Coarrays in GNU Fortran 5

A. Fanfarillo, T. Burnus, V. Cardellini,
S. Filippone, D. Nagle, D. Rouson
{fanfarillo,cardellini}@ing.uniroma2.it, sfilippone@uniroma2.it
burnus@net-b.de, damian@sourceryinstitute.org, dnagle@ucar.edu

What is a coarray? Why use it?

- Coarray Fortran (CAF) is a syntactic extension of Fortran 95/2003 which has been included in the Fortran 2008 standard.
- The main goal is to allow Fortran users to realize parallel programs without the burden of explicitly invoke communication functions or directives (MPI, OpenMP).
- Coarrays are based on the Partitioned Global Address Space model.
- The PGAS model attempts to combine the advantages of a SPMD programming style for distributed memory systems (as employed by MPI) with the data referencing semantics of shared memory systems.
- A coarray could be a scalar or array, static or dynamic, and of intrinsic or derived type.

Coarrays allow Fortran users to write parallel programs with nothing more than Fortran!

Coarrays at work

```
real, dimension(10), codimension[*] :: x, y
integer :: num_img, me
num_img = num_images()
me = this_image()
x(2) = x(3)[7] ! get value from image 7
x(6)[4] = x(1) ! put value on image 4
x(:)[2] = y(:) ! put array on image 2
sync all
x(1:10:2) = y(1:10:2)[4] ! strided get from image 4
```

Every transfer is performed using the regular Fortran array syntax. The basic transfer coarray operations are Put, Get, Strided Put and Strided Get. Synchronization primitives are required to maintain program correctness.

Compiler support for coarrays

- Cray
  - → Pros: Mature, efficient and complete implementation.
  - → Cons: Commercial compiler. Requires proprietary hardware.
- Intel
  - → Pros: Quite complete coarray implementation.
  - → Cons: Commercial compiler. It works only on Linux and Windows.
- Rice, OpenUH, G95
  - → Pros: Free compilers.
  - → Cons: Not complete standard support. Not widely used.

Goal

Realize a complete, efficient and free coarray implementation on a free, efficient and widely used compiler like (upcoming) GNU Fortran 5, to study the relative performance of message-passing and PGAS communication layers on commodity and commercial hardware platforms.

GFortran coarray implementation

- GFortran delegates to an exernal library the actual coarray implementation.
- The most complete implementation is MPI based and it is called **LIBCAF_MPI**.
- A GASNet implementation is provided as "expert version".
- Project code base at `http://www.opencoarrays.org`

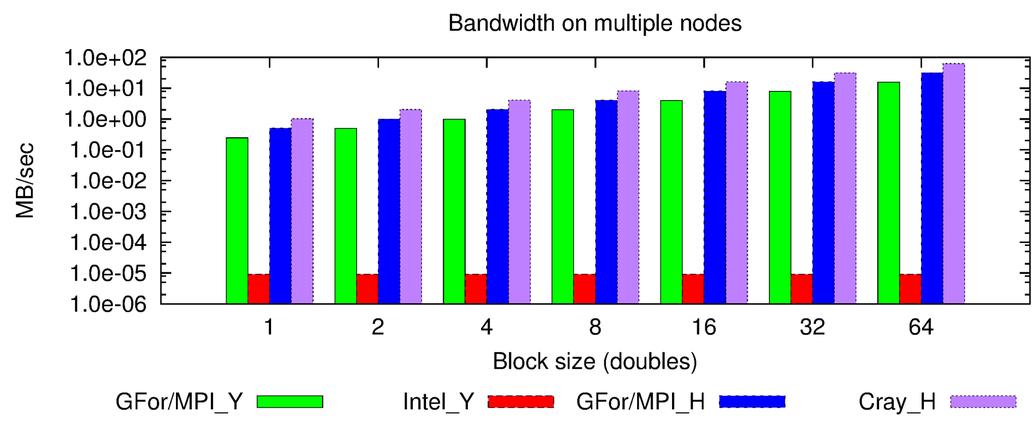
ARMCI, OpenSHMEM and any other communication library can be used by GFortran.

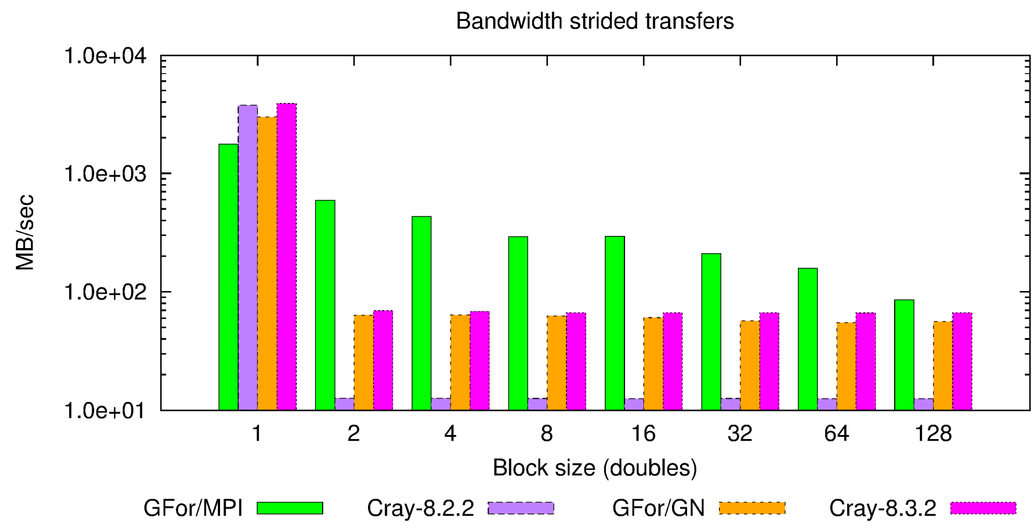
GFortran coarray comparison

- We compare the MPI and GASNet coarray implementations with the Cray and Intel compiler.
- We use the EPCC CAF suite to probe the basic coarray operations and a 3D Distributed Transpose for a comparison with a pure MPI program.

GFortran and LIBCAF_MPI can be run on any architecture capable of building GCC 5.0 and a standard MPI implementation.

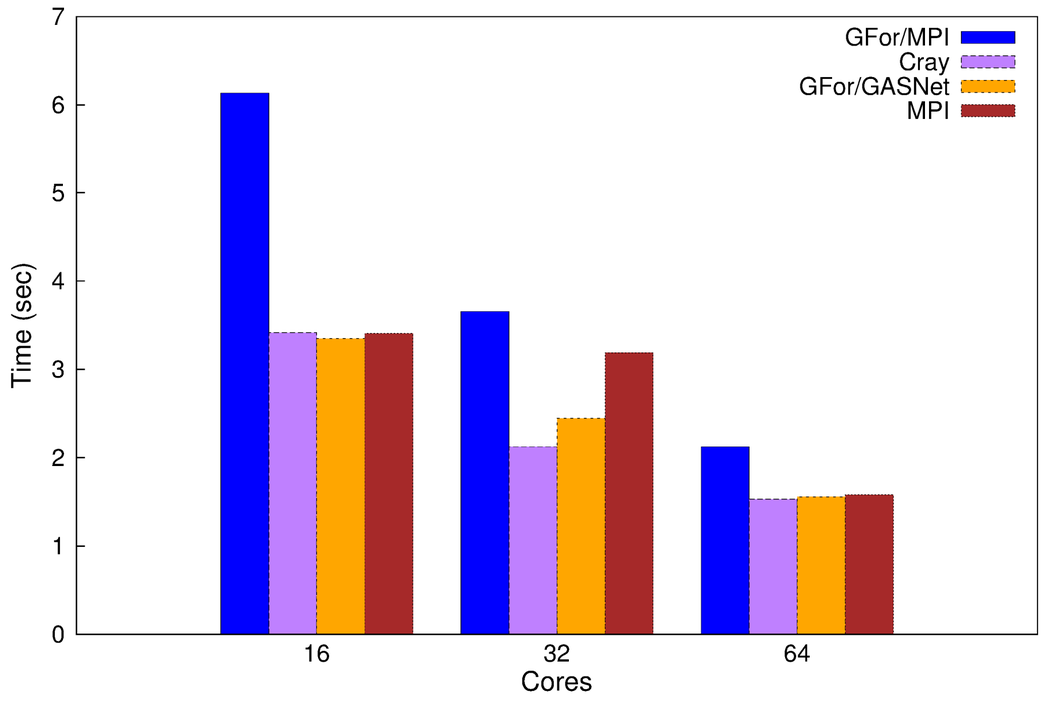
GFortran vs. Intel/Cray





Y and H refer to the Yellowstone and Hopper clusters, respectively.

3D distributed transpose performance



Acknowledgment A special thanks to NCAR for all the support received for this work