The Effective I/O Bandwidth Benchmark (b_eff_io)

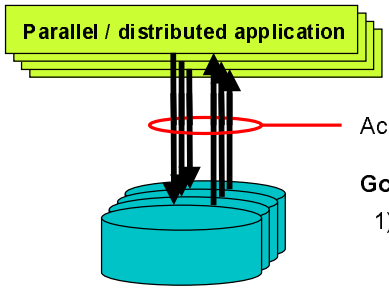
Rolf Rabenseifner
High-Performance Computing-Center Stuttgart (HLRS), University of Stuttgart,
rabenseifner@hlrs.de *www.hlrs.de/people/rabenseifner*
Alice E. Koniges
Lawrence Livermore National Laboratory,
koniges@llnl.gov *www.rzg.mpg.de/~ack*

Euro-Par 2000
Technische Universität München, Aug. 29 - Sep. 1, 2000
www.in.tum.de/europar2k/

Outline

- Our goals
- Starting-points
- Definition of the b_eff_io benchmark
- Results
- Future plans
- Summary

Effective I/O Bandwidth Benchmark



Accumulated I/O bandwidth ?

Goals:

1) detailed insight
   — bandwidth of several patterns, chunk sizes, ...
2) one characteristic value!
   — like Rmax in TOP500
3) comparing I/O of different systems

Starting-Points

- Application benchmarks
  - using real, I/O-intensive applications
- Filesystem benchmarks
  - measuring several parameters around the most friendly disk-usage-pattern
- Hardware benchmarks
  - maximum bandwidth of the disk — special-benchmark
- Why a new benchmark for parallel I/O?
  - application / filesystem / hardware independent
  - but, average on possible application scenarios
  - portable

  ==> MPI-I/O based benchmark

Starting-Points — the I/O Parameter Space

- How to define and measure one characteristic I/O bandwidth value?
- The I/O parameter space — 20 orthogonal parameters:
  - **Application parameters:**
    - (a) the size of contiguous chunks in the memory, (b) on disk, (c) ... (f)
  - **Usage aspects:**
    - (a) how many processes are used
    - (b) how many parallel processors and threads are used for each process.
  - **I/O interface:**
    - (a) Posix I/O buffered or (b) raw,
    - (c) special filesystem I/O of the vendor filesytem,
    - (d) MPI-I/O.
  - **MPI-I/O aspects:**
    - (a) access methods, i.e., first writing of a file, rewriting or reading, (b) ...
    - (c) coordination, i.e., collectively or noncollectively, (d) ... (f)
  - **Filesystem parameters:**
    - (a) which filesystem is used,
    - (b) how many nodes are used as I/O servers, (c) ... (f)

  (full list, see paper)

Starting-Points — the I/O Parameter Space

- Measuring 10 values in each orthogonal dimension
  —> 10^{20} exciting measurements
    —> your benchmark will run the next decade

- Using ***one basic, central parameter set***
  —> varying each single parameter
    —> only 10*20 measurements
      —> fine, you get your results in ~ 1 hour
        —> but, vendors *Xray* and *Ybm* do not like the ***one basic, central parameter set*** preferred by *Zun*, or ?–)

Interrupt — b_eff — the design base for b_eff_io

- b_eff — the effective communication bandwidth benchmark

  Authors: Karl Solchenbach, Hans-Joachim Plum and Gero Ritzenhoefer, Rolf Rabenseifner
- www.hlrs.de/mpi/b_eff
- One characteristic accumulated communication bandwidth number
  := average bandwidth on several communication patterns*)
  average on different message sizes
  maximum over different MPI programming techniques

  *) each process sends a message to another one, each pattern is defined as a set of processor-pairs
- needs a few minutes

Interrupt — Existing I/O Benchmarking Papers

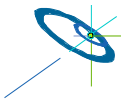
- Existing I/O benchmarking papers:

  e.g., Terry Jones, Alice Koniges, R. Kim Yates,
  Performance of the IBM General Parallel File System,
  to be published in Proceedings of the International Parallel and Distributed Processing Symposium, May 2000. Also available as UCRL JC135828
- many hours of dedicated benchmarking time is used
- characterizing a specific system
- not portable
- Rule: Balanced HPC systems should be able to write the total memory in 10 minutes to disk
  ==> An I/O benchmark should not need hours!
  — 10 minutes may be enough to overrun any cache!

Starting-Points

— back to b_eff_io —

- What can we learn from the communication benchmark b_eff?
- How can we reduce I/O benchmarking to 30 minutes?
- How can b_eff_io determine
  - how one system works
  - with respect to another?
- Comparing parallel I/O on many systems!



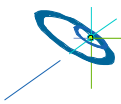
b_eff / b_eff_io — Basic design

b_eff:

- the repeating factors are controlled between:
  - large values, to reduce time measurement errors
  - small values, to reduce the total execution time
- i.e., time driven approach

b_eff_io:

- time driven approach
  - how many bytes can I write, rewrite and read in 10 minutes, using different access patterns
  - remembering the 10-minutes-rule
    ==> caching should not be a problem



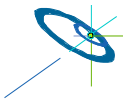
b_eff / b_eff_io — Output

b_eff:

- Minor results: accumulated bandwidth for different communication patterns, message sizes, ...
- Accumulated: **b_eff** (the one characteristic value!)
- Derived: Parallel communication capability of each processor = **b_eff / number of processors**

b_eff_io:

- Minor results: different partition sizes (PEs), pattern types, buffer sizes, ..., portion of the memory, that was written by b_eff_io
- Accumulated: **b_eff_io** (the one characteristic value!)
- Derived: Time to read/write the total memory with b_eff_io = **total memory size / b_eff_io**



H L R S

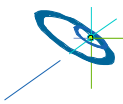
b_eff / b_eff_io — Algorithm

b_eff:

- Maximum over
  - MPI configuration parameters
  - MPI communication methods (Isend, Sendrecv, Alltoallv)
- Fixed: scheduled execution time T = 2 min.

b_eff_io:

- Maximum over
  - Usage aspects:
    - **how many processes are used and**
    - **how many parallel processors and threads are used for each process.**
  - Filesystem parameters
- Fixed: scheduled execution time T = 30 min.



H L R S

Definition of b_eff_io

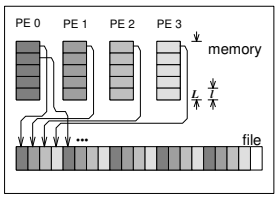
(Release 1.0)

b_eff_io := Maximum over all usage and filesystem parameters } manually

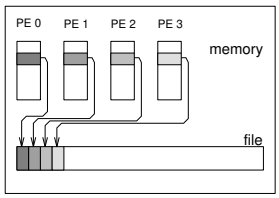
Average on write, rewrite, read
Average on five access pattern types
Average on several chunk size values*)
of measured bandwidth
} auto-matically, in time T=30 min.

*) defines the size of contiguous chunks written to disk and the contiguous chunk in memory written by each MPI call

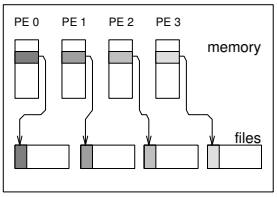
Definition of b_eff_io — the Pattern Types



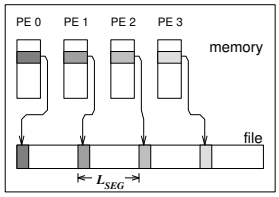
type 0: scattering *)



type 1: strided coll.



type 2: noncoll., separated



type 3 noncoll. / 4 coll. segmented

Pattern that can be optimized

Chunk sizes on disk:
- max (2MB, memory of one node/128) *)
- wellformed: 1MB, *) 32 kB, 1 kB,
- non-wellformed: 1MB+8B, *) 32 kB+8B, 1kB+8B

*) double weighted

Definition of b_eff_io — Bandwidth measurement

- Bandwidth measurement

```
MPI_Barrier()
start_time = MPI_Wtime()   at root only
repeat
   MPI_File_write() or MPI_File_read()
   MPI_Barrier()
   conti = (MPI_Wtime()- start_time) < time_unit
   MPI_Bcast(conti)
while conti
if (write access) MPI_File_sync()
MPI_Barrier()
end_time = MPI_Wtime()   at root only
bandwidth = (accumulated data size)
            / (end_time - start_time)
```

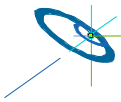
Output of the b_eff_io benchmark program

- the b_eff_io value

weighted average bandwidth for **write** : 21.530 MB/s on 16 processes
weighted average bandwidth for **rewrite** : 29.472 MB/s on 16 processes
weighted average bandwidth for **read** : 93.602 MB/s on 16 processes
Total amount of data written/read with each access method: 17589.682 MBytes
= 26.8 percent of the total memory (65536 MBytes)
b_eff_io of these measurements = 59.552 MB/s
on 16 processes with 128 MByte/PE and scheduled time=30.0 min
on sn6715 hwwt3e 2.0.5.34 unicosmk CRAY T3E
total memory / b_eff_io = 65536 Mbytes / 59.552 MB/s = 18.3 min.

- detailed results
  - as ASCII table
  - one page with 3+5 plots
    - all measurements sorted by access: write / rewrites / reads
    - and same sorted by pattern types: type-0 / type-1 / type-2 / type-3 / type-4

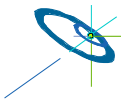
Detailed Results – an Example



Rolf Rabenseifner, Alice Koniges
Höchstleistungsrechenzentrum Stuttgart

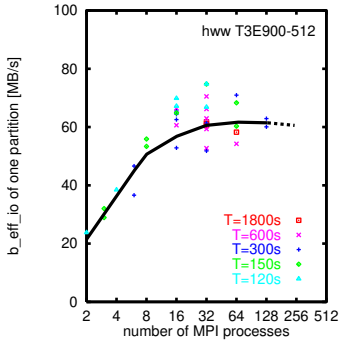
First Results — Comparing two systems

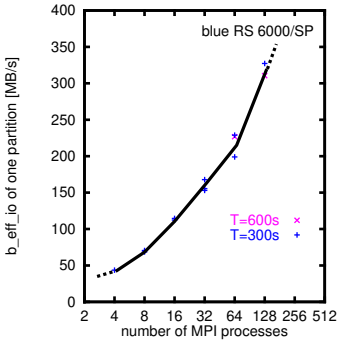
- Cray T3E 900-512 at HLRS/RUS, Stuttgart
  - 512 processors
  - 10 striped Raid-disks, connected via GigaRing
  - mpt.1.3.0.2 with ROMIO, modified: using asynchronous I/O
    - www.hlrs.de/mpi/mpi_t3e.html#StripedIO
    - www.hlrs.de/mpi/ufs_t3e/
  - theoretical peak throughput = 300 MB/s
- IBM RS 6000/SP at LLNL, called "blue pacific"
  - 336 SMP nodes with each 4 processors
  - benchmark: using 1 processor per node
  - IBM General Parallel File System (GPFS) with 20 VSD I/O server
  - ROMIO
  - measured peak performance: 950 MB/s read, 690 MB/s write (on 128 nodes)
- NEC SX-5Be/32M2 at HLRS/RUS, Stuttgart
  - 2 SMP nodes with each 16 processors
  - benchmark only on one SMP node
  - SFS filesystem, 4MB block size
  - I/O requests less than 1 MB are cached on 2 GB filesystem-memory-cache



Rolf Rabenseifner, Alice Koniges
Höchstleistungsrechenzentrum Stuttgart

First Results — Comparing b_eff_io (#processes)

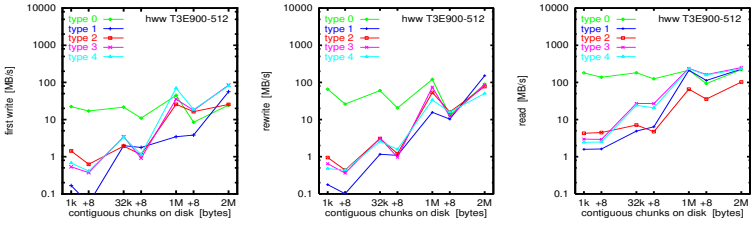




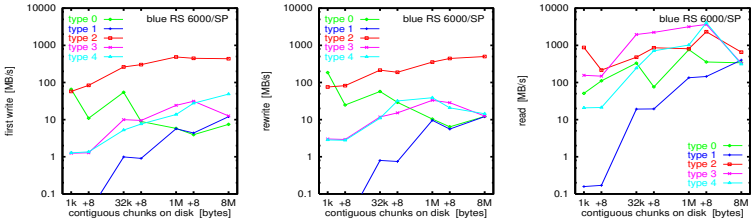
Full bandwidth on **Cray T3E**:
- about 30% of peak performance
- reached already with 8 processors!
  => optimal for any load:
  many small jobs ... one large job

Full Bandwidth **IBM SP**:
- about 35% of peak performance
- reachable only with high-CPU-count jobs
- higher absolute values
  (b_eff_io and total memory size)

First Results on Cray T3E and IBM SP

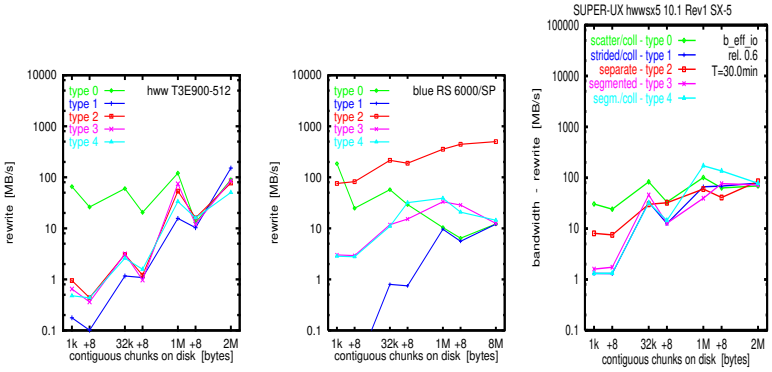


Cray T3E, HLRS, 32 PEs, T=10 min.: b_eff_io = 71 MB/s



IBM SP, LLNL "blue", 128 nodes, T=10 min.: b_eff_io = 311 MB/s

First Results: "rewrite" on Cray T3E, IBM SP, and NEC SX-5



Cray T3E
32 Pes
T=10 min.

IBM SP
128 nodes
T=10 min.

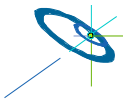
NEC SX-5
4 nodes
T=30 min.

First Results — Interpretation

- maximum bandwidth / partition sizes
- small influence of scheduled time T
- benchmarked platforms: MPI-I/O is optimal only for one pattern type
- but different optimal type on each platform
- non-wellformed data sizes: worse I/O bandwidth
- (re)write bandwidth << read bandwidth
- no chance to predict bandwidth for other patterns

Outlook

- www.top500clusters.org
- Issues
  - collecting hardware characteristics of clusters
  - several benchmark results
  - stored in a database
  - web-interface
  - each reader can define his own weights, and
  - can receive a personal weighted (ranked) list of all clusters
- Status
  - hardware information: some clusters already stored in database
  - benchmarks and web-interface: under discussion
  - b_eff_io under evaluation / chosen



H L R S

Summary

Effective I/O Bandwidth Benchmark (b_eff_io)

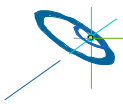
- characteristic average number for I/O bandwidth
- detailed information about several patterns:
  - access pattern types,
  - buffer sizes,
  - access methods (initial write, rewrite, read)
- 30 minutes for a first pass on any platform

First results

- Cray T3E900-512 and NEC SX-5 at HLRS
- IBM RS 6000/SP at LLNL (“blue pacific”)

Usable on MPP systems, SMP systems, and on clusters of SMPs

- more info: www.hlrs.de/mpi/b_eff_io/



H L R S