

Subject: First draft of edits for conditional forms
 From: Van Snyder
 Reference: 04-393r1 04-410r1 05-103 05-104

1 **1 Introduction**

2 Assuming serialized AND/OR, conditional expressions, and computing whether an actual argument
 3 corresponding to an optional dummy argument is present get onto the J3 work plan, the reason for this
 4 paper is to get a running start on the edits.

5 **2 Edits**

6 Edits refer to 04-007. Page and line numbers are displayed in the margin. Absent other instructions, a
 7 page and line number or line number range implies all of the indicated text is to be replaced by associated
 8 text, while a page and line number followed by + (-) indicates that associated text is to be inserted after
 9 (before) the indicated line. Remarks are noted in the margin, or appear between [and] in the text.

or conditional-form

117:22+

11 **7.1.1.6 $\frac{1}{2}$ Conditional forms**

120:13-

12 A **conditional form** is an entity that has the same syntax and precedence as a function reference. It
 13 is not, however, a function reference. The primary difference is that the arguments are not evaluated
 14 unless and until they're needed.

15 R721 $\frac{1}{2}$ *conditional-form* is *conditional-form-name* (*actual-arg-spec-list*)

16 C703 $\frac{1}{4}$ (R721 $\frac{1}{2}$) The *conditional-form-name* shall be the name of one of the conditional forms listed in
 17 Table 7.0 $\frac{1}{3}$.

18 C703 $\frac{1}{2}$ (R721 $\frac{1}{2}$) An *actual-arg* shall not be an *alt-return-spec*.

19 C703 $\frac{3}{4}$ (R721 $\frac{1}{2}$) A two-argument IF conditional form shall not appear except as an actual argument
 20 corresponding to an optional dummy argument.

Table 1: Conditional forms

Name	Purpose
ANDTHEN	Evaluate arguments, stopping if one is false
ORELSE	Evaluate arguments, stopping if one is true
IF (three arguments)	The result is either the second or third argument, depending on whether the first is true
IF (two arguments)	The second argument of IF is the actual argument of the referenced procedure if the first argument of IF is true, else the actual argument of the referenced procedure is absent

21 **7.1.1.6 $\frac{1}{2}$.1 ANDTHEN (A1, A2 [, ...])**

22 **Description.** The arguments are evaluated in order until one is found to be false, or all
 23 arguments are evaluated. When an argument is found to be false, succeeding arguments are not
 24 evaluated.

25 **Class.** Elemental conditional form.

26 **Arguments.** The arguments shall all be of type logical.

27 **Result Characteristics.** The type of the result is logical. The value of the kind type parameter
 28 of the result is processor dependent, but is the same as for one of the arguments.

29 **Result Value.** The result value is true if all the arguments are true, else it is false.

30 **Example.**

```

1      ...
2      LOGICAL, OPTIONAL, INTENT(IN) :: LOGBASIS
3      IF ( ANDTHEN(PRESENT(LOGBASIS),LOGBASIS) ) RESULT = LOG(RESULT)
4      ...

```

5 **7.1.1.6¹/₂.2 ORElse (A1, A2 [, ...])**

6 **Description.** The arguments are evaluated in order until one is found to be true, or all arguments are evaluated. When an argument is found to be true, succeeding arguments are not evaluated.

9 **Class.** Elemental conditional form.

10 **Arguments.** The arguments shall all be of type logical.

11 **Result Characteristics.** The type of the result is logical. The value of the kind type parameter of the result is processor dependent, but is the same as for one of the arguments.

13 **Result Value.** The result value is false if all the arguments are false, else it is true.

14 **7.1.1.6¹/₂.3 IF (COND, A, B)**

15 **Description.** COND is evaluated. If its value is true, the result is A, else the result is B. Neither A nor B is evaluated unless and until the value of the result of the conditional form is needed. No part of the argument that is not selected is evaluated.

18 **Class.** Transformational conditional form.

19 **Arguments.**

20 COND shall be of type logical.

21 A may be a data entity of any type, a reference to the NULL intrinsic function, or a procedure designator.

22 B may be a data entity of any type, a reference to the NULL intrinsic function, or a procedure designator.

23 The relation between the characteristics of A and B shall be as shown in Table 7.0²/₃.

Table 2: Relation of characteristics of A and B

Characteristics of A	Characteristics of B		
	NULL()	Data entity or NULL(MOLD ¹)	Procedure designator or NULL(MOLD ²)
NULL()	Allowed	Allowed	Allowed
Data entity or NULL(MOLD ¹)	Allowed	A and B shall have the same type, kind type parameter values, and rank. B shall be allocatable if and only if A is allocatable	Prohibited
Procedure designator or NULL(MOLD ²)	Allowed	Prohibited	The characteristics of A and B as procedures shall be the same.
Notes: 1: MOLD is a data pointer		2: MOLD is a procedure pointer	

24 If NULL(MOLD) were defined to be a data or procedure entity, depending on MOLD, the column/row headings would be simpler, and the notes wouldn't be necessary.

J3 Remark

25 **Result Characteristics.**

26 *Case (i):* A and B are both references to the NULL() intrinsic function without arguments.
 27 The result characteristics are inferred from the context of the appearance of the
 28 conditional form exactly as would be the characteristics of the result of NULL

1 (13.7.88).

2 *Case (ii):* A and B are data entities. The declared type, kind type parameter values, and
3 rank of the result are the same as those of the argument that is not a reference to
4 the NULL intrinsic function with no arguments. The result has the POINTER
5 attribute if and only if both A and B are pointers.

6 *Case (iii):* A or B is a procedure designator. The characteristics of the result are the same as
7 the characteristics of the argument that is not a reference to the NULL intrinsic
8 function with no arguments.

9 **Result.** If the value of COND is true, the result is A, else the result is B. Neither A nor B is
10 evaluated unless and until the value of the result of the conditional form is needed. No part of
11 the argument that is not selected is evaluated. If the selected argument is a pointer and the
12 other is not, the selected argument shall be associated with a target and result is that target.
13 If the selected argument is a reference to the NULL intrinsic function and the other argument
14 is allocatable, the result is a deallocated allocatable entity.

NOTE 7.6 $\frac{1}{2}$

Since the result of IF is either A or B, not the value of A or B, if a reference to IF appears in a variable definition context, the A and B arguments of IF shall be permitted in a variable definition context. If a reference to IF is the target in a pointer assignment, the A and B arguments of IF shall be suitable as targets in that assignment. The case where A or B is allocatable and the other is a reference to the NULL intrinsic function is only useful if the conditional form corresponds to an allocatable component in a structure constructor or if it corresponds to an allocatable dummy argument that is neither referenced nor allocated.

15 Examples.

```
16 X = X + IF ( Y > 0.0, SQRT(Y), 0.0 )
17 PROC => IF ( ANALYTIC, ANALYTIC_DERIV_FUNCTION, NUMERIC_DERIV_FUNCTION )
```

18 7.1.1.6 $\frac{1}{2}$.4 IF (COND, A)

19 **Description.** COND is evaluated. If its value is true, the result is A, else the result is an absent
20 actual argument. Neither A nor any expression within it is evaluated if COND is false.

21 **Class.** Transformational conditional form.

22 Arguments.

23 COND shall be of type logical.

24 A may be a data entity of any type, a reference to the NULL intrinsic function,
25 or a procedure designator.

26 **Result Characteristics.** The same as the characteristics of A.

27 **Result.** If the value of COND is true, the result is A, else the result is an absent actual
argument.

NOTE 7.6 $\frac{1}{2}$

Since the result of IF is either A or absent, not the value of A, if a reference to it corresponds to a dummy argument that has INTENT(OUT) or INTENT(INOUT), A shall be a variable.

28 Example.

```
29 CALL MODEL ( A, B, IF ( PRESENT(C), C(I,:) ) )
```