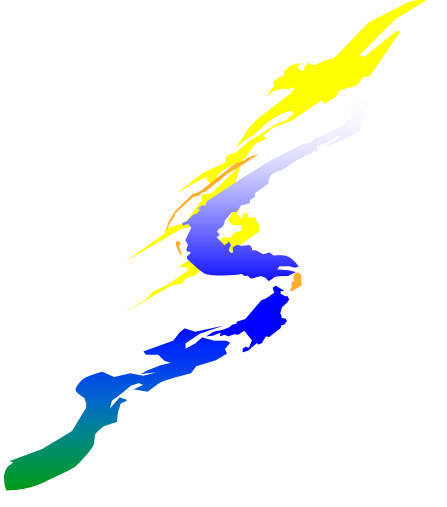
Modular I/O

John Bauer
Deep Computing
bauerj@us.ibm.com



I/O Optimization

- Analyze the I/O pattern
- Determine optimization method
- Optimize in user space
- Relink application with libtkio.a

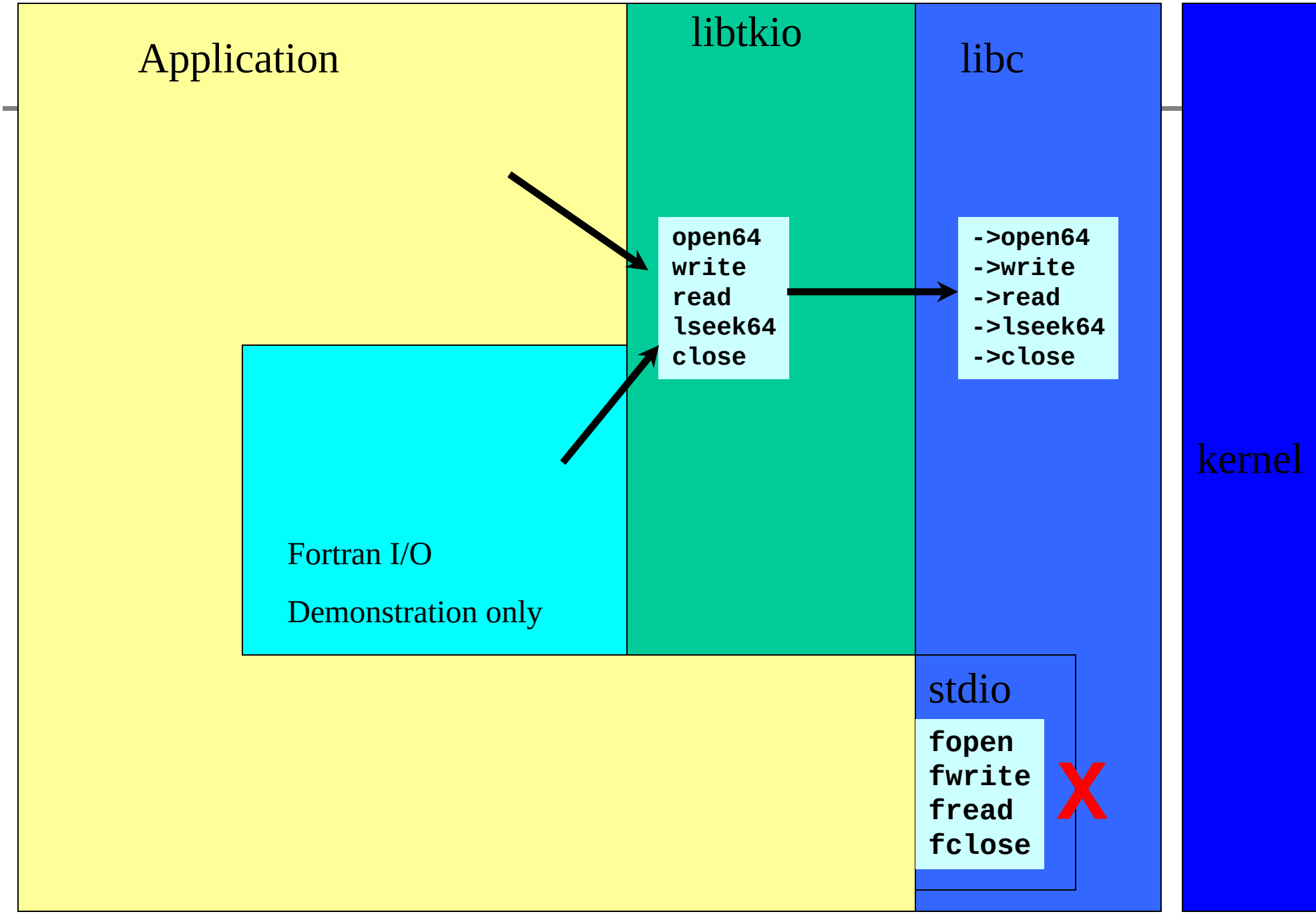
Relink with libtkio.a

- libtkio.a has shared object members
  - tkio.so 32 bit and 64 bit
  - Entry points for
    - open,open64,close,read,write,lseek,lseek64
    - fcntl,ffinfo,fstat,fstat64,fstatfs,fsync
    - ftruncate,ftruncate64
    - unlink,aio_...

Default tkio behaviour

- Uses dlopen and dlsym for runtime linking

tkio entry	calls
open64	libc(shr.o) open64
close	libc(shr.o) close
read	libc(shr.o) read
write	libc(shr.o) write
lseek64	libc(shr.o) lseek64
fsync	libc(shr.o) fsync
…	…


Application
libtkio
libc
kernel
open64
write
read
lseek64
close
->open64
->write
->read
->lseek64
->close
Fortran I/O
Demonstration only
stdio
fopen
fwrite
fread
fclose
X

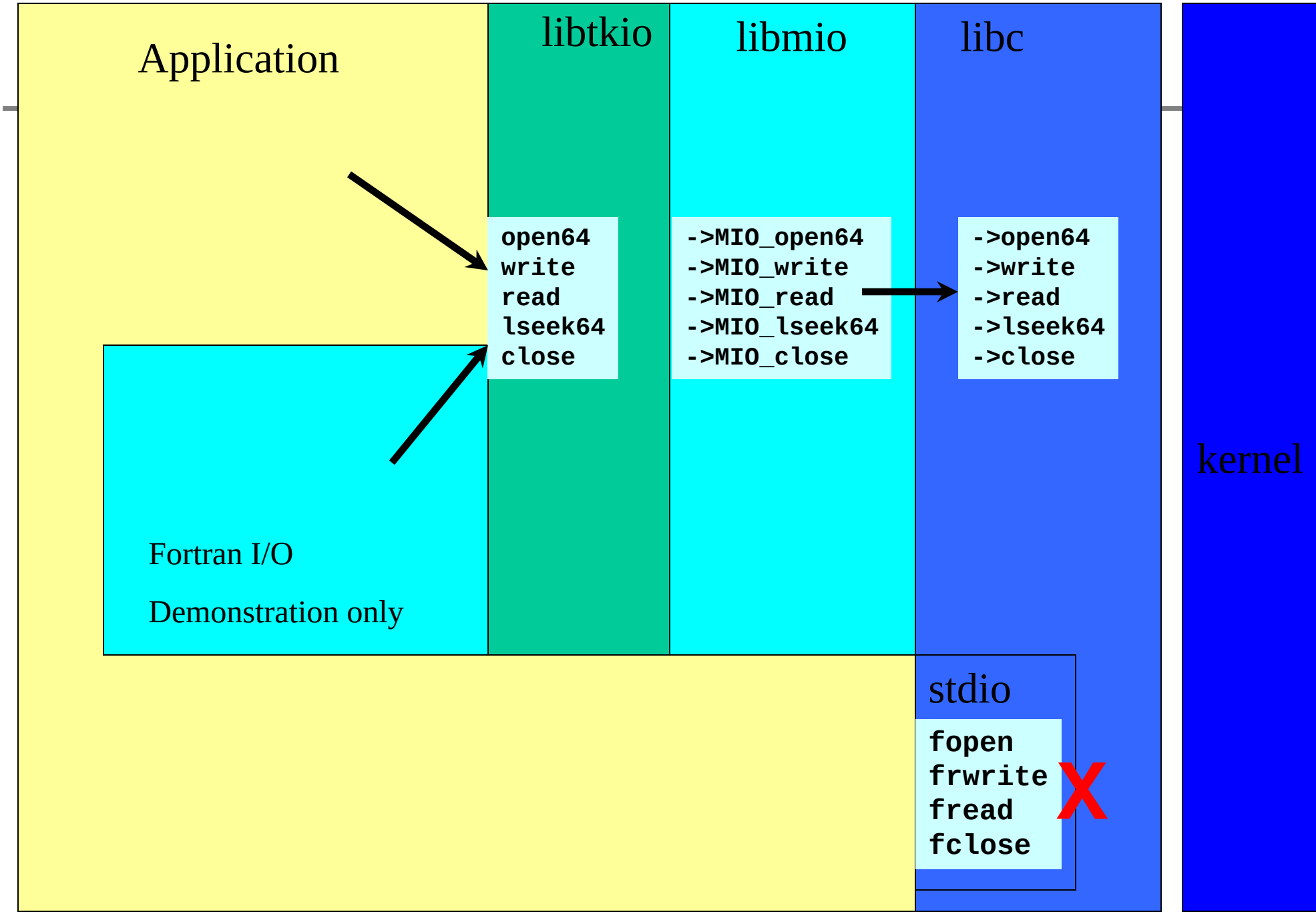
tkio runtime interface

- setenv TKIO_ALTLIB so_name/print/abort
  - so_name is name of shared library
    - Either name.so or libname.a(name.so)
- tkio calls function in so_name that returns a structure filled with I/O entry points to replace default entry points
- /print option outputs a print to stderr indicating success of load
- /abort issues exit(-1) if load is not successfull

tkio using MIO

- setenv TKIO_ALTLIB get_mio_ptrs_64.so

tkio entry	calls
open64	libmio(mio.o) MIO_open64
close	libmio(mio.o) MIO_close
read	libmio(mio.o) MIO_read
write	libmio(mio.o) MIO_write
lseek64	libmio(mio.o) MIO_lseek64
fsync	libmio(mio.o) MIO_fsync
…	…

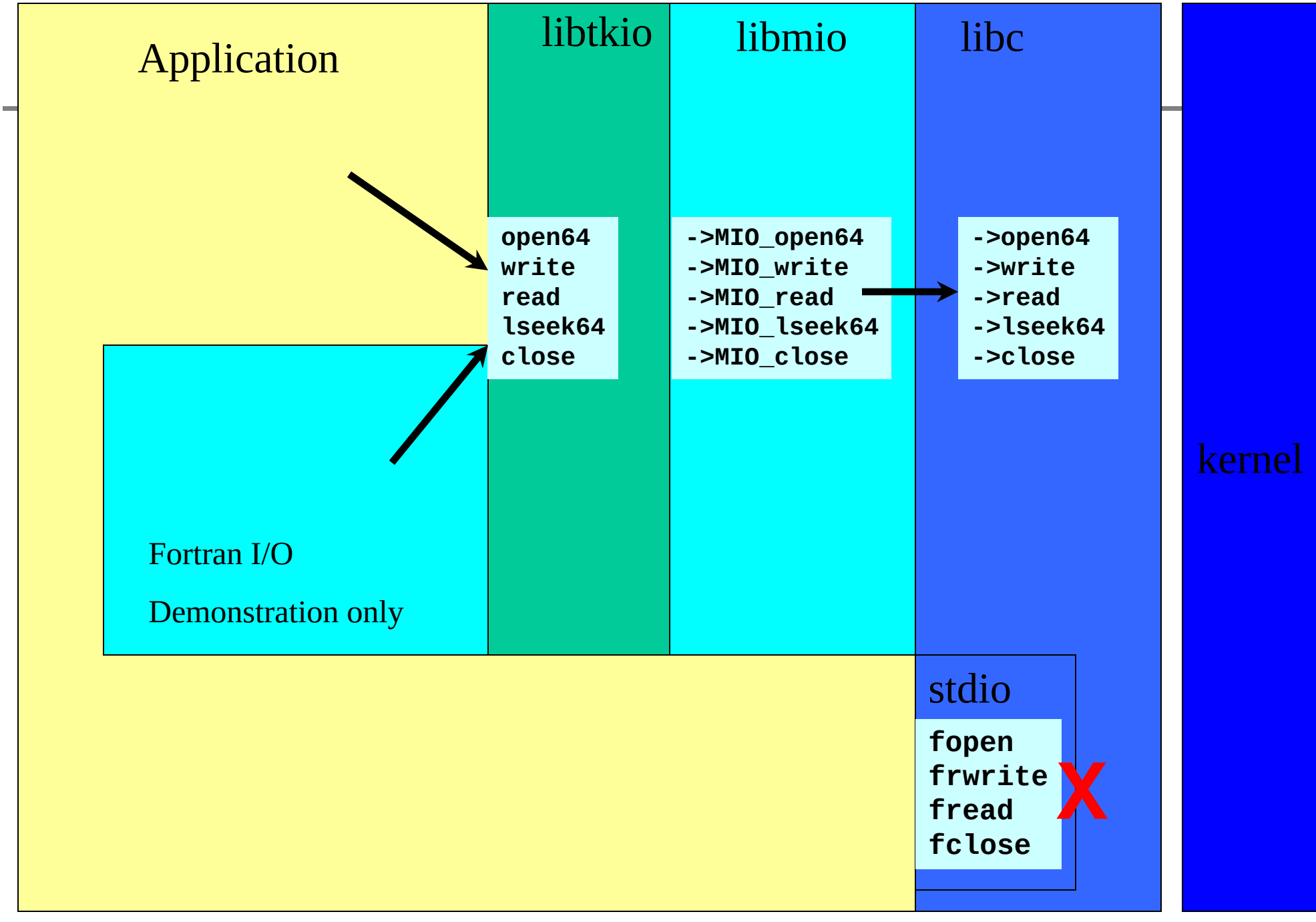

Application
libtkio
libmio
libc
kernel
open64
write
read
lseek64
close
->MIO_open64
->MIO_write
->MIO_read
->MIO_lseek64
->MIO_close
->open64
->write
->read
->lseek64
->close
Fortran I/O
Demonstration only
stdio
fopen
frwrite
fread
fclose
X

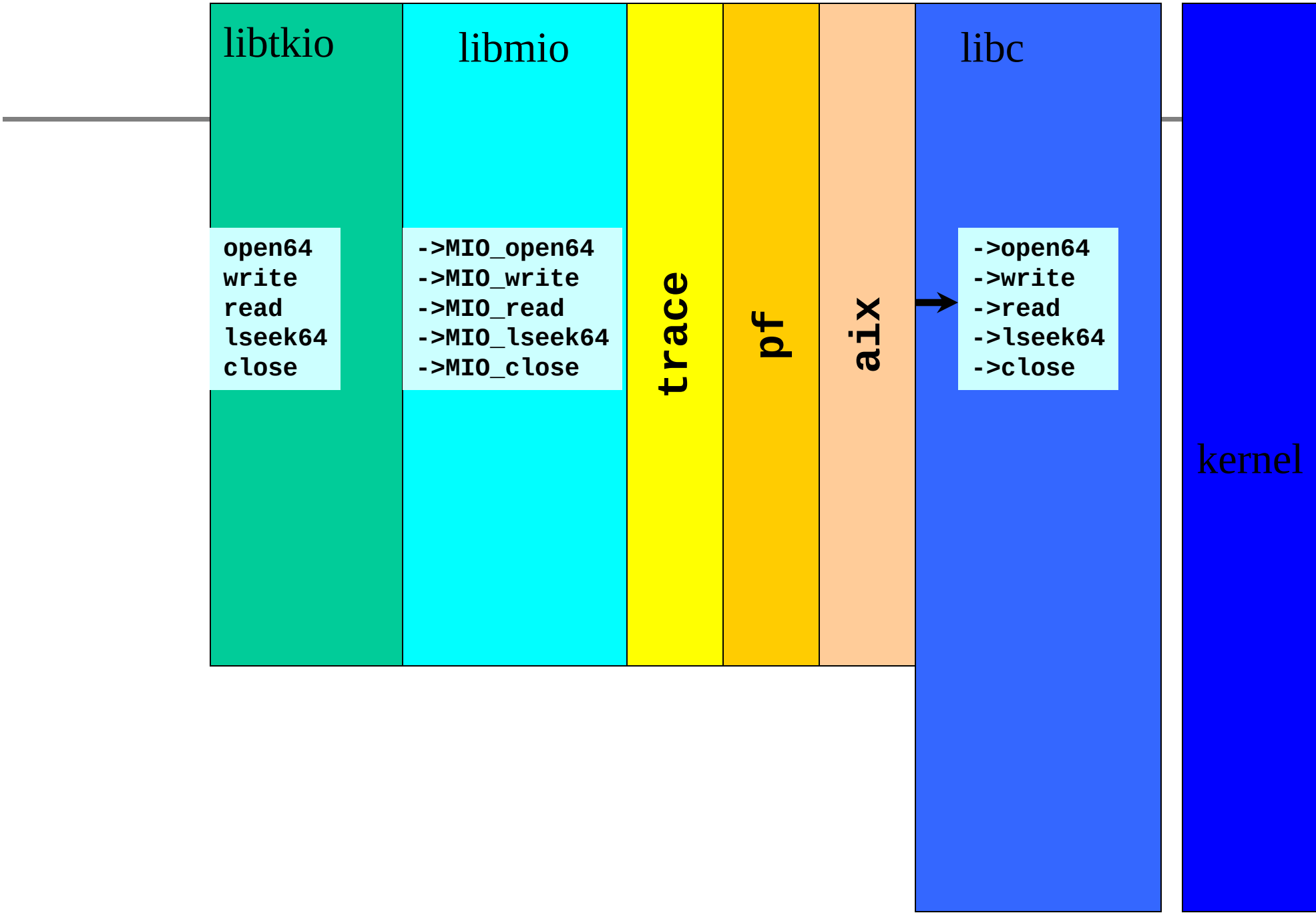
Modular I/O (MIO)

- flexible runtime interface
- MIO modules
  - ***mio***
  - ***trace***
  - ***pf***
  - ***Aix***
- MIO to be released with AIX 5.3H
- ***DataView (not released with AIX)***

MIO run time interface

- MIO_STATS="file name"
- MIO_FILES=" *.dat* [trace | pf | aix] *.inp [aix]"
- MIO_DEBUG="ALL"
- MIO_DEFAULTS="trace/mbytes , pf/cache=10m"


Application
libtkio
libmio
libc
kernel
open64
write
read
lseek64
close
->MIO_open64
->MIO_write
->MIO_read
->MIO_lseek64
->MIO_close
->open64
->write
->read
->lseek64
->close
Fortran I/O
Demonstration only
stdio
fopen
frwrite
fread
fclose
X


libtkio
libmio
libc
open64
write
read
lseek64
close
->MIO_open64
->MIO_write
->MIO_read
->MIO_lseek64
->MIO_close
trace
pf
aix
->open64
->write
->read
->lseek64
->close
kernel

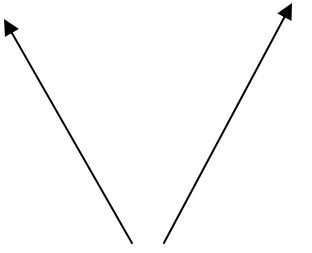
trace module

- summary of file activity
- binary events file
- low cpu overhead
- typical options
  - /stats
  - /mbytes /gbytes /tbytes
  - /events=*mio.evt*

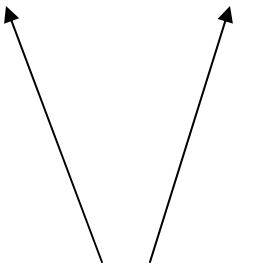
MSC.NASTRAN
trace output from program <->pf

```
Trace close : program <-> pf : /bmwfs/cdh108.T20536_13.SCR300 :
              (281946/2162.61)=130.37 mbytes/s
     current size=0    max_size=16277
  mode =0777  sector size=4096
  oflags =0x302=RDWR  CREAT  TRUNC
  open                1      0.01
  write          478193    462.10      59774      59774     131072  131072
  read          1777376   1700.48     222172     222172     131072  131072
  seek           911572      2.83
  fcntl               3      0.00
  trunc              16      0.40
  close               1      0.03
  size           127787
```

Number of occurrences



Mbytes requested and Mbytes delivered

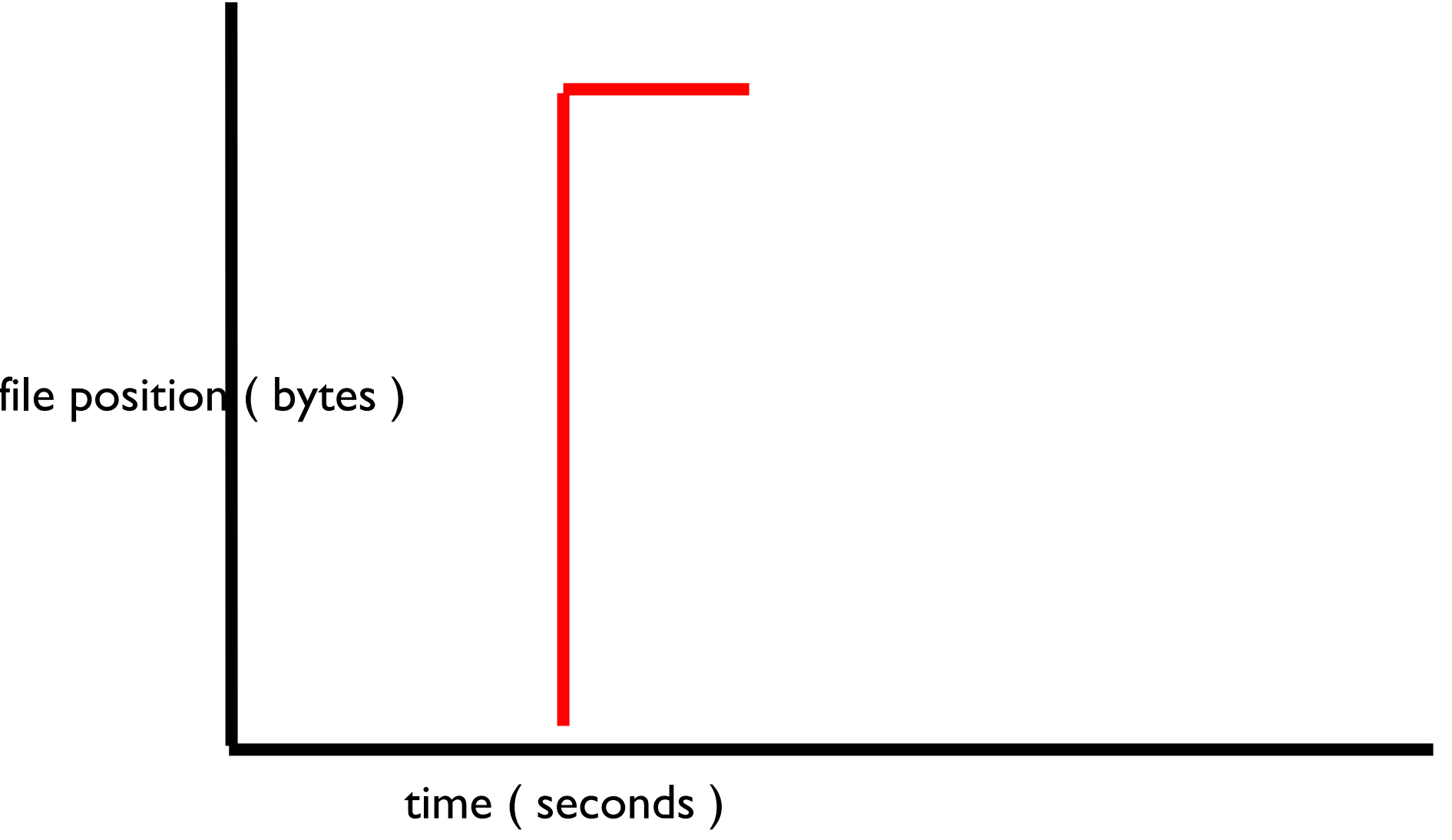


Min/Max Request size in bytes

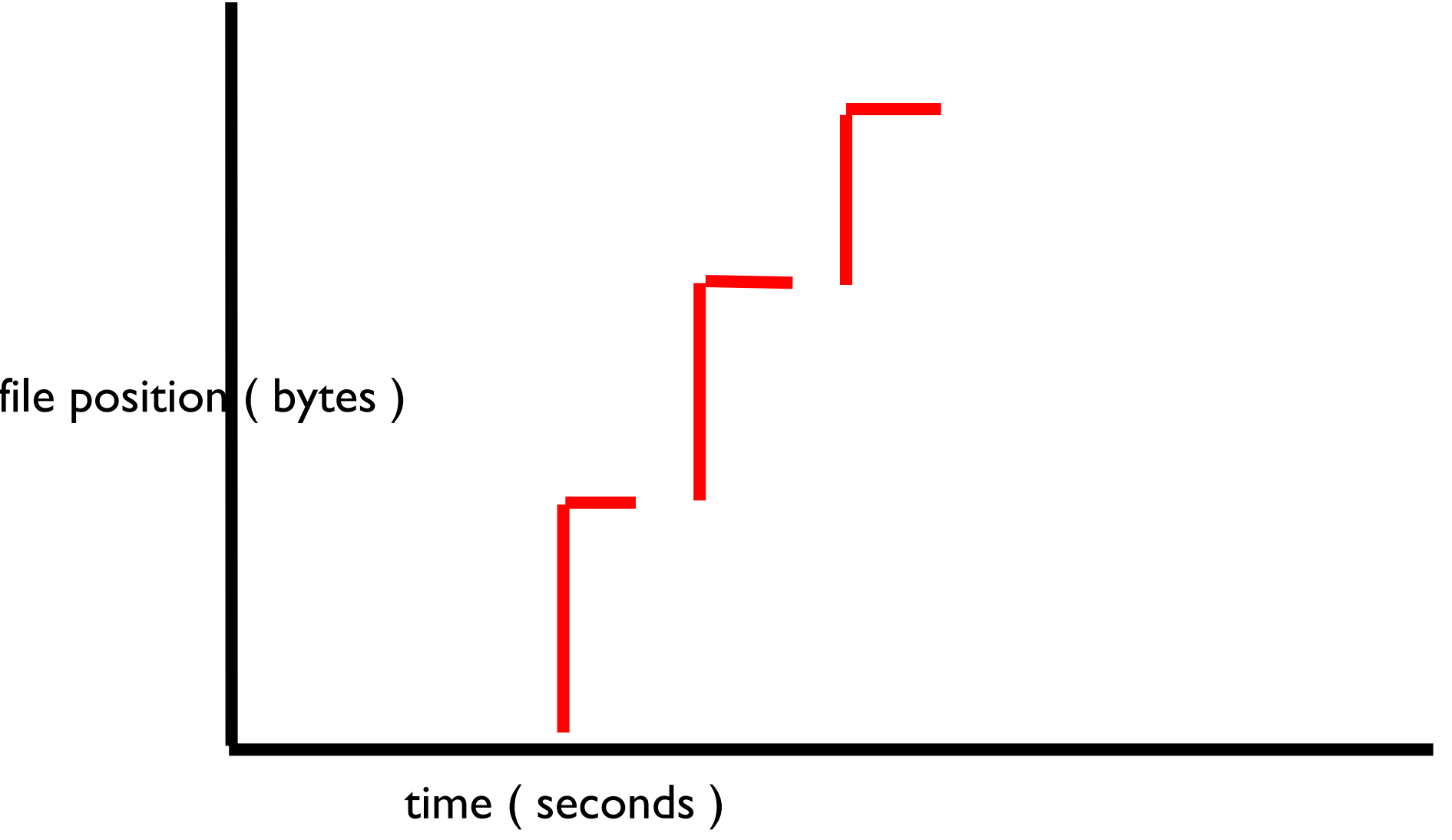
MSC.NASTRAN trace output

```
Trace close : pf <-> aix : /bmwfs/cdh108.T20536_13.SCR300 :
                (276645/1460.73)=189.39 mbytes/s
    current size=0   max_size=16276
  mode =0777  sector size=4096
  oflags =0x8000302=RDWR  CREAT  TRUNC  DIRECT
  open              1     0.01
  write          4382   154.86        684         684  131072 2097152
  awrite        33390     1.42      58491       58491  131072 2097152
   suspend      33390   240.00      242.27 mbytes/s
  read           5178   272.71      10354       10354 1048576 2097152
  aread        103560     5.70     207115      207115  524288 2097152
   suspend     103560   786.04      261.59 mbytes/s
  seek         136950     0.00
  fcntl             3     0.00
  trunc            16     0.40
  close             1     0.00
  size          11013
  pages        138477
```

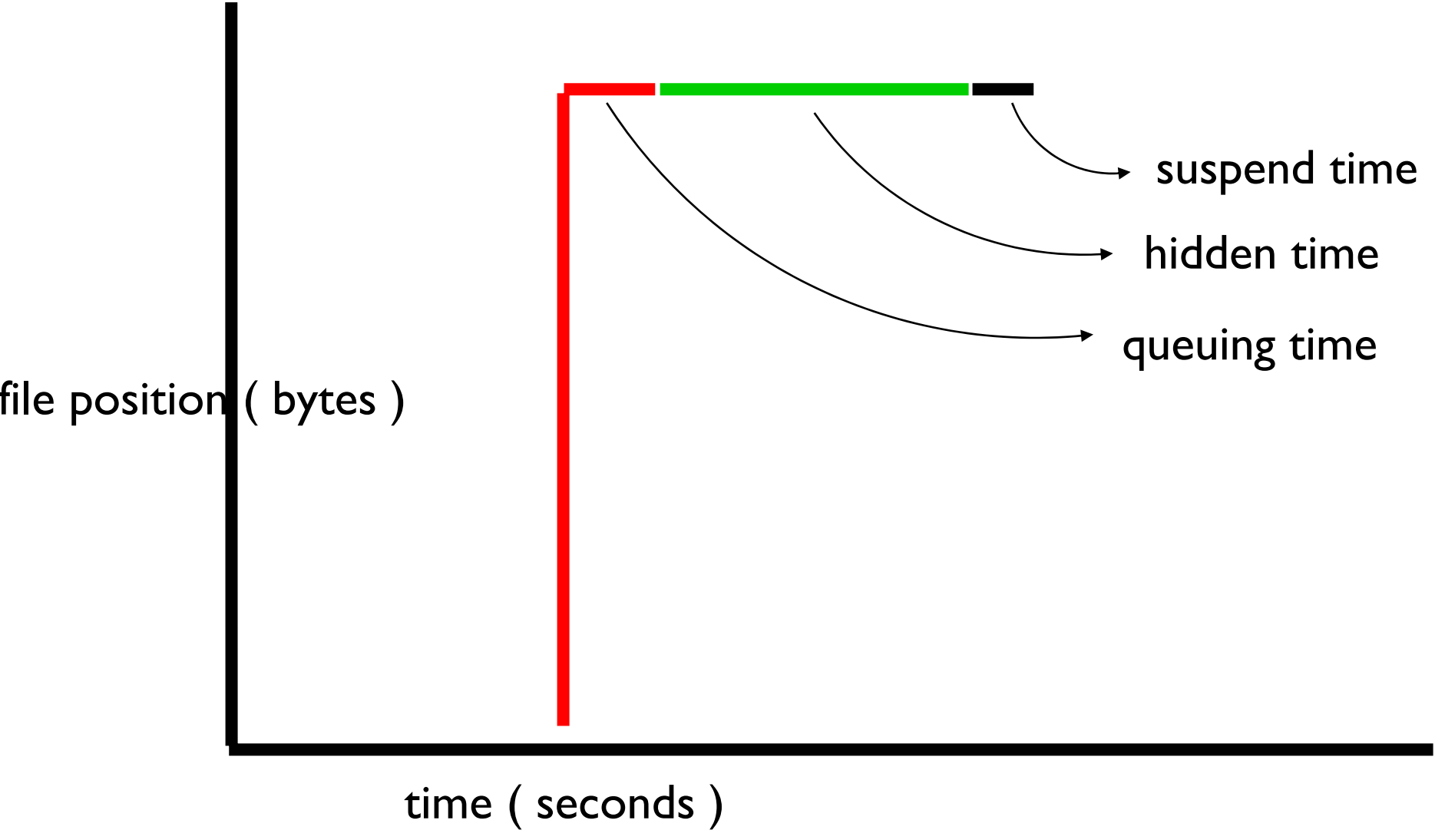
DataView file activity plot



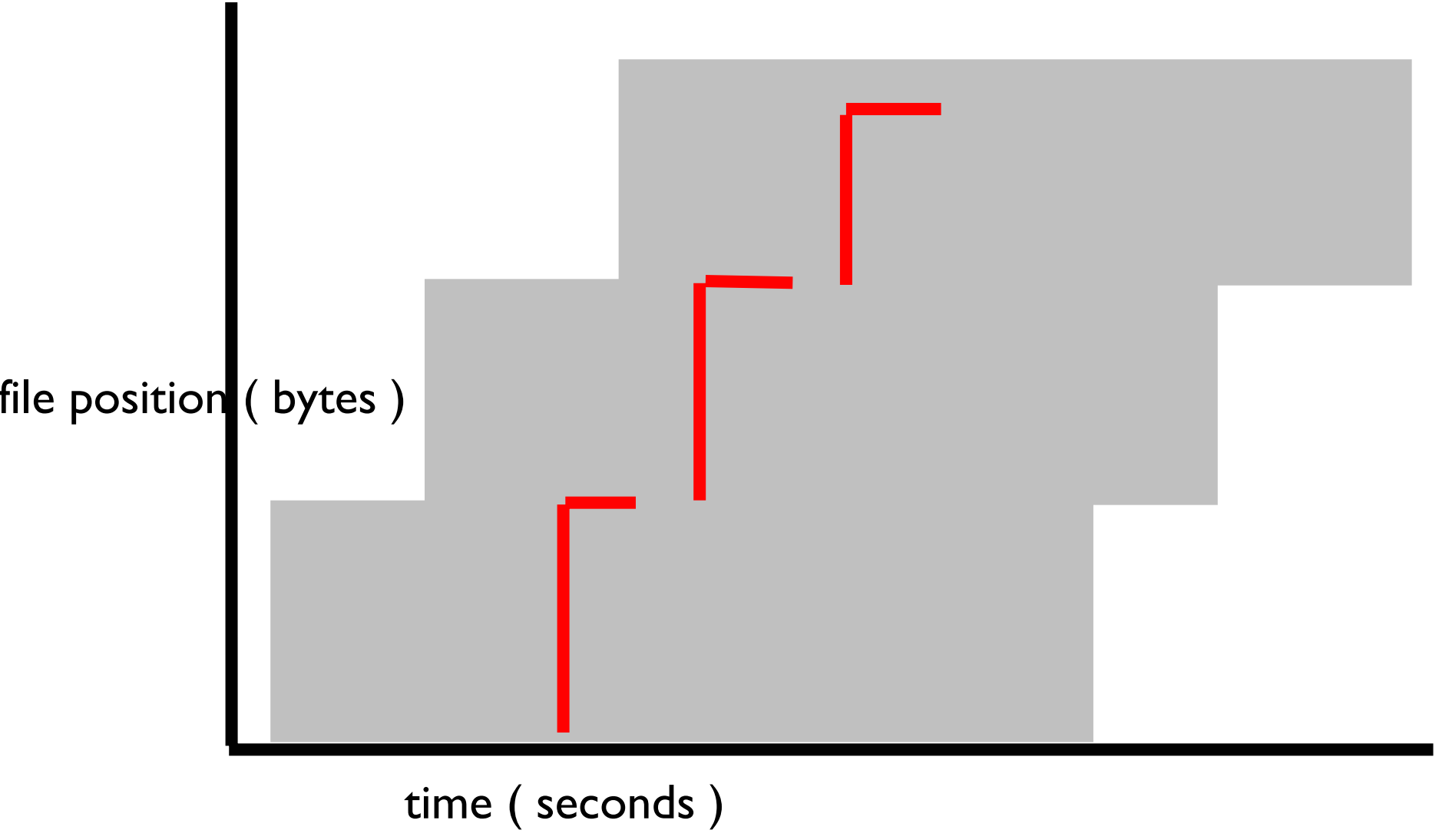
DataView file activity plot

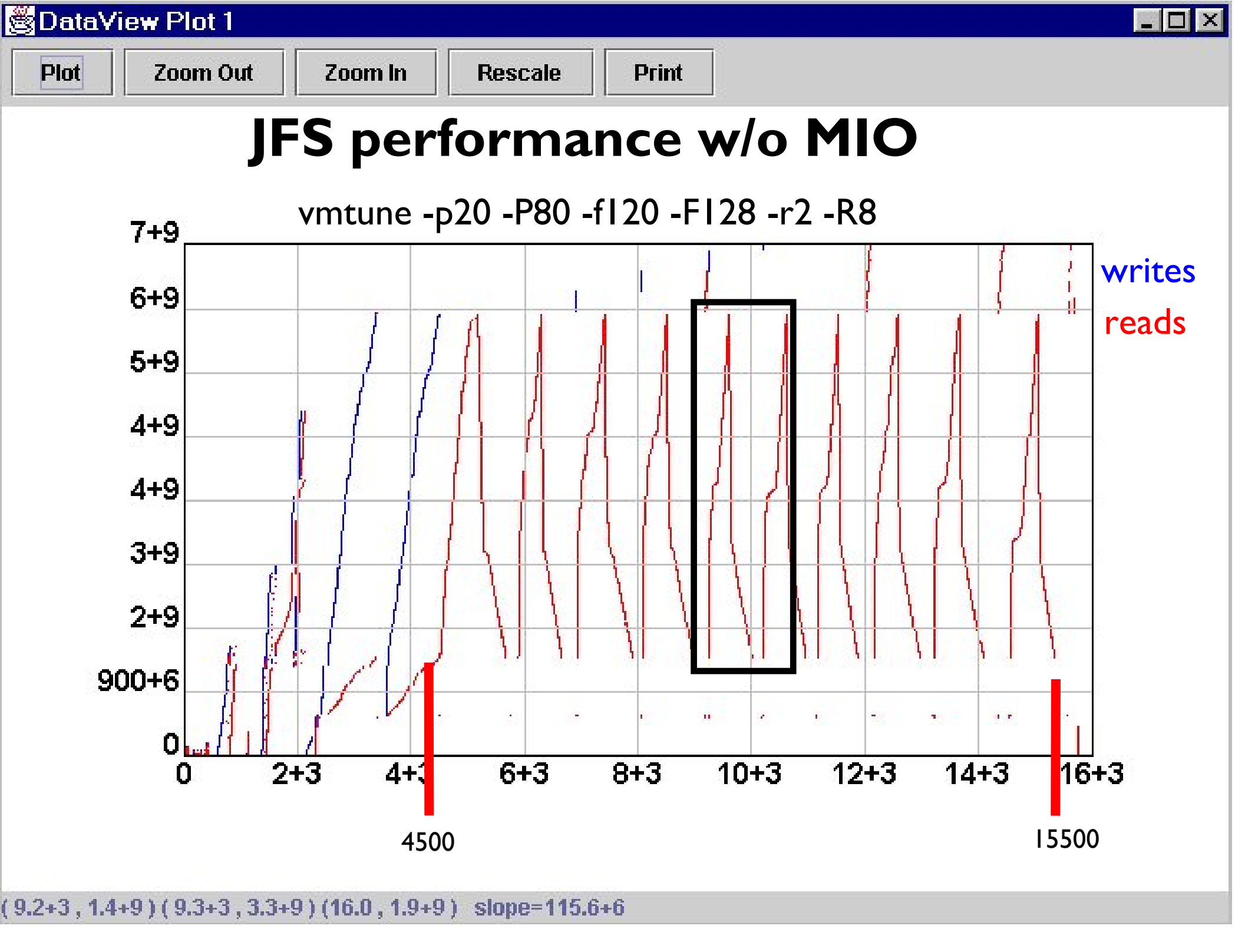


Asynchronous I/O plotting



cache page activity

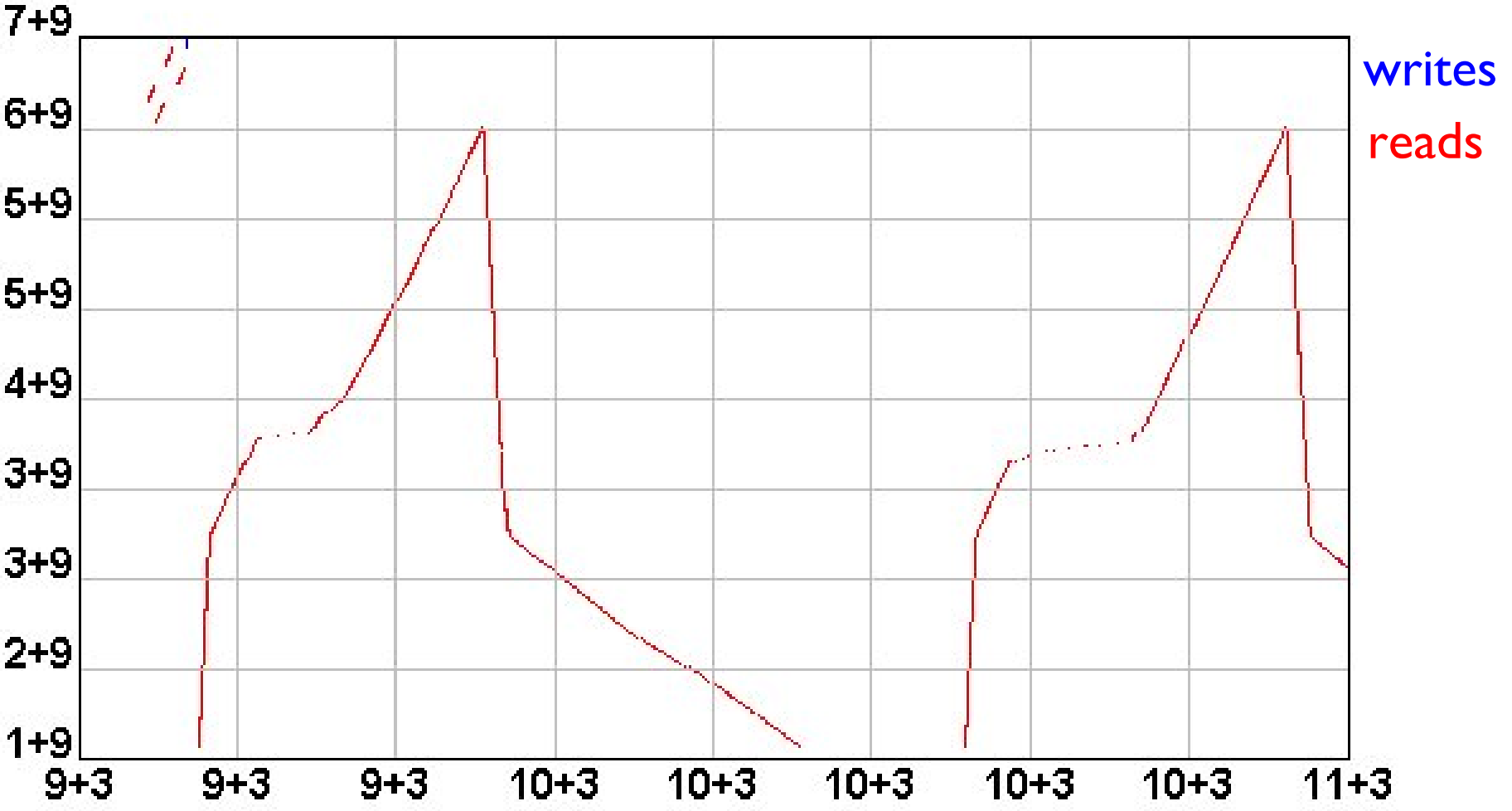



DataView Plot 1
Plot
Zoom Out
Zoom In
Rescale
Print
JFS performance w/o MIO
vmtune -p20 -P80 -f120 -F128 -r2 -R8
7+9
6+9
5+9
4+9
4+9
3+9
2+9
900+6
0
writes
reads
0
2+3
4+3
6+3
8+3
10+3
12+3
14+3
16+3
4500
15500
(9.2+3 , 1.4+9) (9.3+3 , 3.3+9) (16.0 , 1.9+9) slope=115.6+6

DataView Plot 1

Plot | Zoom Out | Zoom In | Rescale | Print

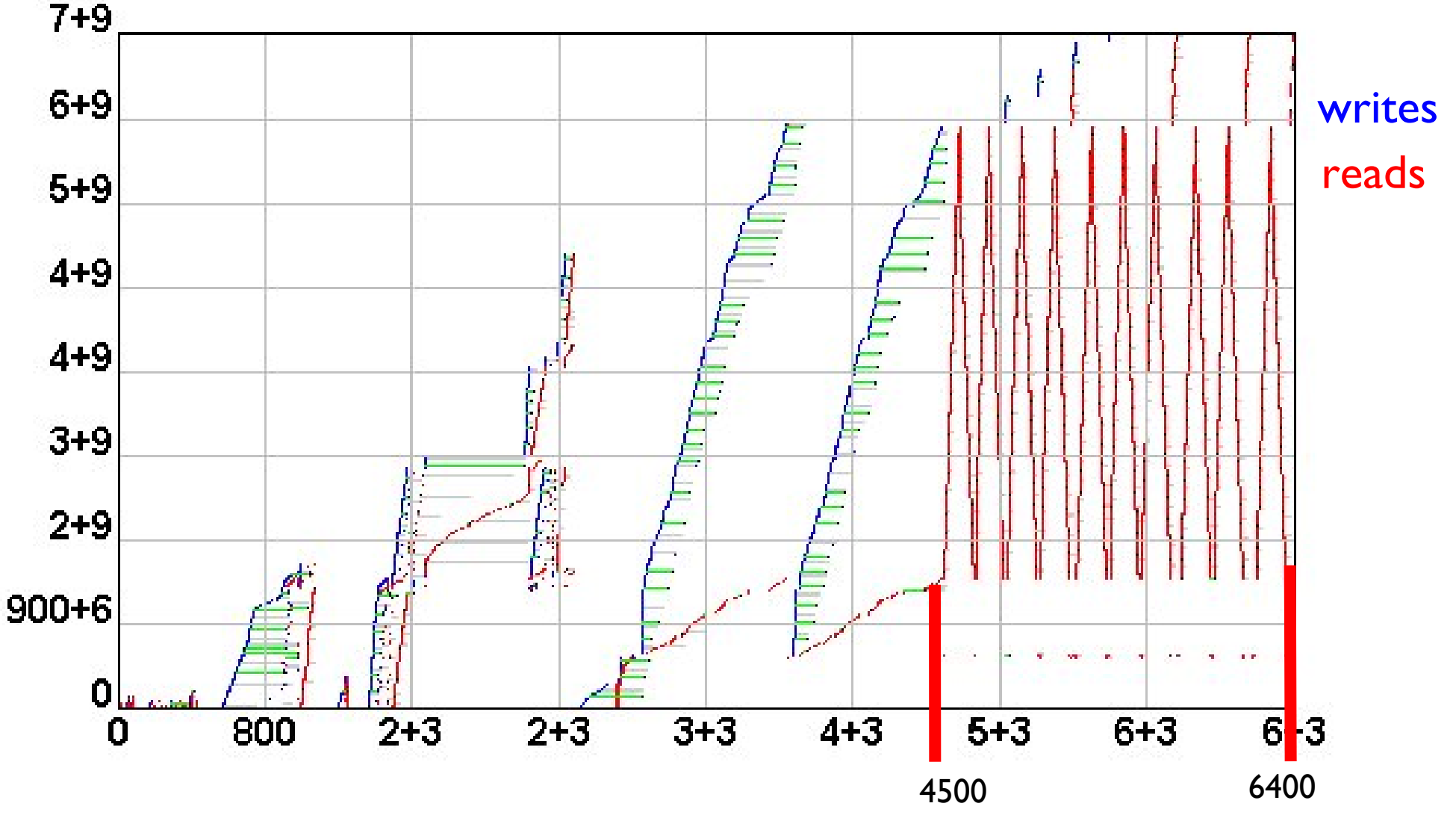
JFS performance w/o MIO



(9.2+3 , 1.4+9) (9.3+3 , 3.3+9) (16.0 , 1.9+9) slope=115.6+6

Plot | Zoom Out | Zoom In | Rescale | Print

JFS performance with MIO

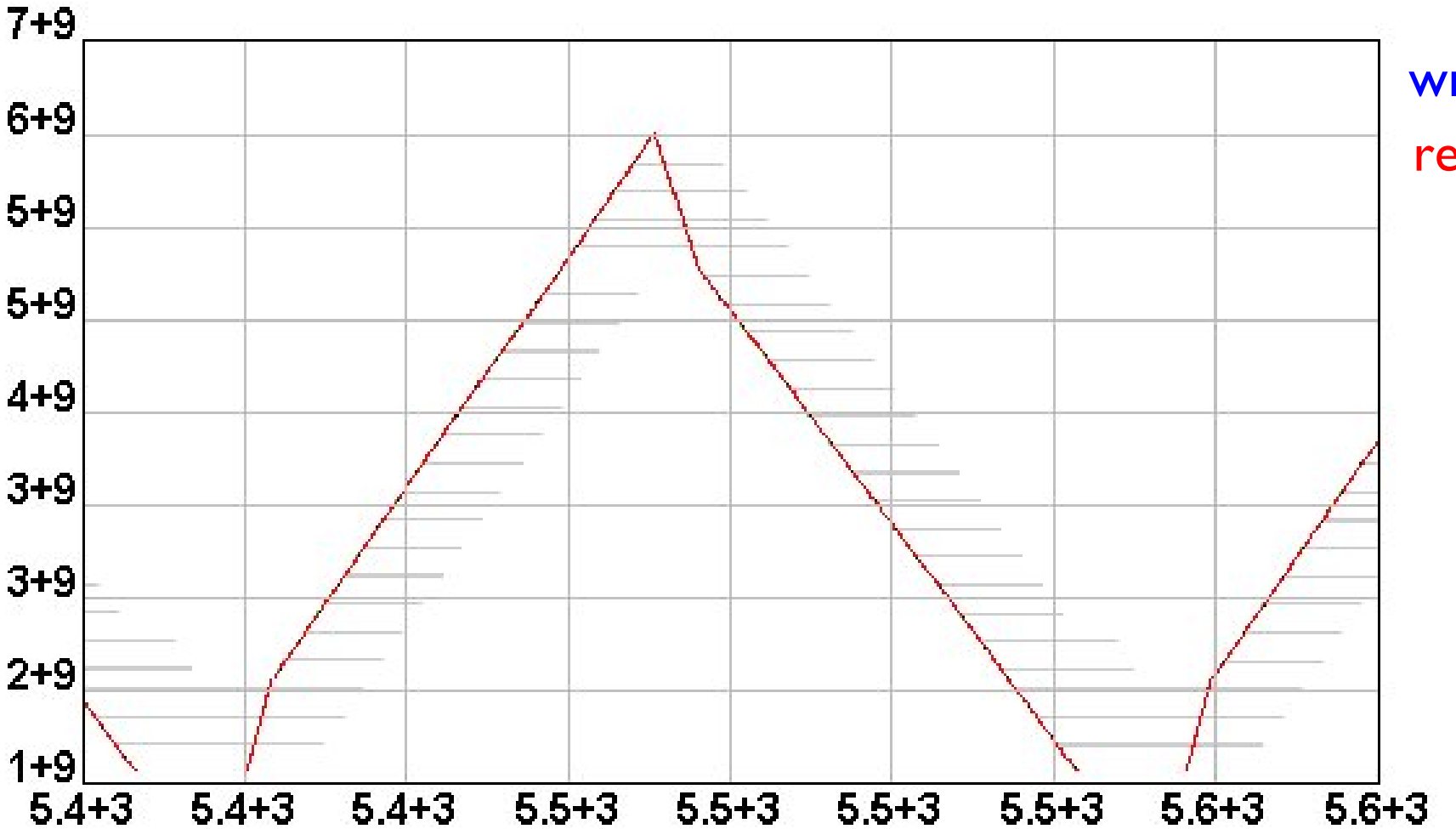


3.3 , -456+6) (6.3+3 , 7+9) (6.3+3 , 7+9) slope=1.1+6

DataView Plot 1

Plot | Zoom Out | Zoom In | Rescale | Print

JFS performance with MIO



(5.41+3 , 2.1+9) (5.47+3 , 6.1+9) (68.30 , 4.0+9) slope=58.07+6

pf module

- detects sequential I/O
- user memory buffering
- options
  - /global
  - /cache_size=10m
  - /page_size=1m
  - /prefetch=1
  - /stride=1
  - /direct
  - /stats

MSC.NASTRAN pf output

```
pf close for /bmwfs/cdh108.T20536_13.SCR300
   global  cache   0:   150 pages of 2097152 bytes
   29739/29749 pages not preread for write
   138316/139754 prefetches : prefetch=3
        29576 write behinds
       478193 writes
      1777376 reads
   page writes 37772/33124
   mbytes transferred
   program -->        59774   --> pf -->        59176  --> aix
   program <--       222172   <-- pf <--       217469  <-- aix
```

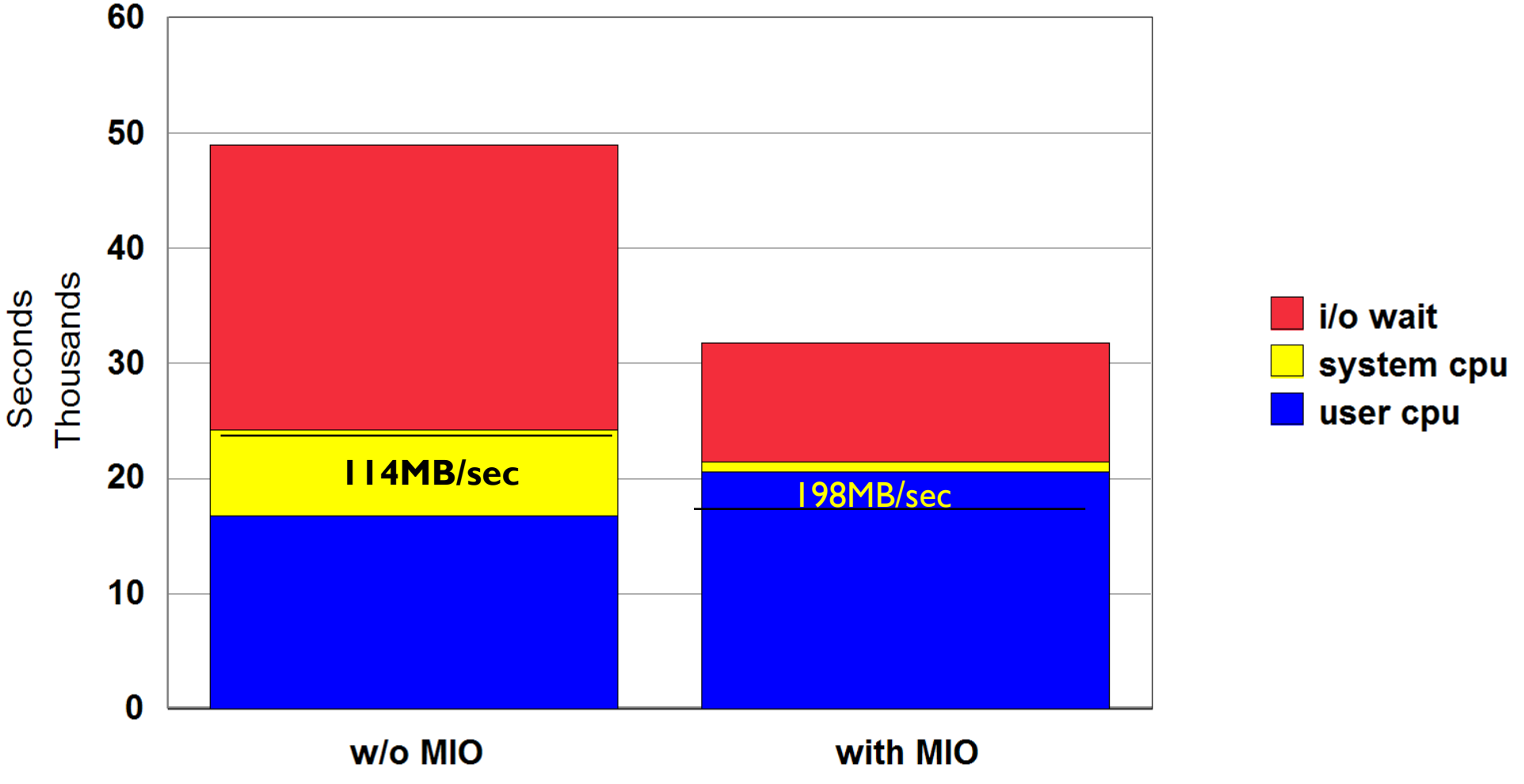
MSC.Nastran performance gains

16 cpu 32GB NH2 node

8 SSA, 16 loops, 4 disk/loop

2.2M dof, 767GB I/O, 8 copies

2GB memory per copy



MIO Summary

- Demonstrated performance gains
- Simple to implement
- Flexible run time interface
- Delivered as a shared object library for AIX
- To be released with AIX 5.3H