
Subject: Re: IDL 8.0 compile_opt changes
Posted by [penteado](#) on Sat, 19 Dec 2009 16:16:33 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Dec 19, 8:19 am, "H. Evans" <blogs...@googlemail.com> wrote:
> Another thought, will the new syntax contain more information about
> the code itself (whether it is an array
> or a method, etc.) it should be easier to downgrade new code to the
> older syntax, shouldn't it? To me, the best formed language doesn't
> overload its syntax.
>
> In the new syntax the '.' would be overloaded, so some clever parsing
> would be required to determine if the '.' was a method call (changed
> to '->') or a structure field (leave it as '.').
>
> Hmmmmm...is this really an improvement? Would we not be muddying the
> waters with ambiguity again? It took long enough to move from the
> overloading of the '()' where it was for both array subscripting and
> function parameter delimiting.
>
> H.

It would only be ambiguous when the idl2 option is not used (explicitly or by default). With it, a.b is only a structure field of there are [], or it is being used as a variable. A function would require the (), and a routine could not appear in an expression.

Which is why idl2 should be the new default, either always (option #1), or based on the file extension (option #5). That way, it would also be possible to have a translator that would generate code compatible with older versions, since it would find no ambiguities.
