Subject: Another whack at embedding conditionals in expressions
From: Van Snyder
Reference: 03-258r1, section 2.8.1; 04-192

Number

TBD

Title

Another whack at embedding conditionals in expressions.

Submitted By

J3

Status

For consideration.

Basic Functionality

Allow to embed decisions within expressions.

Rationale

See 04-192.

Estimated Impact

Small, both for standard and implementors.

Detailed Specification

Provide two new intrinsic functions, named IF here but the particular names are not important. In both cases, the first argument is of type logical, and is evaluated before the function is "invoked." In the three-argument case, the result is the second argument if the first is true, and the third argument if the first is false.

In the two-argument case, a reference to which is permitted only as an actual argument associated with an optional dummy argument, the result is the second argument iff the first is true, else it is an absent actual argument.

Notice that the specification carefully specifies "the result is ...," not "the result is the value of ...." For all other functions, the result is an entity distinct from its arguments. For these functions, the result *is* one of the arguments. The "functions" behave more like run-time macro substitutions than functions.

Illustrative edits w.r.t. 04-007

C1220$\frac{1}{2}$ (R1217) A reference to the two-argument form of the IF intrinsic function shall not appear except as an actual argument corresponding to an optional dummy argument. 266:16+

[Replace "it" by "any function other than the IF intrinsic function (13.7.51$\frac{1}{2}$)".] 276:3

13.5.17$\frac{1}{2}$ Conditional functions 298:2+

IF (MASK, TSOURCE, FSOURCE)	Result is TSOURCE or FSOURCE, depending on MASK.

IF (MASK, TSOURCE)	Result is TSOURCE if MASK is true, else result is an absent actual argument.

13.7.51½ IF (MASK, TSOURCE, FSOURCE) or IF (MASK, TSOURCE) 322:23+

Description. Embed a decision within an expression, or calculate whether an actual argument is present.

Class. Transformational.

Arguments.

MASK	shall be of type logical and shall be scalar.
TSOURCE	may be of any type, and may have any type parameter values. Shall be TKR compatible (5.1.1.2) with FSOURCE. It is not evaluated before the function is invoked. It may be undefined. If it is a pointer it need not be associated. If it is allocatable it need not be allocated.
FSOURCE	shall be TKR compatible with TSOURCE. It shall be polymorphic if and only if TSOURCE is polymorphic. It is not evaluated before the function is invoked. It may be undefined. If it is a pointer it need not be associated. If it is allocatable it need not be allocated.

Result Characteristics.

Case (i): Three arguments: The result characteristics are the same as TSOURCE if MASK is true, else the same as FSOURCE.

Case (ii): Two arguments: The result characteristics are the same as TSOURCE if MASK is true, else the result is an absent actual argument.

Result.

Case (i): Three arguments: The result is the TSOURCE argument if the MASK argument is true, else it is the FSOURCE argument. The result, and therefore the function reference, may appear in a variable-definition context (16.5.7) if TSOURCE and FSOURCE are permitted to appear in a variable-definition context.

Case (ii): Two arguments: The result is the TSOURCE argument if and only if the MASK argument is true. If MASK is false the result is undefined, and the actual argument consisting of the function reference is absent. The result, and therefore the function reference, may appear in a variable-definition context (16.5.7) if TSOURCE is permitted to appear in a variable-definition context.

Examples.

Case (i): The result of IF (PRESENT(X), X, 0.0) is X if X is present, else it is 0.0.

Case (ii): The result of IF (ASSOCIATED(P), P(::2), NULL()) is the array section P(::2), which is not a pointer, if P is associated, and NULL(), which is a pointer, if P is not associated. Both are valid targets in a pointer assignment.

Case (iii): The result of IF (ASSOCIATED(P), P(::2)) is a present actual argument that is the array section P(::2) if P is an associated pointer, else it is an absent actual argument.

Case (iv): The result of IF (PRESENT(D), D(:,J)) is a present actual argument consisting of the array section D(:,J) if D is a present dummy argument, else it is an absent actual argument.

History