

Boost C++ Librariesの概要

高橋 晶 (Akira Takahashi)

[id:faith_and_brave](#)

[@cpp_akira](#)

Boost.勉強会 #7 東京 2011/12/03(土)

- C++標準化委員会のメンバが立ち上げたオープンソースのライブラリ群Boost C++ Libraries
- BoostPro Computingというサポート会社がある
- ライセンスはBoost Software Licence 1.0
 - 商用利用可
 - 改変自由
 - ソースコードにライセンス表記する必要はない(静的リンクの場合)
- 開発に参加してる企業: Adobe, Google, Intelなど...

- BoostはC++標準ライブラリの実験場として機能している
- 2011年に発行されたC++11では、Boostから多くのライブラリが導入された
- Boostから標準入りしたライブラリの例：
スマートポインタ、正規表現、乱数、型特性、スレッドなど。
- Boostが言語機能に与えた影響：
C++11で導入されたラムダ式は、Boost.Lambda作者によって提案された。右辺値参照やnoexcept、Variadic TemplatesもBoostのメンバが主動して導入された。

C++11に導入されたものを除けば、以下のようなライブラリがある:

- ファイルシステム(Filesystem)
- ネットワーク(Asio)
- シリアライズ(Serialization)
- オプション型(Optional)
- 構文解析(Spirit)
- 線形代数(uBLAS)
- 計算幾何(Geometry)
- 統計処理(Accumulators)
- 区間計算(Interval)
- 状態マシン
- etc...

今後導入されるかもしれない、開発中もしくはリリース前のライブラリ:

- Lock-freeコンテナ
- 多倍長整数
- ネットワークの上位プロトコル(HTTP, SMTP, XMPP, ICMP, etc...)
- プロセス管理
- コルーチン(ファイバ)
- ロギング
- 暗号化
- etc...

- ユーザーML
<http://lists.boost.org/mailman/listinfo.cgi/boost-users>
- 開発者ML
<http://lists.boost.org/mailman/listinfo.cgi/boost>
- その他MLまとめ
<https://sites.google.com/site/boostjp/mailing-list>
- Subversionリポジトリ
<http://svn.boost.org/svn/boost/trunk>
- Gitミラー
<https://github.com/ryppl/boost-svn>
- バグ報告
<https://svn.boost.org/trac/boost/>

- Boostには現在、日本人が作ったライブラリはまだないが、バグ報告やパッチ送付、議論への参加といった多くの貢献を行なっている。
- また、boostjpコミュニティでは、リリースノートの翻訳や、逆引きリファレンスの作成を行なっている。

- 事例1 : Boost.Serialization
近藤貴俊(redboltz)
多重継承に関する根深い問題を解決するパッチを送付
<https://svn.boost.org/trac/boost/ticket/3604>

Access violation on diamond inheritance

Opened 2 years ago

Last modified 19 months ago

Reported by: kondo@...

Owned by: ramey

Milestone: ~~Boost 1.41.0~~

Component: serialization

Version: Boost 1.40.0

Severity: Not Applicable

Keywords:

Cc:

Description

phenomenon[Reply](#)

When I serialize the sub-class via virtual base class, some BOOST_CLASS_EXPORT order makes access violation.

main.cpp(attached file) reproduce these behavior on VC++ ver 9. Please check it.

Classes structure is structure.png(attached file).

If BOOST_CLASS_EXPORT order is Target, Sub1, the error doesn't occur. If BOOST_CLASS_EXPORT order is Sub1, Target, the error occurs.

main.cpp

```
#if 0
// OK
BOOST_CLASS_EXPORT(Target)
BOOST_CLASS_EXPORT(Sub1)
```


事例1 : Boost.Serialization

Acknowledgmentsにも名前が載ってます



Serialization

Acknowledgments

- Takatoshi Kondo found and corrected a very obscure and difficult bug in the serialization of virtual base classes.
- [AutoForm Engineering GmbH](#) supported development efforts to extend correct serialization to objects stored in DLLs.
- Cadence Israel supported enhancement and testing of the portable binary archive.
- David Abrahams improved implementation of "export" functionality. This not only eliminated an annoying header sequencing requirement to maintain a list of "known archives".
- Mattias Troyer enhanced the implementation of native binary archives. This includes enhancement and generalization of the library of the wrapper concept.
- Markus Schöpflin tracked down issues with TRU64 compiler resulting in 100% passing.
- [Troy D. Straszheim](#) made the initial version of variant serialization.
- Tonko Juricic helped refine and complete project files for VC 7.1 ide.

「近藤貴俊は、仮想基本クラスのシリアライズにおける
非常に不透明で困難なバグを修正した」

事例2 : Boost.Range

Shunsuke Sogame(mb2sync)

Oven Range Libraryの実装経験により、Rangeアダプタの設計に貢献

Version 2 - Boost 1.43 and beyond

This version introduced Range Adaptors and Range Algorithms. This version 2 is the result of a merge of all of the RangeEx features into Boost.Range.

There were an enormous number of very significant contributors through all stages of this library.

Prior to Boost.RangeEx there had been a number of Range library implementations, these include library implementations by Eric Niebler, Adobe, Shunsuke Sogame etc. Eric Niebler contributed the Range Adaptor idea which is arguably the single biggest innovation in this library. Inevitably a great deal of commonality evolved in each of these libraries, but a considerable amount of effort was expended to learn from all of divergent techniques.

事例3 : Boost.Geometry

高橋 晶(faith_and_brave/cpp_akira)

Boost.Fusionのシーケンスと見なせるあらゆる型を、
Boost.Geometryで使えるようにした。

Contributions

Boost.Geometry contains contributions by:

- Akira Takahashi (adaption of Boost.Fusion)
- Alfredo Correa (adaption of Boost.Array)
- Adam Wulkiewicz (spatial indexes) ^[1]
- Federico Fernández (spatial indexes) ^[2]

手伝ってくれたDigitalGhostさんありがとう！

他にも、ドキュメントに名前までは載ってないけど貢献してるユーザーはたくさんいます。

- [\[config\] Macro BOOST_NO_NOEXCEPT is required](#)
→ @SubaruG
- [irange doesn't end iteration properly when step_size is 2 or more\(#5544\)](#)
→ @hotwatermorning
- [ptree::sort\(\) compilation error\(#5710\)](#)
[binomial_distribution<long long> compilation error\(#5705\)](#)
→ @bolero_MURAKAMI
- [is_rvalue_reference returns wrong result when rvalue reference to a function is passed\(#5795\)](#)
→ @sscrisk
- ["0 + 0" and "0 - 0" lead to a compile time error\(#5724\)](#)
[adjacent_filtered_range::m_pred should be removed\(#5486\)](#)
→ @iorate

- オープンソースのライブラリは、ユーザーが使えば使うほど便利になっていくという特性を持っています。
- より便利になるにはフィードバックが必要です。
- バグ報告の英語は難しくないです。
ほぼ定型文なので、他のチケットを見てマネしましょう。
英語が難しくて投稿できないという方は、boostjpのGoogle Groupに連絡してもらえればお手伝いします。

- boostjpコミュニティでは、Boostの日本語情報を充実させるために、いろいろな活動を行なっています。
- 現状、リリースノートの翻訳や逆引きリファレンスの作成を行なっています。
- 日本語情報が充実すれば、Boostを仕事で使えるように提案するのがやりやすくなるはずです！
「仕事で使えない・・・」と嘆いて終わりではなく、より便利に、より情報を充実させることで自らチャンスを作り出しましょう！

Boostの各ライブラリに詳しいTwitterユーザーをまとめたリストがあります。何かあったらどんどん聞きましょう。

ライブラリ名	リスト
Multi-Index	https://twitter.com/#!/cpp_akira/boost-multi-index/members
Graph	https://twitter.com/#!/cpp_akira/boost-graph/members
Range	https://twitter.com/#!/cpp_akira/boost-range/members
Fusion	https://twitter.com/#!/cpp_akira/boost-fusion/members
Thread	https://twitter.com/#!/cpp_akira/boost-thread/members
Asio	https://twitter.com/#!/cpp_akira/boost-asio/members
Preprocessor	https://twitter.com/#!/cpp_akira/boost-preprocessor/members
Optional	https://twitter.com/#!/cpp_akira/boost-optional/members
Variant	https://twitter.com/#!/cpp_akira/boost-variant/members
Flyweight	https://twitter.com/#!/cpp_akira/boost-flyweight/members
Phoenix	https://twitter.com/#!/cpp_akira/boost-phoenix/members
Python	https://twitter.com/#!/cpp_akira/boost-python/members

Boostの各ライブラリに詳しいTwitterユーザーをまとめたリストがあります。何かあったらどんどん聞きましょう。

ライブラリ名	リスト
Serialization	https://twitter.com/#!/cpp_akira/boost-serialization/members
Wave	https://twitter.com/#!/cpp_akira/boost-wave/members
Build	https://twitter.com/#!/cpp_akira/boost-build/members
Xpressive	https://twitter.com/#!/cpp_akira/boost-xpressive/members
Interprocess	https://twitter.com/#!/cpp_akira/boost-interprocess/members

Twitterのリストが20個しか作れないので悩み中。

Q & A