Subject: Re: fast for loop
Posted by Conor on Mon, 11 Jun 2007 14:01:36 GMT
View Forum Message <> Reply to Message

On Jun 10, 10:55 pm, David Fanning <n...@dfanning.com> wrote:
> airy.ji...@gmail.com writes:
>> Sometimes we could use some special function to avoid them.Sometimes
>> we could use more lines of code to avoid them.
>> There are many skills to make the program more efficient and fast.But
>> in fact loops are ineluctable,the key is how
>> to use it.
>> Anyway,I agree with you ,Mark.IDL indeed need to improve its
>> efficiency on the loops and some arithmatic computing.IDL6.4
>> shows a lot of features which can be proved thier hard working for
>> it.At least 50% elevation of the drawing speed and new
>> OpenGL Object indicates an nice future of the IDL.It's wothy for us to
>> waiting new edtion of the IDL which can give us some
>> fast loops,^_^.
>
> Yeah, and life would be more interesting if pigs could fly.
>
> If fast loops are what you were after, I'd guess you would
> design a language that looked very much like C or FORTRAN.
> IDL is something different and I don't see much point wishing
> it wasn't.
>
> Cheers,
>
> David
> --
> David Fanning, Ph.D.
> Fanning Software Consulting, Inc.
> Coyote's Guide to IDL Programming:http://www.dfanning.com/
> Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Granted, it would be nice to have fast for loops (for those times where you really just have to use one). However, I do also enjoy the challenge of having to come up with fun new ways to avoid them. There's nothing more satisfying than taking a couple lines of code wrapped inside a for loop and turning it into one line of convoluted array operations. Normally I have no artistic talen what-so-ever, so coming up with confusing idl code in order to avoid for loops is my way of expressing my creative talents :)