PIO Library Update

Jim Edwards
CSEG/CGD/NCAR

What is PIO?

- An interface library layer between geophysical models and lower level IO libraries
- Focus on improving output performance to NetCDF format files
- IO interface in CIME, CESM component models and MPAS
- Option available in WRF, ESMF

Introducing PIO 2.0

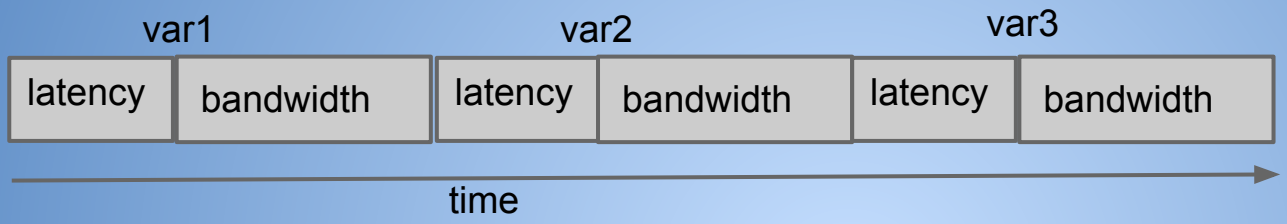
- Complete rewrite in C
- Original F90 API is retained
  - minor API change from PIO1.0
- More scalable data movement methods
- Available for download now
- In CIME this summer

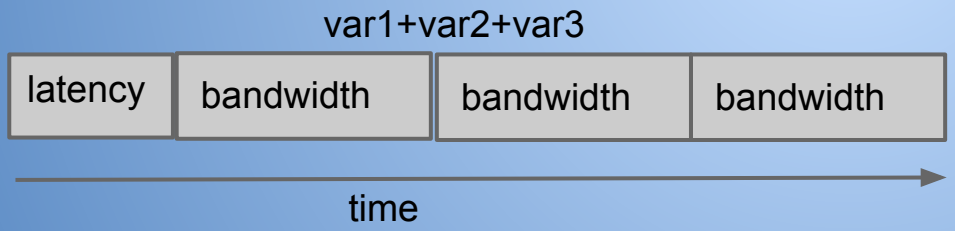
Why do we need PIO?

Both PNetCDF and NetCDF4/HDF5 have parallel interfaces

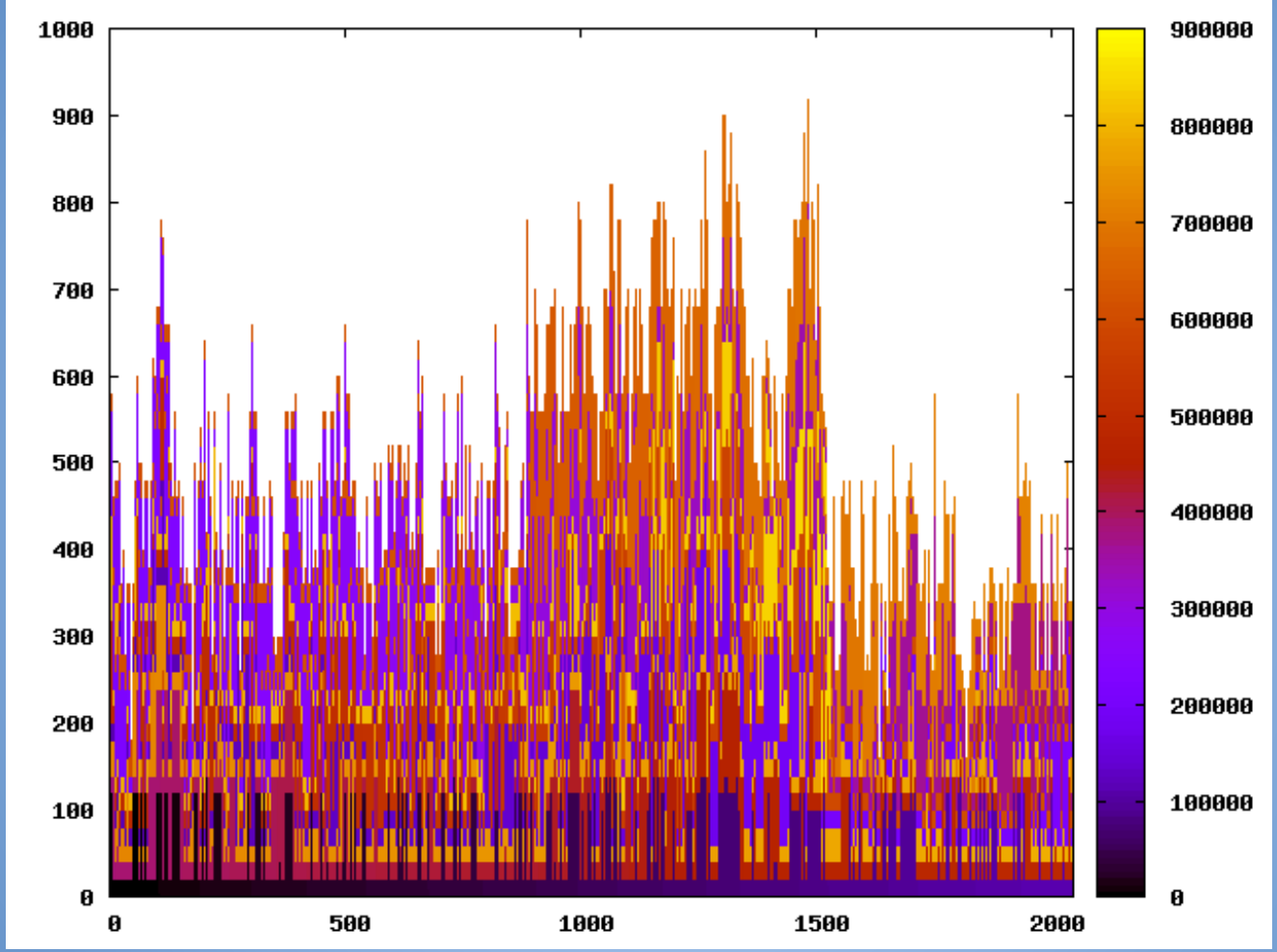
PIO exploits application level knowledge to improve efficiency of communications

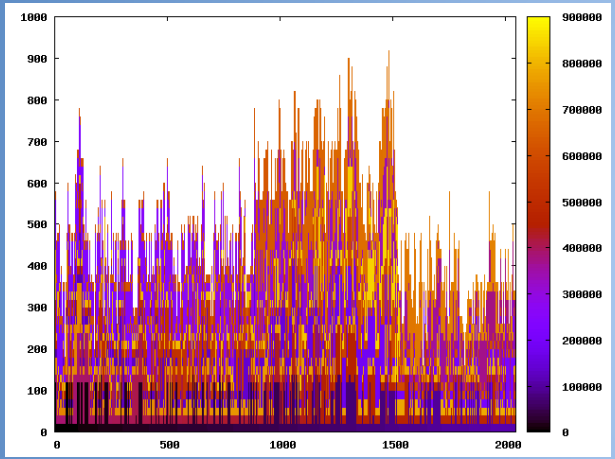
data aggregation





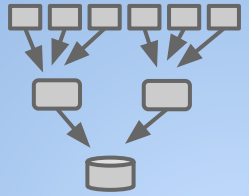
- latency is the fixed cost of the operation
- bandwidth is the cost per unit data
- data aggregation reduces latency by combining operations

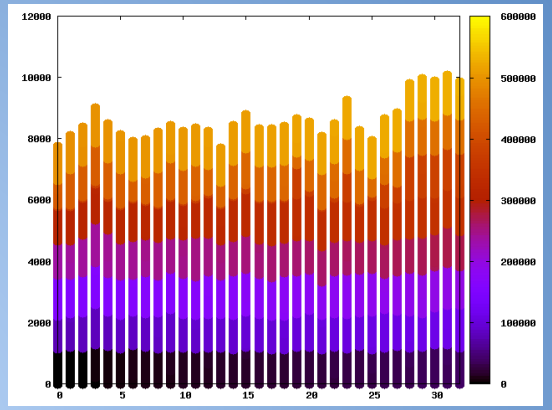

1000
900
800
700
600
500
400
300
200
100
0
0
500
1000
1500
2000
900000
800000
700000
600000
500000
400000
300000
200000
100000
0



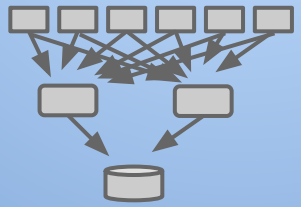
Example decomposition: CLM data on 2048 tasks.

Subset rearranger gives better scaling

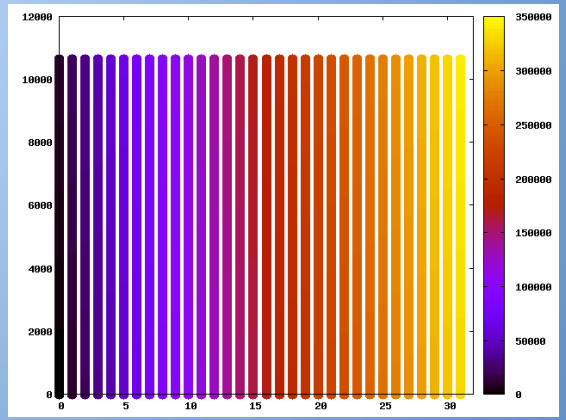




Box rearranger gives optimal data layout



Data rearranged to 32 IO tasks



Subset rearranger performance on yellowstone with data aggregation (MB/s)

iotasks vs vars	8	16	32	64	128	256	512	1024	2048
1	64.2	162.4	128.3	65.7	75.8	47.3	76.8	35.9	46.
5	389.2	391.3	275.5	189.4	146.8	143.	150.6	144.2	117.
10	338.9	584.8	417.4	241.7	175.2	188.0	181.5	143.4	152.
100	977.9	1478.2	1257.4	915.8	898.8	934.6	1005.0	956.8	877.

pnetcdf includes data aggregation support
netcdf4p does not

data aggregation performs well with pnetcdf using both box and subset rearrangers
data aggregation with netcdf4p performs well only with box rearranger.

Collaboration with ALCF and IBM

- Work with Paul Coffman of IBM to improve IO performance of CESM on MIRA
- Led to development of new one-sided optimizations in ROMIO MPI-IO library
- Optimizations and improvements in PNetCDF library
- Improvements in PIO

NetCDF file layout tips

- put non-decomposed data first
- put non-record variables before record variables
- group variables with same decomposition
- define the file once - redef is expensive

shr_spmd

- Routines to improve performance of MPI gather and alltoall operations originally developed by Pat Worley of ORNL
- Currently in several CESM component models, PIO and MCT/MPEU
  - minor differences in implementation
- Plan to move to CIME shared library and provided better algorithm tuning tools

PIO source repository has moved to github

https://github.com/PARALLELIO/ParallelIO

We encourage collaboration and contributions to make a better, more widely used library.

PIO Development plans

- Improved documentation
- In Situ data processing
  - diagnostic calculations
  - time averaging
- Performance auto-tuning
- Continued collaboration with
  - lower level library developers
  - component model developers

Thank you

- Wei-keng Laio
  - Northwestern University - PNetCDF developer
- Paul Coffman
  - IBM - ROMIO developer
- Kevin Paul, John Dennis
  - ASAP/CISL/NCAR - PIO contributers
- Rob Latham
  - ALCF - ROMIO developer
- Jayesh Krishna
  - ALCF - PIO contributer
- CSEG