



ISO/IEC JTC 1/SC 22/WG 5 "Fortran"  
Convenorship: ANSI  
Convenor: Lionel Steve Mr



## Results of WG Consultation on Conditionally Approved Work Items for F202Y

Document type	Related content	Document date	Expected action
Recommendation 2025		2025-05-04	

# Results of WG Consultation on Conditionally Approved Work Items for F202Y

N2239

May 3, 2025

At the 2024 meeting, three prospective work items were "conditionally accepted". INCITS/Fortran was requested to discuss these at a future meeting to provide direction to WG5 - this happened at the February 2025 meeting in Berkeley, California. Given the abbreviated (and end-of-week) schedule for the 2025 WG5 meeting, a request was made to provide INCITS/Fortran's guidance and to ballot WG5 members on formally accepting the work items so that development could start earlier. A WG Consultation was opened April 6, 2025 and closed May 3, 2025. 26 WG5 members voted; the results and comments are given below.

**US20** - Add SCAN and CO\_SCAN intrinsic procedures. (The names would likely change.) See [j3-fortran.org/doc/year/23/23-235r2.txt](http://j3-fortran.org/doc/year/23/23-235r2.txt) for more details. The INCITS/Fortran vote on this was 11 in favor, 0 opposed. Comments made at the time: These are difficult for users to implement on their own with acceptable performance. C++ already has these, and some vendors have already implemented them for their own use.

20 Yes, 5 No, 1 Undecided

Comments:

- N: CO\_SCAN in particular is not part of C++; its likely benefit is too small to outweigh even a small cost.  
SCAN was part of HPF, and several vendors implemented it, and the user appreciation was so underwhelming that they have all dropped it since.
- N: Rather than as intrinsics, scan/co\_scan might better be supported as part of a standard template-generic library for Fortran.

- U: While this might map directly onto an (obsolete) HPF construct, there hasn't been a compelling use case yet.
- N: It is more important to adhere to the schedule in N2232 than to add an item of debatable widespread utility.
- N: Insufficient progress has been made on the present list of work items. This item could be added after the removal of a large item from the present list.
- N: There are too many items on the existing work list. That said, SCAN/CO\_SCAN could be easily implemented.

Convenor's interpretation: Direction is to put US20 on the approved work list, but WG5 could vote later to remove it as development proceeds.

## US22/DIN02 - Unions in interoperable types. See

[j3-fortran.org/doc/year/24/24-117.txt](http://j3-fortran.org/doc/year/24/24-117.txt) It was pointed out that trying to emulate this feature with independently declared types and TRANSFER loses the feature of automatically sizing the union to the maximum of the overlaid types. Most compilers already support unions using a nonstandard syntax. The INCITS/Fortran vote on this was 7 in favor, 0 opposed, 4 undecided.

20 Yes, 5 No, 1 Undecided

Comments:

- N: Not a high priority and potentially risky
- N: These are inherently unsafe. Fortran does not need more unreliability.
- N: The interoperability of Fortran intrinsics require a much more detailed semantics of the bit-patterns of said types than given in the Fortran standard.  
Specifically: KIND arguments to intrinsics are currently not standardised across processors. Using union types without such standardisation will yield non-portable code, going against the purpose of standardising a union type.
- N: The feature is reported to be unsafe.
- U: I find some arguments for this work item to be persuasive, as well as some arguments against this work item. Hence, I am undecided at this time.

- Insufficient progress has been made on the present list of work items. This item could be added after the removal of a large item from the present list.
- Y: I hate to extend use of equivalence association, but take this as a necessary evil.
- Y: Vendors have already implemented this feature for DEC compatibility.

Convenor's interpretation: Direction is to put US22/DIN02 on the approved work list. Most if not all compilers already support this with different syntax and it is a request frequently seen from users.

**DIN01** - Collectives over a specified team. See N2230.pdf The INCITS/Fortran vote on this was 10 in favor, 0 opposed, 1 absent

23 Yes, 3 No, 0 Undecided

Comments:

- N: It is important to adhere to the schedule in N2232. A review of all the items in N2234 would be desirable.
- N: Insufficient progress has been made on the present list of work items. This item could be added after the removal of a large item from the present list.
- N: Same comment as for SSCA/CO\_SCAN although I am less sure of the ease of implementation.

Convenor's interpretation: Direction is to put DIN01 on the approved work list.