Subject: First draft of edits for optional dummy default values
From: Van Snyder
Reference: 04-386r2

# 1 Introduction

Assuming default initial values for optional dummy arguments, J3 repository item 017, get onto the J3 work plan, the reason for this paper is to get a running start on the edits.

Upon reflection, subgroup decided to simplify the specification. The specification in 04-386r2 causes optional dummy arguments that have initial values to behave as though they have the VALUE attribute, with most of the C527 restrictions removed, i.e., the initial value of the dummy argument is that of the initializer. It would be simpler and more useful if the initializer behaved more like a replacement actual argument. In particular, this would remove the restriction on INTENT. It is not entirely the same as an actual argument, in that PRESENT still returns false if there is no associated actual argument or the associated actual argument is absent.

Somewhat different terminology from *initialization* may ultimately be desired.

# 2 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: "*initialization-expr*" ⇒ "*expr*".] 72:15

---

[Editor: Delete "a dummy argument,".] 73:11

---

C524$\frac{1}{3}$ (R506) If *object-name* is not a dummy argument, *expr* shall be an initialization expression. If *object-name* is a dummy argument, *expr* shall be a restricted expression. 73:14+

C524$\frac{2}{3}$ (R504) If *initialization* appears and *object-name* is a dummy argument, OPTIONAL shall be specified and *initialization* shall meet the requirements for an actual argument associated with *object-name* (12.4.1).

---

[Editor: "does not have" ⇒ "is neither a dummy argument nor has".] 74:24

---

[Editor: Insert ", an *object-name* that is a dummy argument," after "block".] 74:36

---

If *initialization* is specified for an optional dummy argument that is not present (12.4.1.6), upon entry to the procedure the effect is as if *initialization* were associated with the dummy argument as an actual argument, but *initialization* is not an actual argument and the dummy argument does not become present. 83:11+

---

(2) An object designator with a base object that is a dummy argument that has initialization, or that has neither the OPTIONAL nor INTENT(OUT) attribute, 125:14-15

---

[Editor: "specification" ⇒ "restricted" thrice.] 126:7,9,14

---

If a restricted expression is the initialization for a dummy argument, and it depends upon the value of a dummy argument that is specified in the same *specification-part* and has initialization, the initialization for the dummy argument upon which the restricted expression depends shall be specified in a prior specification of the *specification-part*. The prior specification may be to the left of the restricted expression in the same statement. 126:19+

---

[Editor: Insert "and does not have initialization" after "present".] 272:31