

High quality library from scratch

Tools, technologies and acceptance to
Boost

Antony Polukhin
Полухин Антон

Boost libraries maintainer (LexicalCast, Any, TypeIndex, Conversion)
+ Boost.CircularBuffer, Boost.Variant

Table of contents

Generic part:

- * The Idea
- * Portability
- * License
- * Testing tools
- * Documentation
- * Checking interest

Boost only:

- * Requisitions
- * Review
- * After Acceptance

The Idea



- * find a **problem**
- * **search** for existing solutions
- * get the **Idea** of how to solve the problem

The Idea

```
boost::function<int(int)> c_func = boost::dll::import<int(int)>(
    path_to_shared_library, "c_func_name"
);
int i = c_func(1);
```

Portability

Better portability means more users and more attention to the library.

- * Platforms: Linux, Windows, MacOS, Android
- * Standards: C++14 C++11 C++98

C++14 Library (Hana)

```
namespace core_detail {
    template <typename Datatype>
    struct default_make {
        template <typename ...X>
        static constexpr auto apply_impl(int, X&& ...x)
            -> decltype(Datatype(detail::std::forward<X>(x)...))
        { return Datatype(detail::std::forward<X>(x)...); }

        template <typename ...X>
        static constexpr auto apply_impl(long, X&& ...) {
            static_assert((sizeof...(X), false),
                "there exists no constructor for the given data type");
        }

        template <typename ...X>
        constexpr decltype(auto) operator()(X&& ...x) const
        { return apply_impl(int{}, detail::std::forward<X>(x)...); }
    };
}
```

```
template <typename Datatype, typename = void>
constexpr auto make = make<Datatype, when<true>>>;
```

```
template <typename Datatype, bool condition>
constexpr core_detail::default_make<Datatype> make<Datatype, when<condition>>{};
```

C++14 Library (Hana)

```
auto ts = make<Tuple>(1, '2', 3.3, std::string{"abcd"});  
BOOST_HANA_RUNTIME_ASSERT(ts == tuple(1, '2', 3.3,  
    std::string{"abcd"}));
```


License

Choose according to library purpose:

- * educational: LGPL Apache BSD MIT Boost
- * making money: [A]GPL+Commercial
- * making money in Russia: Commercial

You may have **no choice** if you **forked** the project

Testing

Testing is essential for high quality library!

- tests must be run often
- test must cover all the library functionality
- static analysis tools are welcomed

[illegible]

Testing in Boost

Test output: teeks99-08f-win2012R2-64on64 - variant - variant_reference_test / msvc-14.0

Rev 5e5e0342a901b42c3b3607cc8147fc2d4fb5f85b / Mon, 19 Jan 2015 20:22:51 +0000

Compile [2015-01-20 00:46:22 UTC]: **fail**

```
call "c:\users\boost\appdata\local\temp\boost_regression\b2_msvc_14.0_vcvarsall_x86.cmd" >nul
cl /Zm800 -nologo @"D:\local\teeks99-08f\results\boost\bin.v2\libs\variant\test\variant_reference_test.test\msvc-14.0\debug\asynch-exceptions-c
variant_reference_test.cpp
..\libs\variant\test\variant_reference_test.cpp(88): error C2784: 'boost::add_reference<T>::type wknd_get(boost::variant<T,> &,int)': could not
with
[
    T=int
]
..\libs\variant\test\variant_reference_test.cpp(47): note: see declaration of 'wknd_get'
..\libs\variant\test\variant_reference_test.cpp(111): note: see reference to function template instantiation 'void base_derived_test<int&,int>(U
with
[
    Derived=int
]
```

Command Line

```
run.py --runner=teeks99-08f-win2012R2-64on64 --toolsets=msvc-14.0 \
"--bjam-options=\"-j2 address-model=64 define=BOOST_MSVC_ENABLE_2014_JUN_CTP --remove-test-targets\" \" \
--comment=..\info.html --tag=develop
```

Tom Kent - runner: teeks99-02

Contact me: [REDACTED]@yahoo.com, [REDACTED]

This runner is a VM on a KVM/QEMU Virtual Machine. Its has 6GB of ram and 2 CPU Cores.
This is running the 64-bit verison of Windows Server 2008r2.

Testing at Github

Test results

Branches	Build	Tests coverage
Develop:	build passing	coverage 95%
Master:	build passing	coverage 95%

Static analysis (Coverity): coverity passed

- TravisCI
- Coveralls
- Coverity

TravisCI

```
13  os:
14    - linux
15
31
32  matrix:
33    - CXX_STANDARD=c++98
34    - CXX_STANDARD=c++0x
35
36  addons:
37    coverity_scan:
38      # Mail to send notification to
39      notification_email: [REDACTED]
40
41      # This specifies the branch pattern for Coverity tests. If git branch and pattern mismatch,
42      # then coverity won't run. Set to `branch_pattern: $TRAVIS_BRANCH` if you want to do analysis on each build.
43      #branch_pattern: disable_coverity_scan
44      branch_pattern: $TRAVIS_BRANCH
45
46  #####
47  # From this point and below code is same for all the Boost libs
48  #####
49    build_command_prepend: "echo $COV_BUILD_OPTIONS $COVERITY_SCAN_BUILD_COMMAND"
50    build_command:  "../.../b2 cxxflags=--coverage\ -std=$CXX_STANDARD linkflags=--coverage "
51    project:
52      name: "$TRAVIS_REPO_SLUG"
53      description: "Build submitted via Travis CI"
54
55
56  before_install:
57    # Set this to the name of the library
58    - PROJECT_TO_TEST=`basename $TRAVIS_BUILD_DIR`
59    - echo "Testing $PROJECT_TO_TEST"
```

TravisCI

apolukhin/Boost.DLL 

build **passing**

 Settings ▾

Current

Build History

Pull Requests

Branch Summary




develop - Coverity tool experiments (no 1)

 #114 passed

 ran for 20 min 37 sec



 19 minutes ago

 Commit fd93bec

 Compare c16a1a8..fd93bec

 Antony Polukhin authored and committed

Build Matrix

Job	Duration	Finished	ENV	OS
 114.1	10 min 29 sec	19 minutes ago	CXX_STANDARD=c++98	linux
 114.2	10 min 8 sec	19 minutes ago	CXX_STANDARD=c++0x	linux

TravisCI

```
2159 **passed** ../../bin.v2/libs/Boost.DLL/test/getting_started.test/gcc-4.6/debug/threading-  
multi/getting_started.test  
2160 common.mkdir ../../bin.v2/libs/Boost.DLL/test/tutorial1.test  
2161 common.mkdir ../../bin.v2/libs/Boost.DLL/test/tutorial1.test/gcc-4.6  
2162 common.mkdir ../../bin.v2/libs/Boost.DLL/test/tutorial1.test/gcc-4.6/debug  
2163 common.mkdir ../../bin.v2/libs/Boost.DLL/test/tutorial1.test/gcc-4.6/debug/threading-multi  
2164 gcc.compile.c++ ../../bin.v2/libs/Boost.DLL/test/tutorial1.test/gcc-4.6/debug/threading-  
multi/tutorial1.o  
2165 gcc.link ../../bin.v2/libs/Boost.DLL/test/tutorial1.test/gcc-4.6/debug/threading-multi/tutorial1  
2166 testing.capture-output ../../bin.v2/libs/Boost.DLL/test/tutorial1.test/gcc-4.6/debug/threading-  
multi/tutorial1.run  
2167 **passed** ../../bin.v2/libs/Boost.DLL/test/tutorial1.test/gcc-4.6/debug/threading-  
multi/tutorial1.test  
2168 common.mkdir ../../bin.v2/libs/Boost.DLL/test/tutorial2.test  
2169 common.mkdir ../../bin.v2/libs/Boost.DLL/test/tutorial2.test/gcc-4.6  
2170 common.mkdir ../../bin.v2/libs/Boost.DLL/test/tutorial2.test/gcc-4.6/debug  
2171 common.mkdir ../../bin.v2/libs/Boost.DLL/test/tutorial2.test/gcc-4.6/debug/threading-multi  
2172 gcc.compile.c++ ../../bin.v2/libs/Boost.DLL/test/tutorial2.test/gcc-4.6/debug/threading-  
multi/tutorial2.o  
2173 gcc.link ../../bin.v2/libs/Boost.DLL/test/tutorial2.test/gcc-4.6/debug/threading-multi/tutorial2  
2174 testing.capture-output ../../bin.v2/libs/Boost.DLL/test/tutorial2.test/gcc-4.6/debug/threading-  
multi/tutorial2.run  
2175 **passed** ../../bin.v2/libs/Boost.DLL/test/tutorial2.test/gcc-4.6/debug/threading-
```

Coveralls

COVERALLS

HOME

FEATURES

SIGN UP

PRICING

DOCS

BLOG

SIGN IN

APOLUKHIN / BOOST.DLL

95%

BRANCH: DEVELOP

GITHUB REPO

LATEST BUILDS

BUILD	BRANCH	COVERAGE	COMMIT	COMMITTER	TYPE	TIME	VIA
#114	develop	94.63	Coverity tool experiments (no 1)	apolukhin	push	15 Jan 2015	travis-ci
#113	develop	94.63	Attempt(no 11) to use coverity tool	apolukhin	push	13 Jan 2015	travis-ci
#112	develop	94.63	Attempt(no 10) to use coverity tool	apolukhin	push	13 Jan 2015	travis-ci
#111	develop	94.63	Attempt(no 9) to use coverity tool	apolukhin	push	13 Jan 2015	travis-ci
#110	develop	94.63	Attempt(no 8) to use coverity tool	apolukhin	push	13 Jan 2015	travis-ci
#109	develop	94.63	Attempt(no 7) to use coverity tool	apolukhin	push	13 Jan 2015	travis-ci
#104	develop	46.08	Attempt(no 2) to use coverity tool	apolukhin	push	13 Jan 2015	travis-ci

Coveralls

FILES

SEARCH:

ALL 15

CHANGED 0

SOURCE CHANGED 0

COVERAGE CHANGED 0

SHOW 10 ENTRIES


▲ COVERAGE	✎	FILE	↕ LINES	↕ RELEVANT	↕ COVERED	↕ MISSED	↕ HITS/LINE
+ 50.0		...-local/boost/dll/detail/posix/path_from_handle.hpp	45	6	3	3	27.0
+ 73.33		...home/travis/boost-local/boost/dll/library_info.hpp	168	45	33	12	4.0
+ 96.67		...cal/boost/dll/detail/posix/shared_library_impl.hpp	176	60	58	2	53.0
+ 98.86		...me/travis/boost-local/boost/dll/shared_library.hpp	496	88	87	1	19.0
+ 98.91		...e/travis/boost-local/boost/dll/detail/elf_info.hpp	270	92	91	1	90.0
+ 100.0		.../boost-local/boost/dll/detail/x_info_interface.hpp	32	2	2	0	4.0
+ 100.0		...avis/boost-local/boost/dll/detail/system_error.hpp	49	5	5	0	4.0
+ 100.0		...l/boost/dll/detail/posix/program_location_impl.hpp	107	2	2	0	29.0
+ 100.0		...ost-local/boost/dll/detail/aggressive_ptr_cast.hpp	44	1	1	0	64.0
+ 100.0		...boost-local/boost/dll/shared_library_load_mode.hpp	239	5	5	0	130.0

SHOWING 1 TO 10 OF 15 ENTRIES

Coveralls

```
18  #ifdef BOOST_HAS_PRAGMA_ONCE
19  # pragma once
20  #endif
21
22  namespace boost { namespace dll { namespace detail {
23
24      inline boost::filesystem::path path_from_handle(void* handle, 27x
        boost::system::error_code &ec) {
25          // RTLD_DI_LINKMAP (RTLD_DI_ORIGIN returns only folder and is not suitable for
this case)
26          // Obtain the Link_map for the handle that is specified.
27          // The p argument points to a Link_map pointer (Link_map
28          // **p). The actual storage for the Link_map structure is
29          // maintained by ld.so.1.
30          const struct link_map * link_map;
31          if (dldinfo(handle, RTLD_DI_LINKMAP, &link_map) < 0) { 27x
32              ec = boost::system::error_code(
33                  boost::system::errc::bad_file_descriptor,
34                  boost::system::generic_category() !
35              ); !
36
37              return boost::filesystem::path(); !
38          }
39
40          return boost::filesystem::path(link_map->l_name); 27x
41      }
```

Coverity

apolukhin-Boost.DLL

HelpGuided TourReturn to Dashboardantoshkka@gmail.comEnter CID(s)

Issues: By Snapshot | Unsaved viewFilters: CID

CID	Type	Impact	Status	Count	First Detected	Owner	Classification	Sev
All 1 issue selected								

path.hpp

```
636 // iterator_facade derived classes don't seem to like implementations in
637 // separate translation unit dll's, so forward to class path static members
638 void increment() { m_path_iterator_increment(*this); }
639 void decrement() { m_path_iterator_decrement(*this); }
640
641 path m_element; // current element
642 const path* m_path_ptr; // path being iterated over
643 string_type::size_type m_pos; // position of m_element in
644 // m_path_ptr->m_pathname.
645 // if m_element is implicit dot, m_pos is the
646 // position of the last separator in the path.
647 // end() iterator is indicated by
648 // m_pos == m_path_ptr->m_pathname.size()
649 }; // path::iterator
650
651 //-----//
652 // non-member functions //
653 //-----//
654
655 // std::lexicographical_compare would infinitely recurse because path iterators
656 // yield paths, so provide a path aware version
```

100634 Uninitialized pointer field

The pointer field will point to an arbitrary memory location, any attempt to write may cause corruption.

In boost::filesystem::path::iterator::iterator(): A pointer field is not initialized in the constructor (CWE-457)

Classification:

Severity:

Action:

Ext. Reference:

Owner:

Enter comments (See the History section below for previous comments)

Apply + Next Apply

Projects & Streams

Detection History

Triage History

Occurrences

Documentation

- * make `gh-pages` branch
- * push HTML to it
- * PROFIT:

The screenshot shows a web browser window displaying the Boost C++ Libraries documentation. The address bar shows the URL `apolukhin.github.io/Boost.DLL/boost_dll/introduction0.html`. The page features the Boost logo and navigation links: Home, Libraries, People, FAQ, and More. Below the navigation bar, there is a section titled "Getting started" with a paragraph explaining that to start with the library, one needs to include the `<boost/dll.hpp>` header. A code block shows the `using namespace boost;` statement. Below this, there are two columns: "In DLL/DSL sources:" and "In code that uses DLL/DSL:". The first column contains C++ code for exporting a function, and the second column contains C++ code for importing a function using `import_alias`.

apolukhin.github.io/Boost.DLL/boost_dll/introduction0.html

Search

boost
C++ LIBRARIES

Home Libraries People FAQ More

← ↑ ↗ →

Getting started

To start with the library you only need to include `<boost/dll.hpp>` header. After that you are free to import and export functions and variables:

```
using namespace boost;
```

In DLL/DSL sources:	In code that uses DLL/DSL:
<pre>// exporting 'C++11' function namespace my_namespace { int cpp11_function(std::string&& param); BOOST_DLL_AUTO_ALIAS(cpp11_function) } // namespace my_namespace</pre>	<pre>auto cpp11_func = dll::import_alias<int(std::string&&)>(path_to_shared_library, "cpp11_function");</pre>

Documentation

- * Motivation
- * Quick Start
- * Tutorial covering basic use cases
- * Reference documentation
- * How to compile and link
- * Rationale for design decisions

Checking interest

- * requests for new features
- * comments on API
- * requests for documentation clarification
- * good ideas and useful links

```
// Class `library_info` can extract information from a library
boost::dll::library_info inf(libraries[i]);

// Getting symbols exported from 'Anna' section
std::vector<std::string> exports = inf.symbols("Anna");
```

You are not ought to implement all requests!

Acceptance to Boost

- * Requirements:

<http://www.boost.org/development/requirements.html>

- * Boost Library Incubator

<http://blincubator.com/>

- * Review manager

The Review



After acceptance



The end



Make your ads || business at
<http://top-me.org/>