

Subject: Allow more general forward type reference for components  
From: Van Snyder

1 **Number**

2 TBD

3 **Title**

4 Allow more general forward type reference for components.

5 **Submitted By**

6 J3

7 **Status**

8 For consideration.

9 **Basic Functionality**

10 Allow a forward reference for the type of a component, so long as any circular dependence that arises  
11 involves a component with the POINTER or ALLOCATABLE attribute.

12 **Rationale**

13 It's a little bit weird that the appearance of the POINTER attribute allows a forward reference for  
14 the type of a component, but ALLOCATABLE doesn't. That the constraint wasn't updated when  
15 allocatable components were done was probably an oversight rather than a conscious decision.

16 I think the only reason for the existence of C438 was to prevent infinite-size types — those that have  
17 themselves as a direct component (as defined below). Prohibiting a forward reference was a simple way  
18 to do this — but it also prevents orderings of type definitions that some would find clearer than the  
19 reverse of a depth- or breadth-first traversal of the dependency DAG.

20 Processors have to cope with forward type references for FUNCTION statements, so that's clearly not  
21 an unsolved problem. Processors have to prevent circular module dependencies, so circular-reference  
22 detection also is clearly not an unsolved problem.

23 **Estimated Impact**

24 Trivial for the standard, probably in the trivial-to-small range for processors.

25 **Detailed Specification**

26 The **Basic Functionality** section covers it. Here are proposed edits, to give an idea of the effect on the  
27 standard.

28 The **direct components** of a derived type are the nonpointer nonallocatable components of that type, 44:29+ New ¶  
29 and the direct components of the types of its components that are of derived type.

30 

This definition of <b>direct components</b> is different from (and simpler than) the definition in 01-007.
--

*Note to J3*

31 [Replace C438 and C439:] 50:18-23

32 C438 (R440) A derived type shall not have a direct component of the same declared type.

33 **History**