

Document: n3368

Author: Martin Uecker & David Svoboda

Date: 2024-10-02

This is an update of n3343, with the main change being grammar fixes in the normative text.

This remaining issues from n3244 were split up into the following series:

n3340 - Slay Some Earthly Demons II

n3341 - Slay Some Earthly Demons III

n3342 - Slay Some Earthly Demons IV

n3343 - Slay Some Earthly Demons V

n3344 - Slay Some Earthly Demons VI

n3345 - Slay Some Earthly Demons VII

n3346 - Slay Some Earthly Demons VIII

n3347 - Slay Some Earthly Demons IX

Meeting notes about previous discussion can be found in n3281. The following numbers correspond to N3220 and the ones with (→) to N3301.

Acknowledgments: Thanks to Joseph Myers, David Svoboda, Chris Bazley, and the UB study group for helpful comments and corrections.

Analysis

Several paragraphs use “shall” in a semantics section (outside of a constraints section) without expressing a requirement and without implying undefined behavior (cf 4 Conformance).

Proposed Change (remove incorrect use of shall)

6.7.4.1

Semantics

11 ~~For~~ Two qualified types are to be compatible ~~both shall have the~~ if and only if they are identically qualified versions of a compatible types; the order of type qualifiers within a list of specifiers or qualifiers does not affect the specified type.

6.7.7.4

Semantics

14 ~~For~~ Two function types are to be compatible ~~both shall~~ if and only if all of the following hold:

- They specify compatible return types.
- ~~Moreover,~~ The parameter type lists ~~shall~~ agree in the number of parameters and in use of the final ellipsis.
- The corresponding parameters ~~shall~~ have compatible types.

6.7.7.3

Semantics

6 ~~For~~ Two array types are to be compatible, ~~both shall~~ if and only if both of the following hold:

- They have compatible element types
- ~~, and i.~~ If both size specifiers are present and are integer constant expressions, ~~then both size specifiers shall~~ they have the same constant value.

If the two array types are used in a context which requires them to be compatible, the behavior is undefined if the two size specifiers evaluate to unequal values.