
Subject: CParser enhancements
Posted by [cbpporter](#) on Wed, 10 Mar 2010 15:30:38 GMT
[View Forum Message](#) <> [Reply to Message](#)

First a short question: is there a class like MemStream, but which owns it's memory, grows with every Put and has a capacity. Basically a Vector<char> but with all the PutNN variants so I can add ints and doubles to it?

I couldn't find any and I'll implement one latter Today.

Now, for the real subject: I've attached some little enhancements to CParser. Basically just support for getting the column. It is 1 based. There is also a GetCharSpaces function, which should maybe have a better name.

Also, I'll need a CParser with optional space skipping which does not skip over end lines. So after a Spaces(), Char('\n') can be true. I'll inherit from CParser and override Spaces0 after I get nightly build with this patch.

I have a strange deja vu about this path, like I already posted it. I hope it's not a double post .

File Attachments

- 1) [Parser.h](#), downloaded 305 times
 - 2) [parser.cpp](#), downloaded 315 times
-

Subject: Re: CParser enhancements
Posted by [mirek](#) on Wed, 10 Mar 2010 15:44:58 GMT
[View Forum Message](#) <> [Reply to Message](#)

cbpporter wrote on Wed, 10 March 2010 10:30First a short question: is there a class like MemStream, but which owns it's memory, grows with every Put and has a capacity. Basically a Vector<char> but with all the PutNN variants so I can add ints and doubles to it?

StringStream. Frankly, it is very common, used much more frequently than MemStream.

Mirek

Subject: Re: CParser enhancements
Posted by [mirek](#) on Wed, 10 Mar 2010 15:46:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

cbpporter wrote on Wed, 10 March 2010 10:30

Also, I'll need a CParser with optional space skipping which does not skip over end lines. So after a Spaces(), Char('\n') can be true. I'll inherit from CParser and override Spaces0 after I get nightly build with this patch.

You can switch whitespace skipping off, then manage them yourself as you wish...

Mirek

Subject: Re: CParser enhancements
Posted by [mirek](#) on Wed, 10 Mar 2010 15:50:32 GMT
[View Forum Message](#) <> [Reply to Message](#)

I do not like the implementation, it is very expensive. Also, tabs are not implemented correctly.

(And now, as I am the one unsatisfied, I have to do it myself

Mirek

Subject: Re: CParser enhancements
Posted by [mirek](#) on Wed, 10 Mar 2010 16:15:44 GMT
[View Forum Message](#) <> [Reply to Message](#)

luzr wrote on Wed, 10 March 2010 10:50

(And now, as I am the one unsatisfied, I have to do it myself

Which I did.

Mirek

Subject: Re: CParser enhancements
Posted by [cbpporter](#) on Wed, 10 Mar 2010 16:58:41 GMT
[View Forum Message](#) <> [Reply to Message](#)

Fine by me!

The new implementation seems faster when GetColumn is used very rarely, but I think it will be considerably slower when I use it at every symbol (which I do when doing "debug mode builds").

I'll try it out.

Thank you!

Subject: Re: CParser enhancements
Posted by [mirek](#) on Wed, 10 Mar 2010 18:31:48 GMT
[View Forum Message](#) <> [Reply to Message](#)

cbpporter wrote on Wed, 10 March 2010 11:58Fine by me!

The new implementation seems faster when GetColumn is used very rarely, but I think it will be considerably slower when I use it at every symbol (which I do when doing "debug mode builds").

I'll try it out.

Thank you!

Actually, the primary motivation was to remove all those column increments spread through the code - I am afraid it was too error-prone.

Mirek

Subject: Re: CParser enhancements
Posted by [cbpporter](#) on Thu, 11 Mar 2010 14:14:48 GMT
[View Forum Message](#) <> [Reply to Message](#)

I'm following you example of the pointer difference, but on a function by function basis (like ReadId), to avoid all those increments.

I'll fork CParser for now into RParser. Anyway, the addition of column info is IMO a welcome addition to the handy little CParser.
