

CoCoALib - Feature #1036

LogStream: stream for log output

29 Mar 2017 11:48 - John Abbott

Status:	Closed	Start date:	29 Mar 2017
Priority:	Normal	Due date:	
Assignee:	John Abbott	% Done:	100%
Category:	Improving	Estimated time:	1.90 hour
Target version:	CoCoALib-0.99550 spring 2017	Spent time:	2.80 hours
Description On which output stream should logging messages from CoCoALib be printed? <ul style="list-style-type: none">• cout mixes logging messages with standard output -- this is helpful if you want to redirect output to a file• clog sends logging output to the C++ log/err stream -- this keeps logging messages separate from standard output CoCoALib could offer a LogStream for logging output; the user can then set this to be an "alias" for cout or clog			
Related issues: Related to CoCoALib - Design #983: Which ostream for verbose/obsolescent log ... Closed 25 Nov 2016			

History

#1 - 29 Mar 2017 11:49 - John Abbott

- Status changed from New to Feedback
- Assignee set to John Abbott
- Estimated time set to 1.90 h

Already implemented: very simple. Written doc.

The only thing missing is an example.

#2 - 29 Mar 2017 11:49 - John Abbott

- % Done changed from 0 to 90

#3 - 29 Mar 2017 11:58 - John Abbott

- Related to Design #983: Which ostream for verbose/obsolescent log msgs? added

#4 - 29 Mar 2017 18:09 - John Abbott

- Target version changed from CoCoALib-0.99560 to CoCoALib-0.99550 spring 2017

#5 - 05 Apr 2017 14:08 - John Abbott

There is one aspect of the design which is not wholly satisfactory:
there is no safe way of sending logging output to a user created stream (unless that stream is in a global variable).

Consider the following:

```
void MyFunc()  
{  
    ostream MyLogStream("cocoa.log");  
    SetLogStream(MyLogStream);  
    // blah blah  
}
```

When control leaves MyFunc the stream MyLogStream will be destroyed, but the hidden global set by SetLogStream will still refer to that now defunct ofstream. The situation is also exception unsafe.

The "correct" design would be an RAII implementation: in other words functions can "redirect" the LogStream only while control remains in that function (upon function exit, LogStream reverts to its previous value). The RAII approach is surely cleaner, but would make it tricky to implement SetLogStream in CoCoA-5; would that really matter?

#6 - 22 Apr 2017 22:52 - John Abbott

- *Status changed from Feedback to Closed*

- *% Done changed from 90 to 100*

I have redesigned the interface to use a RAII-style approach. Documentation and example has been updated. So the doubts mentioned in comment 5 have been properly resolved.

Closing.