

Quassel IRC - Bug #1544

Quassel does not compile with qt 5.14.0

12/24/2019 04:47 PM - buzo

Status:	Confirmed	Start date:	12/24/2019
Priority:	High	Due date:	
Assignee:		% Done:	0%
Category:		Estimated time:	0.00 hour
Target version:	0.14.0	OS:	Any
Version:	0.13.0		
Description			
Compiling Quassel 0.13.1 with qt 5.14.0 gives compilation errors:			
<pre>In file included from /usr/include/qt/QtCore/QDataStream:1, from /build/quassel-core-small/src/quassel-0.13.1/src/common/syncableobject.h:24, from /build/quassel-core-small/src/quassel-0.13.1/src/common/dccconfig.h:25, from /build/quassel-core-small/src/quassel-0.13.1/src/common/dccconfig.cpp:21: /usr/include/qt/QtCore/qdatastream.h:389:1: note: candidate: 'typename std::enable_if<std::is_enum<_Tp>::value, QDataStream&>::type& operator<<(QDataStream&, const T&) [with T = DccConfig::PortSelectionMode; typename std::enable_if<std::is_enum<_Tp>::value, QDataStream&>::type = QDataStream&]' 389 operator<<(QDataStream &s, const T &t) ^~~~~~ In file included from /build/quassel-core-small/src/quassel-0.13.1/src/common/dccconfig.cpp:25: /build/quassel-core-small/src/quassel-0.13.1/src/common/types.h:152:14: note: candidate: 'QDataStream& operator<<(QDataStream&, T) [with T = DccConfig::PortSelectionMode; <template-parameter-1-2> = void]' 152 QDataStream &operator<<(QDataStream &out, T value) { ^~~~~~ In file included from /usr/include/qt/QtCore/qobject.h:54, from /usr/include/qt/QtCore/qiodevice.h:45, from /usr/include/qt/QtNetwork/qabstractsocket.h:44, from /usr/include/qt/QtNetwork/qhostaddress.h:48, from /usr/include/qt/QtNetwork/QHostAddress:1, from /build/quassel-core-small/src/quassel-0.13.1/src/common/dccconfig.h:23, from /build/quassel-core-small/src/quassel-0.13.1/src/common/dccconfig.cpp:21: /usr/include/qt/QtCore/qmetatype.h: In instantiation of 'static void QtMetaTypePrivate::QMetaTypeFunctionHelper<T, Accepted>::Load(QDataStream&, void*) [with T = DccConfig::PortSelectionMode; bool Accepted = true]': /usr/include/qt/QtCore/qmetatype.h:1802:39: required from 'void qRegisterMetaTypeStreamOperators(const char*, T*) [with T = DccConfig::PortSelectionMode]' /build/quassel-core-small/src/quassel-0.13.1/src/common/dccconfig.cpp:34:91: required from here /usr/include/qt/QtCore/qmetatype.h:823:16: error: ambiguous overload for 'operator>>' (operand types are 'QDataStream' and 'DccConfig::PortSelectionMode') 823 stream >> *static_cast<T*>(t); ~~~~~^~~~~~ In file included from /usr/include/qt/QtCore/QDataStream:1, from /build/quassel-core-small/src/quassel-0.13.1/src/common/syncableobject.h:24, from /build/quassel-core-small/src/quassel-0.13.1/src/common/dccconfig.h:25, from /build/quassel-core-small/src/quassel-0.13.1/src/common/dccconfig.cpp:21: /usr/include/qt/QtCore/qdatastream.h:394:1: note: candidate: 'typename std::enable_if<std::is_enum<_Tp>::value, QDataStream&>::type& operator>>(QDataStream&, T&) [with T = DccConfig::PortSelectionMode; typename std::enable_if<std::is_enum<_Tp>::value, QDataStream&>::type = QDataStream&]' 394 operator>>(QDataStream &s, T &t) ^~~~~~ In file included from /build/quassel-core-small/src/quassel-0.13.1/src/common/dccconfig.cpp:25: /build/quassel-core-small/src/quassel-0.13.1/src/common/types.h:166:14: note: candidate: 'QDataStream& operator>>(QDataStream&, T&) [with T = DccConfig::PortSelectionMode; <template-parameter-1-2> = void]' 166 QDataStream &operator>>(QDataStream &in, T &value) { ^~~~~~ [82/172] Building CXX object src/common/CMakeFiles/mod_common.dir/mod_common_autogen/mocs_compilation.cpp.o</pre>			

```
[83/172] Building CXX object src/common/CMakeFiles/mod_common.dir/ctcpevent.cpp.o
[84/172] Building CXX object src/common/CMakeFiles/mod_common.dir/coreinfo.cpp.o
[85/172] Building CXX object src/common/CMakeFiles/mod_common.dir/event.cpp.o
[86/172] Building CXX object src/common/CMakeFiles/mod_common.dir/eventmanager.cpp.o
ninja: build stopped: subcommand failed.
==> ERROR: A failure occurred in build().
```

History

#1 - 01/04/2020 02:14 PM - crabbedhaloablution

There's a similar issue in musescore:

<https://musescore.org/en/node/299162>

They fixed it with a commit:

<https://github.com/musescore/MuseScore/pull/5583>

" Specializes template routines.

Qt 5.14 introduces serialisation/deserialisation for enum classes, this results in ambiguous templates between qdatastream.h and preferences.h. we specialize everything to workaround this."

#2 - 01/07/2020 11:42 PM - rex4539

<https://github.com/quassel/quassel/pull/518>

#3 - 01/08/2020 05:35 PM - buzo

Thanks. When can I expect a new Quassel release containing these changes?

#4 - 04/11/2020 12:45 PM - buzo

The package built from the current Github version works fine on Arch Linux: <https://aur.archlinux.org/packages/quassel-core-small-git/>

Building version 0.13.1 still fails: <https://aur.archlinux.org/packages/quassel-core-small/>

Main bug report on Arch Linux: <https://bugs.archlinux.org/task/65085>

#5 - 06/20/2020 11:58 PM - digitalcircuit

We're hoping to have a 0.14-rc1 in the upcoming weeks, so this will hopefully be addressed soon. The Quassel CI was also updated to test against Debian testing/etc (<https://github.com/quassel/quassel/commit/1193d9d1f3f7c1c26e2c659b137046a31fe43bb4>) to catch this sooner in the future.

Thanks for your patience!

#6 - 06/21/2020 12:04 AM - digitalcircuit

- Status changed from *New* to *Confirmed*

- Target version set to 0.14.0

#7 - 08/01/2020 02:27 PM - Morbius

Thanks for fixing this issue. About five months ago I opened a related [MacPorts ticket](#) , and I am looking forward to the next Quassel release.

#8 - 04/10/2021 06:57 PM - buzo

By the way, qt 6 is out since a few months ...