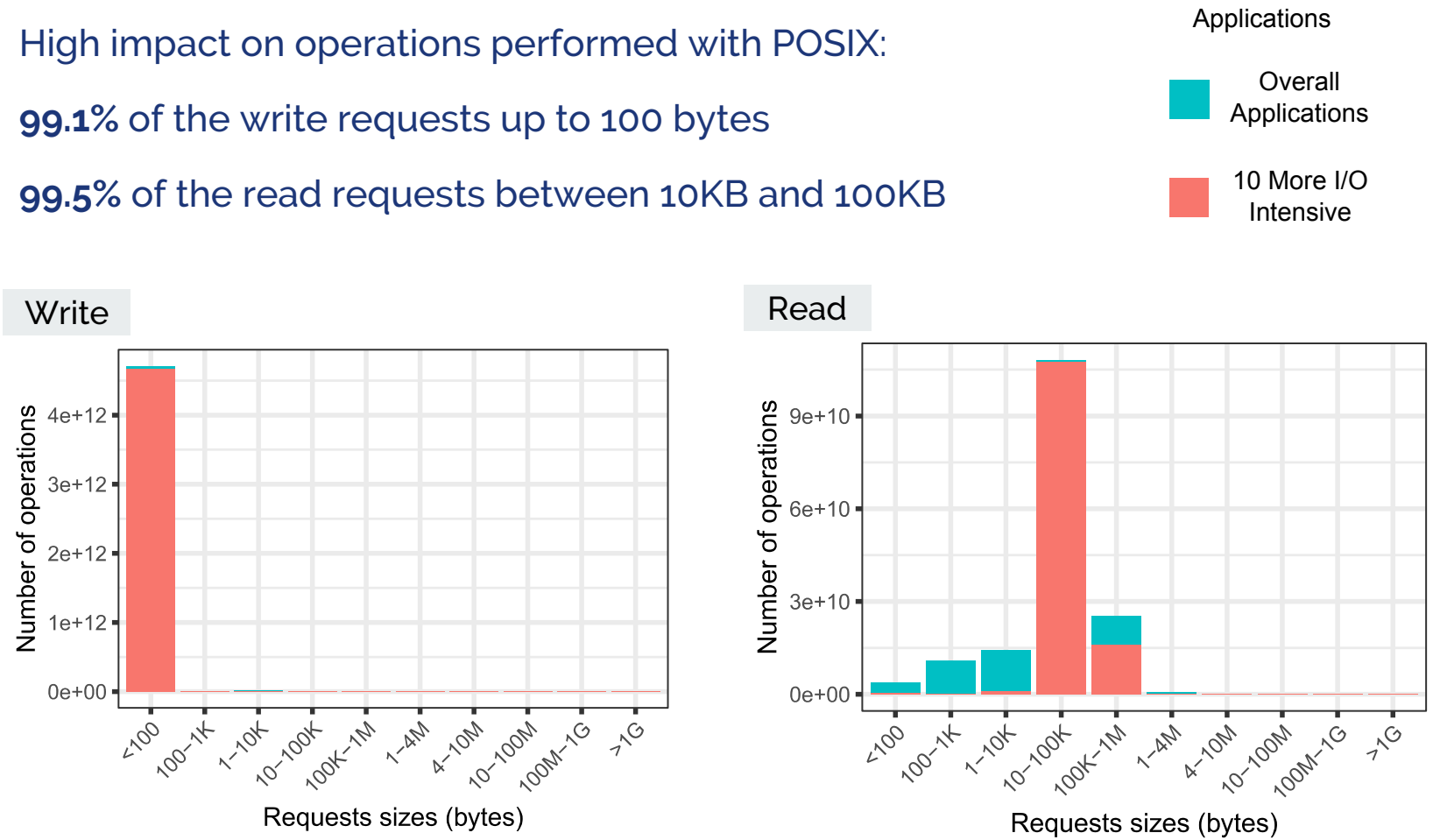
Request Size Analysis

I/O Intensive applications

High impact on operations performed with POSIX:

99.1% of the write requests up to 100 bytes

99.5% of the read requests between 10KB and 100KB



Request Size Analysis

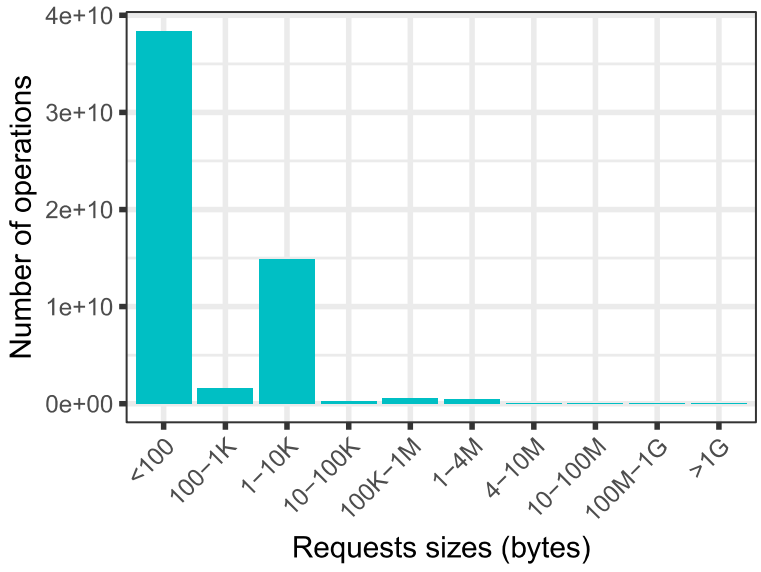
I/O Intensive applications

Without the 10 more I/O intensive applications:

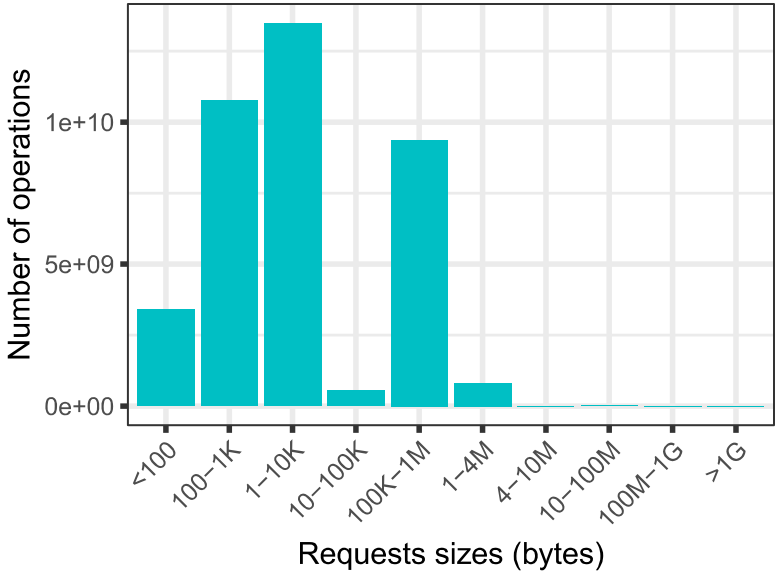
New most common read access size: between 1kB and 10KB - 35.1%

Still a large number of I/O operations

Write



Read



Conclusions

Conclusions

A considerable amount of analyzed data

POSIX is still surprisingly used

Applications are not taking advantage of optimizations provided by high level interfaces

Small write requests - less than 100 bytes

The 10 more I/O intensive applications affect the observed access sizes

Conclusions

How much data is being transferred by **each access size**

Which are the **access sizes** responsible for the **largest data amount**

How much system time is spent in the operations performed by **each access size**

Questions?

Thank you!

A Full Year I/O Request Size Analysis of HPC Applications on the Intrepid Supercomputer

VALÉRIA S. GIRELLI[1], JEAN LUCA BEZ[1], FRANCIELI Z. BOITO[2], PABLO J. PAVAN[1] AND PHILIPPE O. A. NAVAUX[1]

[1]Universidade Federal do Rio Grande do Sul - Instituto de Informática, Brazil
[2]Univ. Grenoble Alpes, Inria, CNRS, Grenoble INP, LIG, 38000 Grenoble, France

WSPPD 2018

.inf INSTITUTO DE INFORMÁTICA UFRGS

UFRGS UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL