The new features of Fortran 2008

John Reid,

Convener ISO Fortran Working Group

Abstract

Following completion of the Fortran 2003 standard, WG5 decided that the next revision would be minor and come out five years later.

A preliminary choice of features was made in 2005 and the final choice in 2007. The draft, see

`ftp://ftp.nag.co.uk/sc22wg5/N1701-N1750/N1723.pdf`

is now out for country comments, with a deadline of 31 August.

We give an overview of the new features.

BCS Fortran
12 June 2008

Coarrays

- SPMD – Single Program, Multiple Data

- Replicated to a number of **images**

- Number of images fixed during execution

- Each image has its own set of variables

- Images mostly execute asynchronously

- Coarrays have second set of subscripts in [ ]
  for access between images

- Synchronization: `sync all`, `sync images`,
  `sync memory`, `allocate`, `deallocate`,
  critical construct

- Intrinsics: `this_image`, `num_images`,
  `co_lbound`, `co_ubound`, `image_index`.

Full summary:

`ftp://ftp.nag.co.uk/sc22wg5/N1701-N1750/N1724.pdf`

Example

```
real :: p[*]
if (this_image()==1) then
   read(*,*) p
   sync all
else
   sync all
   p = p[1]
end if
```

Implementation model

The compiler may arrange that a coarray occupies the same set of addresses within each image. Probably, same executable replicated to each image.

Optimization

Between synchronizations, the compiler can optimize as if the image is on its own, using its temporary storage such as cache, registers, etc.

Recent changes to coarrays

A substantial reduction was proposed by the US at the February meeting and accepted.

It is to separate parallel programming features into a 'core' set that remain in Fortran 2008 while the following features are moved into a separate Technical Report on 'Enhanced Parallel Computing Facilities':

1. The collective intrinsic subroutines.

2. Teams and features that require teams.

3. The `notify` and `query` statements.

4. File connected on more than one image, unless preconnected to the unit specified by `output_unit` or `error_unit`.

It was also decided to remove hyphens from the words 'co-array', 'co-rank', etc., (cf 'cosine' and 'cotangent').

Enhanced module facilities (TR)

If a huge module is split into several modules:

- Internal parts exposed

- Any change leads to compilation cascade

Solution:

- Submodules containing definitions of procedures whose interfaces are in the module itself

- Users continue to access the public parts of the module

- Submodules have full access by host association

We are committed to including this feature in Fortran 2008.

It is described in Metcalf, Reid, and Cohen, OUP.

Major items deleted in 2007

- BITS

- Intelligent macros

Summary of other features

`ftp://ftp.nag.co.uk/sc22wg5/N1701-N1750/N1735.pdf`

Medium items for enhanced performance

- Contiguous attribute

- `DO CONCURRENT`
  Iterations of the loop are independent

Minor technical changes (1)

- Rank plus corank limited to 15.

- Guarantee support of `selected_int_kind(18)`

- A recursive type may be based on allocatable components

- A named array constant may take its shape from its initialization expression

- A pointer may be initialized with a target

- The kind of a `forall` index may be specified in the `forall` header

- An `allocate` statement can give a polymorphic variable the shape and type of another variable without copying the value

- The real and imaginary parts of a complex entity may be accessed with `%re` and `%im` notation

Minor technical changes (2)

- A pointer function reference may appear in a variable-definition context

- `newunit=` available in `open` statement to find a unused unit

- `g0` edit descriptor choses a suitable field width

- `*` for indefinite repetition of edit list

- During processing an i/o statement, i/o to another unit OK in a procedure invoked

- The `block` construct with declarations

- `exit` allowed in any labelled construct

- `stop` code can be any `integer` or `character` initialization expression

- Many (19) intrinsic procedures for bit processing

- `storage_size` returns the size in bits

Minor technical changes (3)

- `selected_real_kind` has optional argument `radix` for specifying the radix

- intrinsics `asin`, `acos`, `atan`, `sinh`, `cosh`, `tanh` extended to complex arguments

- `atan2` may be accessed by the name `atan`

- intrinsics `acosh`, `asinh`, `atanh`

- intrinsics for Bessel functions

- intrinsics for error and gamma functions

- intrinsic `norm2` for careful calculation of Euclidean norm

- intrinsic `parity` tests whether the number of true values is odd

- Optional argument `back` added to `maxloc` and `minloc`

- intrinsic `findloc` to find location of a value in an array

Minor technical changes (4)

- Can execute an external program

- Constants in `iso_Fortran_env` hold kind values

- Can enquire about the compiler and compiler options used

- An empty `contains` section is allowed

- Internal procedure allowed as actual argument

- Allocatable/pointer attribute may be used for generic resolution

- A null pointer that corresponds to a missing dummy argument is interpreted as absent

- Elemental procedures that are not pure

- The `entry` statement becomes obsolescent

Annex. Why coarrays should be part of the Standard

- More rigourous check for wrinkles.

- Experience with TRs has not been happy.

- Coarrays have to be incorporated into the compiler.

- The coarray edits are scattered. Maintaining them separately would not be practical.