Subject: Re: PipeStream - bidirectional Stream
Posted by mirek on Wed, 20 Feb 2013 19:47:13 GMT
View Forum Message <> Reply to Message

dolik.rce wrote on Wed, 20 February 2013 14:40mirek wrote on Tue, 19 February 2013 00:40Set lims to value that allows fast Get/Put only in actual mode and forces _Get/_Put otherwise.

E.g. in read mode, set wrlim to buffer begin. Then when Put is used, it will go to _Put, where the situation can be fixed (and rdlim set to buffer begin).
Well, that sounded easier than it was It also required to get rid of the internal counter of available bytes, because the Stream methods wouldn't update it. This lead to some hidden bugs which took me quite some time to figure out...

Anyway, I commited the changes to sandbox and it should be working now. The last thing I'm not sure about is the GetLeft() method, returning the number of bytes available for reading. It is not virtual in Stream, but calculated as GetSize() - GetPos(). GetSize could be implemented in PipeStream to work correctly, but GetPos in it's current form will return wrong values Any ideas about this? What about making at least one of GetPos,GetLeft virtual too?

IMO, GetPos/GetLeft/GetSize etc... does not make much sense in non-seekable stream anyway (but we should update docs).

BTW, working on filter-streams, I have changed IsEof implementation to use Term instead. So as long as Term returns <0 value, sequential reading should work as expected.

Mirek