Subject: Re: IDL 8.4?
Posted by chris_torrence@NOSPAM on Wed, 15 Oct 2014 16:01:16 GMT
View Forum Message <> Reply to Message

On Wednesday, October 15, 2014 8:55:10 AM UTC-6, rryan%s...@gtempaccount.com wrote:
> I stumbled onto the "Whats's New in 8.4" page this morning...
>
>
>
> http://www.exelisvis.com/docs/WhatsNew.html
>
>
>
> I was under the notion that 8.4 wasn't coming out until November, but I guess it's out early. Anywho. It looks like they're really moving toward object-oriented techniques and a very pythonic experience, so I wanted to know if anyone has the inside track on some things:
>
>
>
> (1) what is this lambda-inline function business? How is it different than execute()? Can it be used with the VM? What advantages does it have?
>
>
>
> (2) they're adding static methods to all variables? does this mean variables must be deleted (a la obj_destroy)? Can variables still change types (like x=0 & x=x+2.0 & help,x)? What happens when I evoke a string-based method on a numerical type? How does the execution time for these new static methods compare to the "functional" or "procedural" versions? I guess, what advantage do you add here, other than complete loss of backwards compatibility (or forward, I don't know. I just know I can't use these methods and give to a friend who's still running <8.4).
>
>
>
> (3) Thye've added a boolean datatype. Is it really boolean, or is it just a byte or int that is capped at 1? I mean, will intrinsically boolean things (which should've returned booleans from day 1 which return ints such as: x = 1 lt 2 or t=keyword_set(SOME_VARIABLE)) return booleans now? So now you could do use keyword_set() to test if a keyword is set versus using keyword_set to see if the keyword is set to something non-zero (like it's always been).
>
>
>
> I've got more questions, but these are the most troubling.
>
>
>
> Russell

Hi Russell,

You beat me to the announcement. I just posted a long blurb about IDL 8.4 if you want to read about the new features.

Regarding your specific concerns, I would recommend downloading IDL 8.4 and playing around with the new features. In short, it's all "good" stuff - we haven't broken anything, and nothing has changed which would break backwards compatibility.

As is always true, if you use new features, those won't be compatible with older versions of IDL. So if you are developing a library for other people, you certainly want to be careful.

Regarding your specific questions about "static methods on variables" - IDL variables still work exactly like they did before. You can still change types, you don't need to clean them up, etc. The new methods should be the same speed or perhaps just a tiny bit slower (there is a small overhead for calling the method). If you are calling a method a million times in a loop you might notice, but for the normal IDL case (making one call with a million elements) you won't even notice.

Overall, our development goals are:
1. Keep existing users happy by adding feature requests and fixing bugs.
2. Move the language forward (yes, "like python") to attract new users.

Cheers,
Chris
ExelisVIS