

Fortran Standards Update Winter 2012-3

Dan Nagle

CISL/CSG

Welcome

Let me Introduce Myself

- NCAR Representative on the Fortran Committees, US (J3) and International (WG5)
- Chair of J3 since 2002 or so
- Member since 1999
- PhD Computational Science, GMU 2011
- MS Physics, VT 1980
- In CSG since January 2012
- x1299 dnagle@ucar.edu ML 55 K

Overview

- The Further Interoperability with C has been published as ISO/IEC TS 29113:2012
 - Published 2012/11/23 and available for 162 CHF
 - N1942 from www.nag.co.uk/sc22wg5 is free
- TS 18508 Additional Parallel Features in Fortran is in progress
 - Latest draft is 12-201 from www.j3-fortran.org
- WG5: will make a new small revision "f1x"

Interoperability TS

- Standardizes C access to Fortran objects with no close analog in C
 - assumed-shape, allocatable, pointer arrays
 - arrays of derived type components
- In short, C access to Fortran descriptors
- Standardizes Fortran access to C objects with no close analog in Fortran
 - void pointers and arrays without definite rank

Coarrays TS

- Work item approved by SC22 at their September 2012 meeting
- To be done within a year
 - Want a draft for balloting by June 2013
- Work list approved by WG5 at the Markham meeting (June 2012 at IBM Canada Labs)
 - WG5 document N1930
- Bill Long of Cray is the project Editor

Work List Items in N1930

- Accepted:
 - Teams
 - Collective Procedures
 - Atomics
 - Events
- Deferred:
 - Parallel I/O
 - Inter-Image Pointers

Teams

- Restrict the effective image set to a subset
 - References & Collectives
 - Allocate/Deallocate a coarray or call/return of a procedure with a local coarray
 - Synchronization
- Subsets of the whole program can be written independently
- Better performance by mapping to hardware
- Independent library use of coarrays

Team Specifics

- Teams subdivide via a block construct
- An image is a member of one team at any time – image numbers are relative to the team
- A team will be identified by a team variable
 - likely, an opaque derived type in `iso_fortran_env`
- An image can query its image number relative to an ancestor team by some mechanism
- Allocate/Deallocate within the same team

Collectives

- A collective procedure is executed by each image of a set of images within a team
- A collective procedure takes a value from each invoking image and computes a global value
- They are possibly more efficient than might be done portably by a non-expert programmer
 - HW-dependent fan-in/fan-out
- Roughly, they correspond to the similar MPI broadcast/reduction routines

Collective Specifics

- Result variable defined as if by intrinsic assignment (effectively almost synchronous)
- Operate over all images of a team
- Typically, a reduction over a team
 - sum, max, min, user-defined reduction
- Also a broadcast to the team
- Result may go to a specified image, or all images within the team

Atomics

- Low-level but very efficient synchronization mechanism
- Already have `atomic_define` and `atomic_ref`
- New: `atomic_ { cas, add, and, or, xor }`
 - for add, and, or, xor, also "fetch and ..." versions

Events

- More robust replacement for the earlier notify/query mechanism
- Use event variables (which are subject to ordinary scoping rules)
 - to allow libraries to use events without interference

Event Specifics

- Allow one-sided ordering
- Standard opaque derived type in `iso_fortran_env`
- Procedures to post, test, await an event
- Which waiter responds when several are waiting is unspecified

Deferred I/O Proposals

- One proposal is to describe parallel I/O with once-through a file with record granularity
- One proposal is to describe direct access parallel I/O where different records may be read or written by different images
 - How to synchronize?
- No clear winner emerged

Deferred Pointer Proposals

- IBM Inter-Image Pointers
- Rice U CAF Group Copointers
- German (LRZ) Coscalars
- How to synchronize?

Fortran 2015

- a/k/a "f1x"
- Collect the two TSs, many interpretations, editorial improvements, and "wart removal"
- A candidate new feature must
 - fix a deficiency in an existing feature, or
 - fix a discrepancy between two existing features, -and-
 - be easy to describe and easy to implement

f1x Schedule

- Gather features in 2013
 - List set at the June 2013 WG5 meeting
 - Need suggestions for the February 2013 J3 meeting to fit into the US position for WG5
- Final selection of features in 2014
- Final text in 2015
- Publish in 2016

US Wish List – 12-183r5

- Formats: fix g0.d and add a0
- Report non-standard intrinsic modules and non-standard entities in standard modules
- Control host association
- Fix description of dim= on some procedures
- Single statement to declare generic names
- Scope (public/private) of enumerators
- Require specification of external attribute

TS 24772

- WG23 Programming Language Vulnerabilities
- <http://grouper.ieee.org/groups/plv/>
- Guidance only – no normative text
 - N0410 is the latest draft of 24772
- Main text is language-independent
- Annexes are language-specific
- WG5 N1947 is a draft of the Fortran annex

Your Input

- Give me your suggestions for the further coarrays TS or for the f1x revision
 - Let's talk them over so I understand them and so I can tell you whether they fit
- Any ideas must be small
- and f1x ideas must fix something
 - That is, not a wholly new idea
- x1299 dnagle@ucar.edu ML 55 K

Fortran Standards Update Winter 2012-3

Dan Nagle

CISL/CSG

Thank-you