

## Proposed edits to N1265 Fortran vulnerabilities after meeting 27 Feb 2023

### 4.9 Polymorphism

In para 3, line 1, after “dynamic type that is permitted to be” add “the declared type or”.

#### 4.10.6 Asynchronous variables

In final line change “packages such as MPI that have procedures” to “packages such as MPI which has the procedures `MPI_Irecv` and `MPI_Isend`” and delete comment.

### 6.4 Floating-point arithmetic [PLF]

Delete comment – I have reread 6.4 and found nothing special for Fortran.

#### 6.47.2 Avoidance mechanisms

Submitted by S. Lionel

Add an additional bullet: Be aware that certain Fortran dummy arguments interoperate with a "C descriptor", a Fortran standard concept. These will require special code in the other language procedure to properly receive or pass the argument.

#### 6.57.1 Applicability to language

Delete para 3 – it is redundant with para 4.

#### 6.58.1 Applicability to language

In line 2, change “1950’s” to “1950s” – plural, not possessive.

#### 6.61.1 Applicability to language

Replace “There are several mechanisms (see clause 4.10) ... alternative action if some have failed. “ and the comments by

3. By using invoking a procedure that has an image selector in square brackets or invokes a collective procedure.

The following mechanisms are available for separating the alteration of the value of a variable on one image from its access by another image:

1. The `sync all` and `sync images` statements (clause 4.10.1).
2. Events (clause 4.10.1).
3. The `critical` construct (clause 4.10.1).
4. Locks (clause 4.10.2).
5. Teams (clause 4.10.3).
6. Collectives (clause 4.10.8).

All changes of values of atomic variables (clause 4.10.8) occur sequentially.

#### 6.61.2 Avoidance mechanisms for language users

Delete comment. This is one of the mechanisms (a rather poor one) and does not deserve special attention.

#### 6.62.1 Applicability to language

Replace “(processes)” by “(clause 4.10.9)”, delete “See clause 4.8 for ... image halt or a continuation” and delete the comment – the concept of the failed image was added to Fortran explicitly for the sake of massively parallel systems.

## 6.62.2 Avoidance mechanisms for language users

Add bullet

- If a procedure needs to abort, do not execute a `stop` statement – instead return with an error flag set.

In bullet 2, delete “, stopped\_images,” and “and stopped”.

## 6 63.1 Applicability to language

At the end of the first sentence add “if each image is regarded as a thread” and replace “4.10” by “6.61.1”. Delete comment.

## 6.64.1 Applicability to language

In the first line change “do” to “does”.

## 6.65.1 Applicability to language

Delete comment.

## 7.1 Source form

Delete comment.

### 7.1.1 Applicability to language

In line 1 change “permits a” to “has an obsolescent”.

In line 5 change “is” to “being”.

After 7.1 add new clause

## 7.2 Unformatted files

### 7.2.1 Applicability to language

In Fortran unformatted output of a variable or expression, the internal representation of its value is written exactly as it stands to the storage medium and can be read back directly with neither roundoff nor conversion overhead into a variable of the same type, type parameters, and shape. If the variable is a pointer, it must be associated with a target and the target is transferred; when read back the target must have the shape of the target that was written. If the variable is allocatable, it must be allocated; when read back it must be allocated and have the shape of the variable that was written. The variable is not permitted to be of a type with an ultimate component that is allocable or a pointer.

If the file is read within a program execution other than the one in which it was written, there is a danger that incorrect values will be obtained.

### 7.2.2 Avoidance mechanisms for language users

When using an unformatted file:

- Ensure that the properties of each variable read exactly match those of the variable or expression that was written.
- Limit access to the same computer system, the same compiler, and the same compiler options unless it is certain that the same internal representations are in use.