
Subject: Re: fast for loop

Posted by [airy.jiang](#) on Mon, 11 Jun 2007 02:00:34 GMT

[View Forum Message](#) <> [Reply to Message](#)

On 6 11 , 2 33 , mark...@gmail.com wrote:

- > This is a feature request. I've been programming all day and my brain
- > is so tired of constantly trying to think of ways to avoid for loops.
- > IDL needs some kind of a fast for loop. That's all.
- >
- > Mark

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 arithmetic 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 edition of the IDL which can give us some fast loops, ^_^.

Subject: Re: fast for loop

Posted by [David Fanning](#) on Mon, 11 Jun 2007 02:55:29 GMT

[View Forum Message](#) <> [Reply to Message](#)

airy.jiang@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 arithmetic 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 edition 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.")

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 arithmetic 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 edition 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 :)

Subject: Re: fast for loop

Posted by [David Fanning](#) on Mon, 11 Jun 2007 17:39:51 GMT

[View Forum Message](#) <> [Reply to Message](#)

Conor writes:

> 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 :)

I'm a big proponent of the Programming as Poetry school of thought. But I would encourage you to read more Billy Collins:

<http://tinyurl.com/29dzj5>

And less of the more contemporary poets:

<http://www.moriapoetry.com/perrone998.html>

Your programs will be better for it. :-)

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.")

Subject: Re: fast for loop

Posted by [JD Smith](#) on Mon, 11 Jun 2007 19:00:36 GMT

[View Forum Message](#) <> [Reply to Message](#)

On Mon, 11 Jun 2007 07:01:36 -0700, Conor wrote:

> 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 arithmetic 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 edition 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.

> 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 :)

It's funny because it's true. Some of the tricks we resort to to get good performance out of IDL fall in the category of elegant. Most do not. I've long advocated a specially compiled for loop which drops essentially all the features of the IDL interpreter, which no doubt are what make a single round trip through the for loop so slow (warning: this is informed speculation). This "optimized side loop", which might get enabled with a compiler flag, would have some inherent inflexibility, but should offer much better performance.

JD

Subject: Re: fast for loop
Posted by [Conor](#) on Mon, 11 Jun 2007 19:26:53 GMT

On Jun 11, 1:39 pm, David Fanning <n...@dfanning.com> wrote:

> Conor writes:
>> 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 :)
>
> I'm a big proponent of the Programming as Poetry school
> of thought. But I would encourage you to read more Billy
> Collins:
>
> <http://tinyurl.com/29dzj5>
>
> And less of the more contemporary poets:
>
> <http://www.moriapoetry.com/perrone998.html>
>
> Your programs will be better for it. :-)
>
> 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.")

personally, I'm a fan of this type of poetry:

http://www.foo.be/docs/tpj/issues/vol3_2/tpj0302-0012.html

I was even pondering writing some obfuscated IDL, but it's just not the same as perl. IDL just doesn't have that naturally confusing syntax that makes obfuscation so wonderful :)

Subject: Re: fast for loop

Posted by [David Fanning](#) on Mon, 11 Jun 2007 19:36:05 GMT

[View Forum Message](#) <> [Reply to Message](#)

Conor writes:

> personally, I'm a fan of this type of poetry:

>
> http://www.foo.be/docs/tpj/issues/vol3_2/tpj0302-0012.html
>
> I was even pondering writing some obfuscated IDL, but it's just not
> the same as perl. IDL just doesn't have that naturally confusing
> syntax that makes obfuscation so wonderful :)

You are probably not yet familiar with Craig Markwardt's
IDL-ku verse:

<http://www.dfanning.com/tips/questions.html>

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.")

Subject: Re: fast for loop

Posted by [Conor](#) on Mon, 11 Jun 2007 19:46:05 GMT

[View Forum Message](#) <> [Reply to Message](#)

On Jun 11, 3:36 pm, David Fanning <n...@dfanning.com> wrote:

> Conor writes:
>> personally, I'm a fan of this type of poetry:
>
>> http://www.foo.be/docs/tpj/issues/vol3_2/tpj0302-0012.html
>
>> I was even pondering writing some obfuscated IDL, but it's just not
>> the same as perl. IDL just doesn't have that naturally confusing
>> syntax that makes obfuscation so wonderful :)
>
> You are probably not yet familiar with Craig Markwardt's
> IDL-ku verse:
>
> <http://www.dfanning.com/tips/questions.html>
>
> 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.")

:)
