

Slay Some Earthly Demons V

Document: n3343

Author: Martin Uecker

Date: 2024-09-08

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

- J.2.19 needs more work, cf. SC22WG14.25600 (**note to editor:** parts applied by mistake)
- J.2.21 done (incl. additional change), (**note to editor:** there is a “not“ missing in 6.3.2.1)
- n3340 J.2.40 (→ J.2.38) withdrawn in June, now revised (cf. SC22WG14.25600)
- J.2.56 done
- J.2.57 done (option 1)
- n3341 J.2.58 (→ J.2.54) not discussed in June
- J.2.60 (→ J.2.56) direction to editor (**note to editor:** not yet done)
- n3342 J.2.63 (→ J.2.59) no consensus in June, now revised to make implementation-defined
- n3343 wording change not related to an entry in Annex J**
- J.2.67 done
- J.2.69 done (alternative wording for semantic section)
- n3344 J.2.75 (→ J.2.70) now revised (cf. SC22WG14.25600)
- J.2.77 (→ J.3.13. 2) not discussed in June, but fixed editorially
- J.2.78 (→ J.3.13. 3) not discussed in June, but fixed editorially
- n3345 J.2.79 (→ J.2.72) not discussed in June
- n3346 J.2.80-J.2.82 (→ J.2.73-J.2.75) not discussed in June (cf. SC22WG14.25600)
- n3347 J.2.87 (→ J.2.80) not discussed in June

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 they are~~ identically qualified versions of a compatible type; 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 they~~ specify compatible return types. ~~Moreover,~~ the parameter type lists ~~shall~~ agree in the number of parameters and in use of the final ellipsis, ~~and the~~ corresponding parameters ~~shall~~ have compatible types.

6.7.7.3

Semantics

6 ~~For~~ Two array types ~~are to-be~~ compatible, ~~both-shall if they~~ have compatible element types, and if both size specifiers are present and are integer constant expressions, ~~then-both-size-specifiers-shall the size specifiers~~ 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.

6.7.7.4

Semantics

14 ~~For~~ Two function types ~~are to-be~~ compatible, ~~both-shall if they~~ specify compatible return types. ~~Moreover,~~ ~~and if the~~ the parameter type lists ~~shall~~ agree in the number of parameters and in use of the final ellipsis, ~~and if the~~ corresponding parameters ~~shall have~~ compatible types. ...