Subject: Re: A simple DLM question
Posted by Xin Tao on Wed, 29 Aug 2012 16:23:51 GMT
View Forum Message <> Reply to Message

That's very clear explanation, thanks. But I would still say that it's not very intuitive to me that 3 is a constant but -3 is not.

On Wednesday, August 29, 2012 10:55:28 AM UTC-5, Chris Torrence wrote:
> Hi all,
>
>
>
> Just to clarify, this isn't a bug. It just happens that since the minus sign is an operator, then a number such as -1 gets turned into an expression, which is stored in a temporary variable. It is treated the same way as say "-a" where "a" is a variable.
>
>
>
> Jimmy said "The macro (in idl_export.h, if you're interested) doesn't do extensive checking, and you should only free variables that are temps, not expressions or constants."
>
>
>
> That isn't quite correct. In this particular case, you *must* check (v != argv[0]), regardless of whether it is a constant or temp variable. That is because if the type conversion was not needed, then "v" will be equal to argv[0]. In this case, your code does not "own" argv[0] since it didn't allocate it. Freeing "v" is bad in that case, because IDL will attempt to free it later, and a double free will occur.
>
>
>
> Probably the best way to write the line of code is:
>
>
>
> if (v != argv[0]) IDL_Deltmp(v);
>
>
>
> Here, you don't need to use the macro, because you *know* that if "v" is not equal, then it must be a temporary that is owned by you, and you can free it without doing any further checks.
>
>
>
> Hope this helps.
>
>
>

> -Chris
>
> ExelisVIS