Subject: EXIT from any labeled construct
From: Van Snyder
Reference: 03-258r1, section 2.1.2, 04-109, 04-156r1, 04-395r1, WG5/N1626-J3-024

1   Detailed Specification

Allow an EXIT statement with a *do-construct-name* that is the same as the name of any enclosing construct, not just a DO construct. When the EXIT statement is executed, execution of the named construct is terminated. For compatibility, an EXIT statement without a *do-construct-name* continues to refer to the nearest enclosing DO construct.

2   Editorial strategy

Replace *do-construct-name* in R844 with *construct-name*. Allow it to be the name of any construct that encloses the EXIT statement. Add a new subclause 8.1.7 that describes the EXIT statement but not loop termination. Specify there that the EXIT applies to the construct named by the *construct-name*. Do not change the interpretation of an EXIT statement that doesn't mention a *construct-name*.

3   Edits

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

[Editor: Replace **8.1.6.4.4 Loop termination**:]   168:3-23

8.1.6.4.4 Loop termination

A loop terminates, and the DO construct becomes inactive, when any of the following occurs:

   (1)   Determination that the iteration count is zero or the *scalar-logical-expr* is false, when tested
         during step (1) of the above execution cycle
   (2)   Execution of an EXIT statement that belongs to the DO construct
   (3)   Execution of an EXIT statement or a CYCLE statement that is within the range of the DO
         construct, but that belongs to an outer construct
   (4)   Transfer of control from a statement within the range of a DO construct to a statement that
         is neither the *end-do* nor within the range of the same DO construct
   (5)   Execution of a RETURN statement within the range of the DO construct
   (6)   Execution of a STOP statement anywhere in the program; or termination of the program
         for any other reason.

When a DO construct becomes inactive, the DO variable, if any, of the DO construct retains its last defined value.

[Editor: insert a new subclause immediately before 8.2:]   169:1-

8.1.7 EXIT statement

The EXIT statement provides one way of terminating a construct.

R844   *exit-stmt*   **is**   EXIT [*construct-name*]

C829   (R844) If an *exit-stmt* refers to a *construct-name*, it shall be within the range of that construct;
         otherwise, it shall be within the range of at least one *do-construct*.

An EXIT statement belongs to a particular construct. If the EXIT statement refers to a construct name, it belongs to that construct; otherwise, it belongs to the innermost DO construct in which it appears.

When an EXIT statement that belongs to a DO construct is executed, it terminates the loop (8.1.6.4.4) and any active loops contained within the terminated loop. When an EXIT statement that belongs to

1   a non-DO construct is executed, it terminates any active loops contained within that construct, and
2   completes execution of that construct.

3   [Editor: Delete the first "DO" from the definition of **belong** in the glossary.]   426:8