

**Proposal for C2Y**  
**WG14 N3492**

**Title:** Improved treatment of error conditions for functions that round result to narrow type (updates N3404)  
**Author, affiliation:** CFP group  
**Date:** 2025-02-11  
**Proposal category:** Editorial  
**Reference:** N3435, N3404

This update to N3404 is intended to address comments by Christopher Bazley in [SC22WG14.28504].

This document proposes clarification of the language and unification of the requirements around domain errors for functions that perform arithmetic and deliver results to a narrower floating type.

**Suggested change:**

`fadd` – change to the last sentence of 7.12.15.2#2

A domain error may occur ~~for infinite arguments~~ if both  $x$  and  $y$  are infinite and have opposite sign.

`fsub` – change to the last sentence of 7.12.15.3#2

A domain error may occur ~~for infinite arguments~~ if either both  $x$  and  $y$  are infinite with the same sign or both  $x$  and  $y$  are infinite and unsigned.

`fmul` – change to the last sentence of 7.12.15.4#2

A domain error ~~occurs~~ may occur ~~for one infinite argument and one zero argument~~ if one of  $x$  and  $y$  is infinite and the other is zero.

`fdiv` – change to the second to the last sentence of 7.12.15.5#2

A domain error ~~occurs~~ may occur ~~for~~ if either both ~~arguments~~  $x$  and  $y$  are infinite or both ~~argument~~  $x$  and  $y$  are zero.

`fma` – change to the last sentence of 7.12.15.6#2

A domain error may occur ~~for an infinite argument~~ if one of  $x$  and  $y$  is zero and the other is infinite. A domain error may occur if  $z$  is infinite and one of  $x$  and  $y$  is infinite while the other is nonzero finite or infinite, such that the product of  $x$  and  $y$  has the opposite sign of  $z$ .

**Rationale:**

This change makes explicit which cases may cause domain errors in `fadd`, `fsub`, and `fma`. The case of `fsub` is expanded to include unsigned (projective) infinity, should it be implemented, in the one case of magnitude subtraction of unsigned infinite values.

The change also brings the reporting requirements of all functions in 7.12.15 into harmony by weakening the specification for `fmul`, and `fdiv` from "error occurs" to "error may occur". The alternative suggestion, to strengthen all reporting to "error occurs", could break some existing implementations.

This change requires no implementation changes.