Subject: Re: Storing data in an array inside a structure MUCH slower in IDL 8.2.2
Posted by chris_torrence@NOSPAM on Thu, 21 Feb 2013 18:11:24 GMT
View Forum Message <> Reply to Message

Hi all,

I can reproduce the problem using Mark's code. As you pointed out, this is indeed because of operator overloading.

In IDL 8.2.1 and earlier, if you tried to do "mystruct.data[i]", and the "data" field was an object (say a HASH), then IDL would throw an error. The only way you could access the index "i" for an overloaded object was to use parentheses, like "(mystruct.data)[i]".

In IDL 8.2.2 we now check if that field is an overloaded object, and then either do regular array indexing or operator overloading. This eliminates the need for the parentheses. Unfortunately, the check for the overloaded object was being done in a naive, expensive way.

I have fixed the bug, and the fix will be in the next IDL service pack.

With the fix:
IDL> mgh_test_structure, N_DATA=1000000,/use
IDL 8.2.3: loading 1000000 long-integer values into structure-field array took 0.4820 s

Thanks for finding the bug and providing such a simple reproduce case!

-Chris
ExelisVIS