

Operators that cannot be Overloaded – WHY?

Source: <http://coding.derkeiler.com/Archive/C/ CPP/comp.lang.cpp/2004-10/4036.html>

From: Brad Eck (brad.eck_at_sitesdynamic.com)

Date: 10/27/04

Date: 26 Oct 2004 16:18:28 -0700

"The only operators that cannot be overloaded are :: (scope resolution), . (member selection), and .* (member selection through pointer to function). Quoting from Stroustrup's 3rd edition of *The C++ Programming Language*, section 11.2 (page 263), these three operators 'take a name, rather than a value, as their second operand and provide the primary means of referring to members. Allowing them to be overloaded would lead to subtleties.'"

First, I've seen ?: and sizeof added to this list. Someone also mentioned 'delete' but Stroustrup doesn't refer to it as one. In fact, it's in the list of overloadable operators.

Second, what is meant by 'take a name'? What is meant by 'subtleties'? Any thoughts as to the detailed reasoning behind these?