By: Philippe Le Mouël
Rogue Wave Software Inc.
philippe@roguewave.com

Doc. No.: X3J16/96-0171
WG21/N0989
Date: September 22 1996

IOstreams Issues List
Library Clause 27

Revision History

Pre-Hawaii	X3J16/96-0171 WG21/N0989
Post-Stockholm	X3J16/96-0165 WG21/N0983
Pre-Stockholm	X3J16/96-0099 WG21/N0917
Post-Santa-Cruz	X3J16/96-0079 WG21/N0897
Pre-Santa Cruz	X3J16/96-0009 WG21/N0827
Post-Tokyo	X3J16/95-0221 WG21/N0821
Pre-Tokyo	X3J16/95-0194 WG21/N0794
Pre-Monterey	X3J16/95-0089 WG21/N0689
Pre-Austin	X3J16/95-0034 WG21/N0634

Summary of open Issues

27.4.3 ios_base P. 2

Active 27-108 cin, cout … construction and initialization (Box)
Active 27-109 can iword and pword fail, and if so how
Active 27-110 use of iword and pword with xalloc

27.4.4 basic_ios P. 3

Active 27-205 imbue should not call rdbuf( )->pubimbue
Active 27-207 availability of char_traits in header ios (Box)
Active 27-209 imbueing getloc()::codecvt into the argument stream buffer (Box)

27.6.1 basic_istream P. 4

Active 27-413 sentence missing (Box)

27.8 fstreams P. 5

Active 27-805 filebuf::imbue semantics
Active 27-814 basic_filebuf::imbue has no description

Miscellaneous P. 6

Active 27-922 Cleaning the iostreams header synopsis #include of other header

Summary of closed Issues P. 7

`ios_base` issues

Issue Number: 27-108
Title: cin, cout … construction and initialization (Box)
Section: 27.3 **Standard iostream objects [lib.iostream.objects]**
Status: **active**
Description:

The standard iostreams objects (cin, cout, cerr, clog, wcin, wcout, wcerr, wclog) need to be constructed and associations established before dynamic initialization of file scope variables is begun.

Possible Resolution:

Requestor: pre-Stockholm Iostreams WG

Issue Number: 27-109
Title: can iword and pword fail, and if so how
Section: 27.4.3.4 **ios_base storage functions [lib.ios.base.storage]**
Status: **active**
Description:

Can iword and pword fail, and if so how (throw bad_alloc and/or set a bit.)

Possible Resolution:

Requestor: Jerry Schwarz (jss@a.crl.com)

Issue Number: 27-110
Title: use of iword and pword with xalloc
Section: 27.4.3.4 **ios_base storage functions [lib.ios.base.storage]**
Status: **active**
Description:

The old iostream library was using an array of union (*long* and *void**) to implement the underlying data maintained by the function *iword* and *pword*, whereas the current WP describes *iword* as maintaining an array of *long* and *pword* as maintaining an array of *void* pointers. The problem with this new scheme, is when trying to combine the use of *iword*, *pword* and *xalloc*. The static function *xalloc* returns a value of type *int*, that can be used as parameter of the functions *iword* and *pword*. The purpose of *xalloc* is to allow users to fix location of *iword* and *pword* data between several stream objects (make sure a certain data will be located at the same place in several stream objects). The problem is that *xalloc* only maintains one set of indices, but *iword* and *pword* are two different arrays. Therefore using *xalloc* to generate indices to be used with *iword* and *pword* will create holes within the underlying arrays. For example, if you want to store X values of type *long* and then store one value of type *void**, the array maintained by *pword* will have its X first elements empty.

Possible Resolution:

Go back to the use of an array of union to maintain *iword* and *pword* data as in the old iostream.

Requestor: Philippe Le Mouël (philippe@roguewave.com)

`basic_ios` issues

Issue Number: 27-205
Title: imbue should not call rdbuf( )->pubimbue
Section: 27.4.4.2 **Member functions [lib.basic.ios.members]**
Status: **active**
Description:

basic_ios::imbue(const locale&) should call rdbuf( )->pubimbue(loc) only if rdbuf( ) is not a null pointer. Even better, it should not call rdbuf( )->pubimbue(loc) at all. Changing the locale that controls stream conversions is best separated from changing the locale that affects numeric formatting, etc. Anyone who knows how to imbue a proper pair of codecvt facets in a streambuf won't mind having to make an explicit call.

Possible Resolution:

The first part of the issue has already been resolved, the description of function *locale imbue(const locale& loc)* says:

Effects: Calls ios_base::imbue(loc) (27.4.3.3) and if rdbuf( ) != 0 then rdbuf( )->pubimbue(loc) (27.5.2.2.1).

Concerning the second part you have two possibilities:

If we leave the basic_ios::imbue function unchanged, when users call the imbue function from basic_istream, basic_ostream or the classes derived from them, they are actually changing both the locale of the stream object, and the locale of the stream buffer object attached to the stream object. This is not a real problem, because the stream object is only affected by the ctype, num_get , num_put, and numpunct facets, while the stream buffer object is affected by the codecvt facet. Therefore, even if you want to have several stream pointing to the same stream buffer objects (with a different locale object for each of them), you can easily do it by having all the different stream locale objects having the same codecvt facet. You could also do it by imbuing the stream buffer at last. The advantage of this scheme is that in simple cases you need to imbue just once in the stream object without having to wonder about the stream buffer object attached to it. The drawback is that you need to be more careful when you imbue in a stream object, and make sure that the locale object you are imbuing contains the correct codecvt facet; otherwise, you need to imbue the stream buffer object attached to the stream with another locale.

The other possibility is to remove the call to rdbuf( )->pubimbue(loc), in which case you just imbue the stream object itself. The problem is that in simple cases you need to imbue both the stream object and the stream buffer attached to it. The advantage is you only imbue the object that needs to have a change of locale.

This issue has to be treated with issue 27-209.

Requestor: Public Comment

Issue Number: 27-207
Title: availability of char_traits in header ios (Box)
Section: 27.4.2 **Template struct ios_traits [lib.ios.traits]**
Status: **active**
Description:

The *ios_traits* was previously defined in header *ios*. Character traits have now been consolidated into *char_traits* defined in the string header. Should they still be available after a *#include<ios>*.

Possible Resolution:

Requestor: pre-Stockholm Iostreams WG

Issue Number: 27-209
Title: imbueing getloc()::codecvt into the argument stream buffer (Box)
Section: 27.4.4.2 **Member functions [lib.basic.ios.members]**
Status: **active**
Description:

The first Box of section 27.4.4.2 says:

Note: need to modify so as to describe the occurrence of imbueing getloc()::codecvt into the argument stream buffer.

Possible Resolution:

Requestor: pre-Stockholm Iostreams WG

---

basic_istream issues

Issue Number: 27-413
Title: sentence missing (Box)
Section: 27.6.1.1 **Template class basic_istream [lib.istream]**
Status: **active**
Description:

X3J16/95-0149==WG21/N0749 deleted a sentence that prevented the formatted and unformatted input functions from calling other streambuf virtuals (such as seek).

Possible Resolution:

Requestor: pre-Stockholm Iostreams WG

---

file stream issues

Issue Number: 27-805
Title: filebuf::imbue semantics
Section: 27.8.1.4 **Overridden virtual functions [lib.filebuf.virtuals]**
Status: **active**
Description:

basic_filebuf::imbue has silly semantics. Whether or not sync() succeeds has little bearing on whether you can safely change the working codecvt facet. The most sensible thing is to establish this facet at construction. (Then pubimbue and imbue can be scrubbed completely.) Next best is while is_open() is false. (Then imbue can be scrubbed, since it has nothing to do.) Next best is to permit any imbue that doesn't change the facet or is at beginning of file. Next best is to permit change of facet any time provided either the current or new facet does not mandate state-dependent conversions. (See comments under seekoff.)

Possible Resolution:

In my current version of the WP, I do not have any description for the virtual filebuf imbue function. See issue 27-814.

Requestor: Public Comment

Issue Number: 27-814
Title: basic_filebuf::imbue has no description
Section: 27.8.1.4 **Overridden virtual functions [lib.filebuf.virtuals]**
Status: **active**
Description:

The virtual function basic_filebuf::imbue has no description.

Possible Resolution:

Requestor: Philippe Le Mouël (philippe@roguewave.com)

Miscellaneous issues

Issue Number:	27-922
Title:	Cleaning the iostreams header synopsis #include of other headers
Section:	Chapter 27 and Annex D
Status:	**active**
Description:	

Several public comments pointed out that the C++ header synopsis #include of other C++ headers were not correct.

Possible Resolution:

Add to header <iostream> synopsis:	<istream> and <ostream>	
Add to header <sstream> synopsis:	<string>	(already brought by locale)
Add to header <streambuf> synopsis:	<locale>	
Add to header <strstream> synopsis:	<streambuf>	

Requestor: Judy Ward

Summary of closed Issues

27.4.2 ios_traits

Closed 27-001 Making newline locale aware (Stockholm)
Closed 27-002 is_whitespace is inconsistent (Stockholm)
Closed 27-003 Mention of base struct string_char_traits (Tokyo)
Closed 27-004 example of changing the behavior of is_whitespace is incorrect (Stockholm)
Closed 27-005 not_eof specification (Stockholm)
Closed 27-006 streamsize should be ***SZ_T*** not ***INT_T*** (Tokyo)
Closed 27-007 ios_traits typedefs are 'char' oriented (Stockholm)
Closed 27-008 ios_traits::length is missing **Returns:** clause (Stockholm)
Closed 27-009 ios_traits::get_state should be specified (Stockholm)
Closed 27-010 ios_traits::get_pos should be specified (Stockholm)
Closed 27-011 Return type for ios_traits::copy is incorrect (Stockholm)

27.4.3 ios_base

Closed 27-101 ios_base manipulators (Santa-Cruz)
Closed 27-102 ios_base::width semantics are incorrect (Santa-Cruz)
Closed 27-103 proposal for adding ios_base::maxwidth (Santa-Cruz)
Closed 27-104 ios_base unitbuf and nounitbuf manipulators (Santa-Cruz)
Closed 27-105 ios_base storage functions are not exception safe (Santa-Cruz)
Closed 27-106 Init class should be an implementation detail (Stockholm)
Closed 27-107 ios::failure doesn't have the same functionality (Stockholm)

27.4.4 basic_ios

Closed 27-201 remove throw specifications for clear and setstate (Tokyo)
Closed 27-202 tie not required to be associated with an input sequence (Tokyo)
Closed 27-203 operator bool( ) needs to be fixed (Stockholm)
Closed 27-204 replace int_type by char_type in int_type fill( ) and int_type fill(int_type) (Tokyo)
Closed 27-206 clear() should not unconditionally clear the error state (Stockholm)
Closed 27-208 move member functions from basic_ios to ios_base (Box) (Stockholm)

27.5.2 basic_streambuf

Closed 27-301 imbuing on streambufs. When, how often, etc... (Stockholm)
Closed 27-302 sungetc has an incorrect return type (Tokyo)
Closed 27-303 not_eof needs to be used where appropriate (Santa-Cruz)
Closed 27-304 uflow needs editing (Santa-Cruz)
Closed 27-305 basic_streambuf::showmanyc Incorrect return clause (Santa-Cruz)
Closed 27-306 basic_streambuf::uflow has incorrect default behavior (Santa-Cruz)
Closed 27-307 basic_streambuf::uflow has nonsense returns clause (Santa-Cruz)
Closed 27-308 streambuf inlines (Santa-Cruz)
Closed 27-309 two return clauses for streambuf::underflow (Santa-Cruz)
Closed 27-310 streambuf::pbackfail has incorrect **Notes:** clause (Santa-Cruz)
Closed 27-311 caching results of calls to locale functions (Santa-Cruz)
Closed 27-312 sync does not say what happens to the input sequence (Box) (Stockholm)

27.6.1 basic_istream

Closed	27-401	isfx what does it do? (Santa-Cruz)
Closed	27-402	ipfx example is incorrect (Santa-Cruz)
Closed	47-403	Clarification of exceptions thrown (Santa-Cruz)
Closed	27-404	istream functions need to check for NULL streambuf (Stockholm)
Closed	27-405	confusing English in formatted requirements (Stockholm)
Closed	27-406	operator>>(char_type *) failure (Stockholm)
Closed	27-407	operator>>(char_type) failure (Stockholm)
Closed	27-408	ws manipulator (Stockholm)
Closed	27-409	unsigned short extractors cannot use unsigned long get function (Stockholm)
Closed	27-410	putback function has wrong description (Stockholm)
Closed	27-411	getline should not set failbit when reading no characters (Stockholm)
Closed	27-412	operator >>(basic_streambuf *sb), should not set badbit if sb is null (Stockholm)
Closed	27-414	readsome, putback and unget need to check for good (Box) (Stockholm)
Closed	27-415	streampos need to be replaced (Box) (Stockholm)

27.6.2 basic_ostream

Closed	27-501	op<<(char) needs to be consistant with the other formatted inserters (Stockholm)
Closed	27-502	op<<(void *) should it be const volatile void * (Santa-Cruz)
Closed	27-503	ostream functions need to check for NULL streambuf (Stockholm)
Closed	27-504	exceptions in ostream (Santa-Cruz)
Closed	27-505	incorrect conversion specifier for operator<<(unsigned long) (Stockholm)
Closed	27-506	wrong default behavior for padding (Stockholm)

27.6.1-27.6.2 basic_istream, basic_ostream

Closed	27-601	op[<<|>>](ios_base&) needed for manipulators (Stockholm)
Closed	27-602	positional typedefs in istream/ostream derived classes are not needed (Stockholm)
Closed	27-603	read/write should take a void * instead of a char_type * (Stockholm)
Closed	27-604	Should we require ios::in to be set for istream's and ios::out to be set for ostream's (Stockholm)
Closed	27-605	Should get/put use iterators? (Stockholm)
Closed	27-606	seekg and seekp should have their first parameter passed by value. (Stockholm)
Closed	27-607	locale getnum needed for void* extractor (Box) (Stockholm)

27.6.3 Standard manipulators

Closed	27-651	setfill description is wrong (Stockholm)
Closed	27-652	smanip is not a single type (Stockholm)

27.7 string streams

Closed	27-701	str() needs to clarify return value on else clause (Stockholm)
Closed	27-702	string stream classes need to have string_char_traits and allocator parameters (Stockholm)
Closed	27-703	stringbuf postconditions (Stockholm)
Closed	27-704	stringbuf::stringbuf constructor (Stockholm)
Closed	27-705	Incorrect calls to setg and setp in Table 14 (Stockholm)
Closed	27-706	Incorrect calls to setg and setp in table 16 (Stockholm)
Closed	27-707	setbuf function is missing (Stockholm)

27.8 fstreams

Closed 27-801 filebuf::underflow example is incorrect (Stockholm)
Closed 27-802 filebuf::is_open is a bit confusing (Stockholm)
Closed 27-803 ofstream constructor missing trunc as openmode (Stockholm)
Closed 27-804 ofstream::open missing trunc in openmode (Stockholm)
Closed 27-806 filebuf::seekoff **Effects:** clause needs work (Stockholm)
Closed 27-807 filebuf::underflow performance questions (Stockholm)
Closed 27-808 Editorial fixes in wording for fstreams (Stockholm)
Closed 27-809 description of function setbuf is missing (Stockholm)
Closed 27-810 openmode notation is not consistent in basic_ifstream and basic_ofstream (Stockholm)
Closed 27-811 description of function sync is missing (Stockholm)
Closed 27-812 filebuf::overflow example is incorrect (Stockholm)
Closed 27-813 basic_filebuf::overflow does not specifies its return value on success (Stockholm)
Closed 27-815 description of function seekpos is missing (Stockholm)
Closed 27-816 (i)(o)fstream open functions should not use is_open (Stockholm)

Miscellaneous

Closed 27-901 input/output of unsigned char, and signed char (Stockholm)
Closed 27-902 default locale (Santa-Cruz)
Closed 27-903 ipfx/opfx/isfx/osfx not compatible with exceptions (Santa-Cruz)
Closed 27-904 iosfwd declarations incomplete (Stockholm)
Closed 27-905 iostream type classes are missing (Santa-Cruz)
Closed 27-906 add a typedef to access the traits parameter in each stream class (Stockholm)
Closed 27-907 Use of “instance of” vs. “version of” in descriptions of class ios (Stockholm)
Closed 27-908 unnecessary ‘;’ (semicolons) in tables (Stockholm)
Closed 27-909 Editorial issues (typo’s) (Stockholm)
Closed 27-910 remove streampos in favor of pos_type (Stockholm)
Closed 27-911 stdio synchronization (Stockholm)
Closed 27-912 removing **Notes:** from the text (Stockholm)
Closed 27-913 Incorporating **Notes:** into the text (Stockholm)
Closed 27-914 rethrowing exceptions (Stockholm)
Closed 27-915 The use of specialization (Stockholm)
Closed 27-916 missing descriptions of specializations (Stockholm)
Closed 27-917 Editorial changes (Stockholm)
Closed 27-918 Validity of OFF_T to POS_T conversion (Stockholm)
Closed 27-919 Question on Table 2 assertions (Stockholm)
Closed 27-920 destination of clog and wclog (Stockholm)
Closed 27-921 default locale argument to constructor (Stockholm)

Annex D

Closed 27-1001 description of function setbuf is not sufficient (Stockholm)
Closed 27-1002 strstreambuf Editorial issues (Stockholm)
Closed 27-1003 istrstream Editorial issues (Stockholm)
Closed 27-1004 ostrstream Editorial issues (Stockholm)