

---

Subject: Re: overloading operators rules

Posted by [piotr5](#) on Mon, 11 May 2009 14:47:34 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

I think a third rule would be to watch out for operator-priority-list. it wouldn't make much sense to define an operator which in combination with other operators needs to be put into brackets all the time. for example in boolean logic "\*" is used for and, "+" is used for or, even though the two could be used the other way around, simply because it is more convenient to have a tighter binding between 2 things which live in an and-condition. (for example (a&b)|(c&d) is more natural for us to understand than (a|b)&(c|d)...)

another interesting rule which stl seems to obey is that some operators must have symmetric parameters-types. some algorithms of stl are only defined when both parameters of the operator are of the same type. I don't remember where I had this problem though, but I think it had something to do with the "<" operator...

---