Qt Project Tool

Table of Contents

The “qpt” (“Qt Project Tool”) is a command line utility that aims to extract information from Qt projects (.pro files) making it easy to use project set of files and settings as input to other command line tools.

As example other tools can be: Static Analyzers, Code Counters and others.

When "qpt" is used with Static Analyzer tools it also can work as a Qt Creator plugin, making it easy and fast to change source code inside Qt Creator.

What "qpt" does is to generate a list of sources, headers, defines and include path that can be passed to other tools. The information is printed to stdout and can be linked with other tools through pipes.

1 Syntax

By typing *qpt <ENTER>* or *qpt -h* | *--help <ENTER>* the help appears:

There are two syntax set for qpt, the help show both sets.

1.1 Syntax of qpt as wrapper for static analysis tools

```
qpt 0.7 - sourceforge.net QtProjectTool
Usage: [To use as a wrapper to any tool (for example static analyzers)]:
qpt [switchs] [options] --tool=<val> qtProject
Options:
  -m,--qmake-pathname=<val>  Full path of qmake tool
                             default: $QTDIR/bin/qmake when $QTDIR exists or qmake found in the $PATH
  -t,--tool=<val>            tool to use (single name or full pathname), choices are:[cppcheck,krazy2]
  -q,--qtc-plugin            It must be set when using inside QtCreator as an External Tool
  qtProject                  The Qt Project File (.pro)
```

1.2 Syntax of qpt to extract information from a Qt Project file

```
qpt 0.7 - sourceforge.net QtProjectTool
Usage: [To extract information from a Qt Project (useful in scripts)]:
qpt [switchs] [options] qtProject
Options:
  -m,--qmake-pathname=<val>  Full path of qmake tool, default: $QTDIR/bin/qmake when $QTDIR exists or qmake
found in the $PATH
  -S,--list-sources          List the source files of the project
  -H,--list-headers          List the header files of the project
  -A,--list-allSources       List both sources and headers of the project
  -D,--list-defines          List the defines set in the project
  -I,--list-includes         List the include path of the project
  -Q,--list-qtincludes       List Qt include path
  qtProject                  The Qt Project File (.pro)
```

2 Examples

2.1 Simple Listing to extract information from a Qt Project file

2.1.1 Listing only sources ($$SOURCES content from Qt Project file)

```
$ qpt -S /home/carlos/code_quality/doxyanalyzer/doxyanalyzer.pro

/home/carlos/code_quality/doxyanalyzer/extra_structure/extrautils.cpp
/home/carlos/code_quality/doxyanalyzer/libmd5/md5.c
/home/carlos/code_quality/doxyanalyzer/qtools/qbuffer.cpp
/home/carlos/code_quality/doxyanalyzer/qtools/qcollection.cpp
```

2.1.2 Listing all files ($$SOURCES + $$HEADERS content from Qt Project file)

```
$ qpt -A ../qprojtool/qprojtool.pro

/home/carlos/tmp/projects/projects/qprojtool/backendtool.cpp
/home/carlos/tmp/projects/projects/qprojtool/backendtool.h
/home/carlos/tmp/projects/projects/qprojtool/main.cpp
/home/carlos/tmp/projects/projects/qprojtool/manager.cpp
/home/carlos/tmp/projects/projects/qprojtool/manager.h
/home/carlos/tmp/projects/projects/qprojtool/qcommandline.cpp
/home/carlos/tmp/projects/projects/qprojtool/qcommandline.h
/home/carlos/tmp/projects/projects/qprojtool/qmake.cpp
/home/carlos/tmp/projects/projects/qprojtool/qmake.h
/home/carlos/tmp/projects/projects/qprojtool/qprojcommandline.cpp
/home/carlos/tmp/projects/projects/qprojtool/qprojcommandline.h
/home/carlos/tmp/projects/projects/qprojtool/tool.cpp
/home/carlos/tmp/projects/projects/qprojtool/tool.h
```

2.2 *Code Counters*

2.2.1 Example using "qpt" with “cloc” and “xargs”

```
$ qpt -A /home/carlos/code_quality/doxyanalyzer/doxyanalyzer.pro | xargs cloc

     336 text files.
     336 unique files.
       0 files ignored.

http://cloc.sourceforge.net v 1.53  T=6.0 s (56.0 files/s, 73957.7 lines/s)
-------------------------------------------------------------------------------
Language                     files          blank        comment           code
-------------------------------------------------------------------------------
C++                            125          33561          29699         284763
C/C++ Header                   209          10744          25787          58855
C                                2             51             55            231
-------------------------------------------------------------------------------
SUM:                           336          44356          55541         343849
-------------------------------------------------------------------------------
```

2.2.2 Example using "qpt" with “cloc” generating a temporary output file

```
The same result as the above can be obtained with 2 commands:

$ qt -A /home/carlos/code_quality/doxyanalyzer/doxyanalyzer.pro >allsources.txt

$ cloc --list-file=allsources.txt
```

2.3 *Using qpt as wrapper for Static Analysis tools*

Qpt has a configuration file that keeps basic information about the allowed tools to use.

This allows to call qpt just passing the tool to use and the Qt project file.

So far the tools are cppcheck and the KDE Source Code Checker krazy2 which source code can be downloaded from krazy2 gitorious repository.

2.3.1 Example using “qpt” with “cppcheck” tool on the command line

```
 $ qpt --tool=cppcheck qml-folderlistmodel.pro


 [qpt running]: /usr/bin/cppcheck  -I /home/carlos/qt_tests/common/inc -DCMNTRACE -DDEBUG -f -q --template '{file}:{line}:
 {message}' --inline-suppr --enable=style --enable=unusedFunction  --inconclusive --file-list=/tmp/qpt.Lh6413
 '
 <error file="/home/carlos/qt_tests/quick2/qml-folderlistmodel/dirmodel.cpp" line="434" id="missingInclude" severity="style"
 msg="Include file: &quot;dirmodel.moc&quot; not found."/>
 'dirmodel.h:125:Member variable 'DirModel::mShowDirectories' is in the wrong order in the initializer list.'
 '/home/carlos/qt_tests/quick2/qml-folderlistmodel/ioworkerthread.cpp:70:Statements following return, break, continue, goto or throw will
 never be executed.'
 '/home/carlos/qt_tests/quick2/qml-folderlistmodel/dirmodel.cpp:428:The function 'awaitingResults' is never used'
```

2.3.2 Example using “qpt” with “krazy2” tool on the command line

```
$ qpt --tool=/usr/local/Krazy2/bin/krazy2 /home/carlos/projects/qbiblioteca/qbiblioteca.pro

[qpt running]: /usr/local/Krazy2/bin/krazy2 --verbose --export textedit -

=>c++/captruefalse test in-progress.done
=>c++/constref test in-progress.done
=>c++/copyright test in-progress.done
=>c++/cpp test in-progress.done
=>c++/crashy test in-progress.done
=>c++/doublequote_chars test in-progress.done
=>c++/doxytags test in-progress.done
=>c++/dpointer test in-progress.done
=>c++/emptystrcompare test in-progress.done
=>c++/endswithnewline test in-progress.done
=>c++/explicit test in-progress.done
=>c++/foreach test in-progress.done
=>c++/i18ncheckarg test in-progress.done
=>c++/iconnames test in-progress.done
=>c++/includes test in-progress.done
=>c++/inline test in-progress.done
=>c++/license test in-progress.done
=>c++/normalize test in-progress.done
=>c++/nullstrassign test in-progress.done
=>c++/nullstrcompare test in-progress.done
=>c++/operators test in-progress.done
=>c++/passbyvalue test in-progress.done
=>c++/postfixop test in-progress.done
=>c++/qbytearray test in-progress.done
=>c++/qclasses test in-progress.done
=>c++/qmethods test in-progress.done
=>c++/qminmax test in-progress.done
=>c++/qobject test in-progress.done
=>c++/sigsandslots test in-progress.done
=>c++/spelling test in-progress.done
=>c++/staticobjects test in-progress.done
=>c++/strings test in-progress.done
=>c++/syscalls test in-progress.done
=>c++/typedefs test in-progress.done
/home/carlos/projects/qbiblioteca/src/src/acervo.cpp:39:using TRUE, TRUE and FALSE macros
/home/carlos/projects/qbiblioteca/src/src/acervo.cpp:40:using TRUE, TRUE and FALSE macros
/home/carlos/projects/qbiblioteca/src/src/mailCode/Common/ExecPathEnvironment.cpp:2 (:missing tags: email address, an acceptable
copyright
/home/carlos/projects/qbiblioteca/src/src/mailCode/Common/ExecPathEnvironment.cpp:0:), an acceptable copyright
/home/carlos/projects/qbiblioteca/src/src/mailCode/Common/ExecPathEnvironment.h:2 (:missing tags: email address, an acceptable
copyright
/home/carlos/projects/qbiblioteca/src/src/mailCode/Common/ExecPathEnvironment.h:0:), an acceptable copyright
/home/carlos/projects/qbiblioteca/src/src/mailCode/NetWork/Internet.cpp:2 (:missing tags: email address, an acceptable copyright
/home/carlos/projects/qbiblioteca/src/src/mailCode/NetWork/Internet.cpp:0:), an acceptable copyright
/home/carlos/projects/qbiblioteca/src/src/mailCode/NetWork/Internet.h:2
(**************************************************************************/:missing tags: email address, an
acceptable copyright
/home/carlos/projects/qbiblioteca/src/src/pesquisa.cpp:224:code that should be considered crashy
/home/carlos/projects/qbiblioteca/src/src/acervo.cpp:493:single-char QString operations for efficiency
/home/carlos/projects/qbiblioteca/src/src/acervo.cpp:494:single-char QString operations for efficiency
/home/carlos/projects/qbiblioteca/src/src/acervo.cpp:495:single-char QString operations for efficiency
/home/carlos/projects/qbiblioteca/src/src/acervo.cpp:861:single-char QString operations for efficiency
/home/carlos/projects/qbiblioteca/src/src/acervo.cpp:862:single-char QString operations for efficiency
/home/carlos/projects/qbiblioteca/src/src/acervo.cpp:863:single-char QString operations for efficiency
```

3 Using “qpt” with QtCreator (qtcreator)

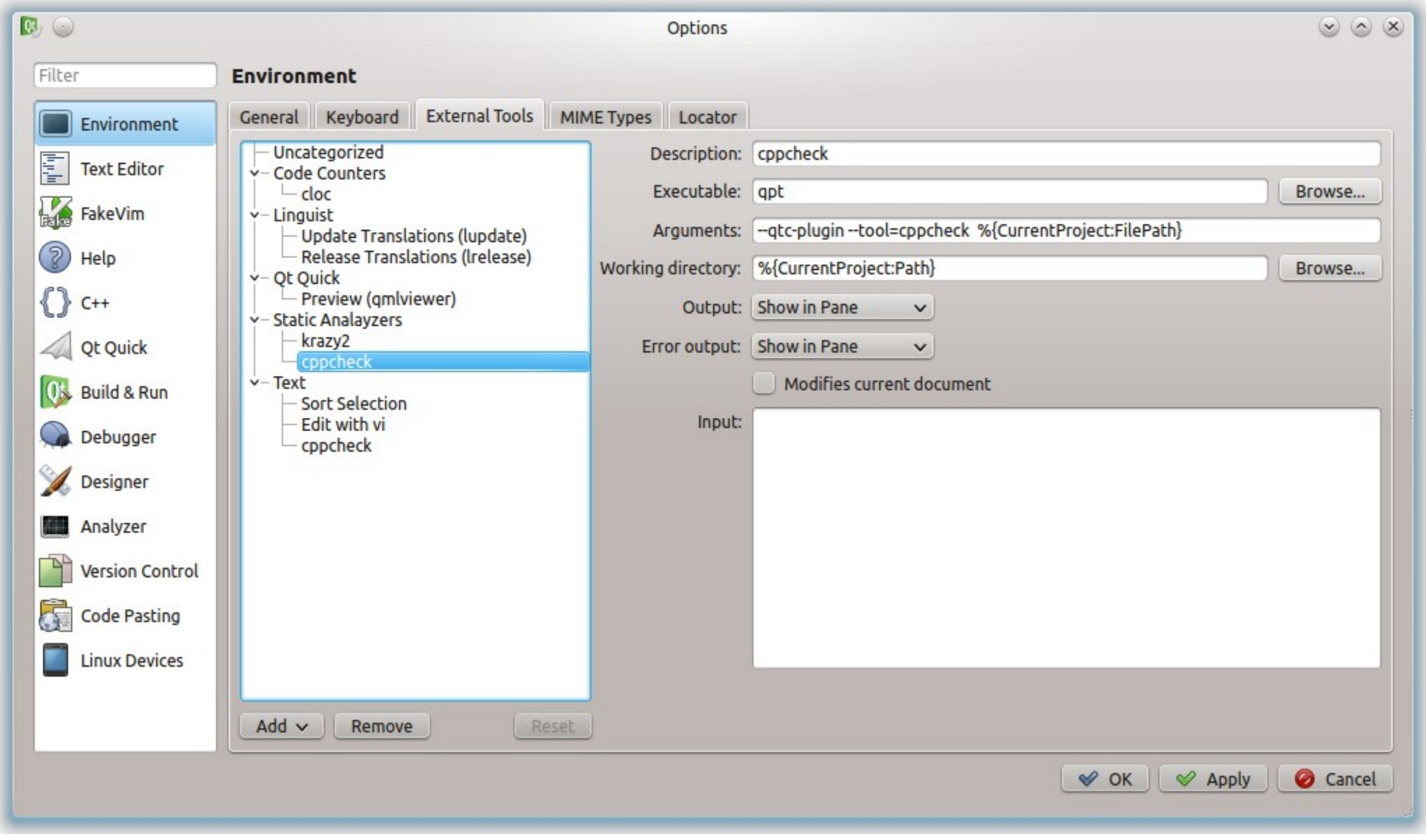
Inside QtCreator "qpt" works as an external tool, there is an option "***-q***" or "***-qtc-plugin***" that enable "qpt" act as a plugin for Static Analyzer tools.

To configure "qpt" in QtCreator use the menu *Tools->External->Configure...* then a dialog appears where we can enter new groups nd new tools. Basically only three fields are necessary:

1. Executable: qpt
2. Arguments: always the current project as variable ***%{CurrentProject:FilePath}*** plus “qpt” options
3. Working directory: always the project working directory as variable ***%{CurrentProject:Path}***

3.1 Configuring in QtCreator

3.1.1 Configurations for cppcheck



Environment

Description: cppcheck

Executable: qpt

Arguments: --qtc-plugin --tool=cppcheck %{CurrentProject:FilePath}

Working directory: %{CurrentProject:Path}

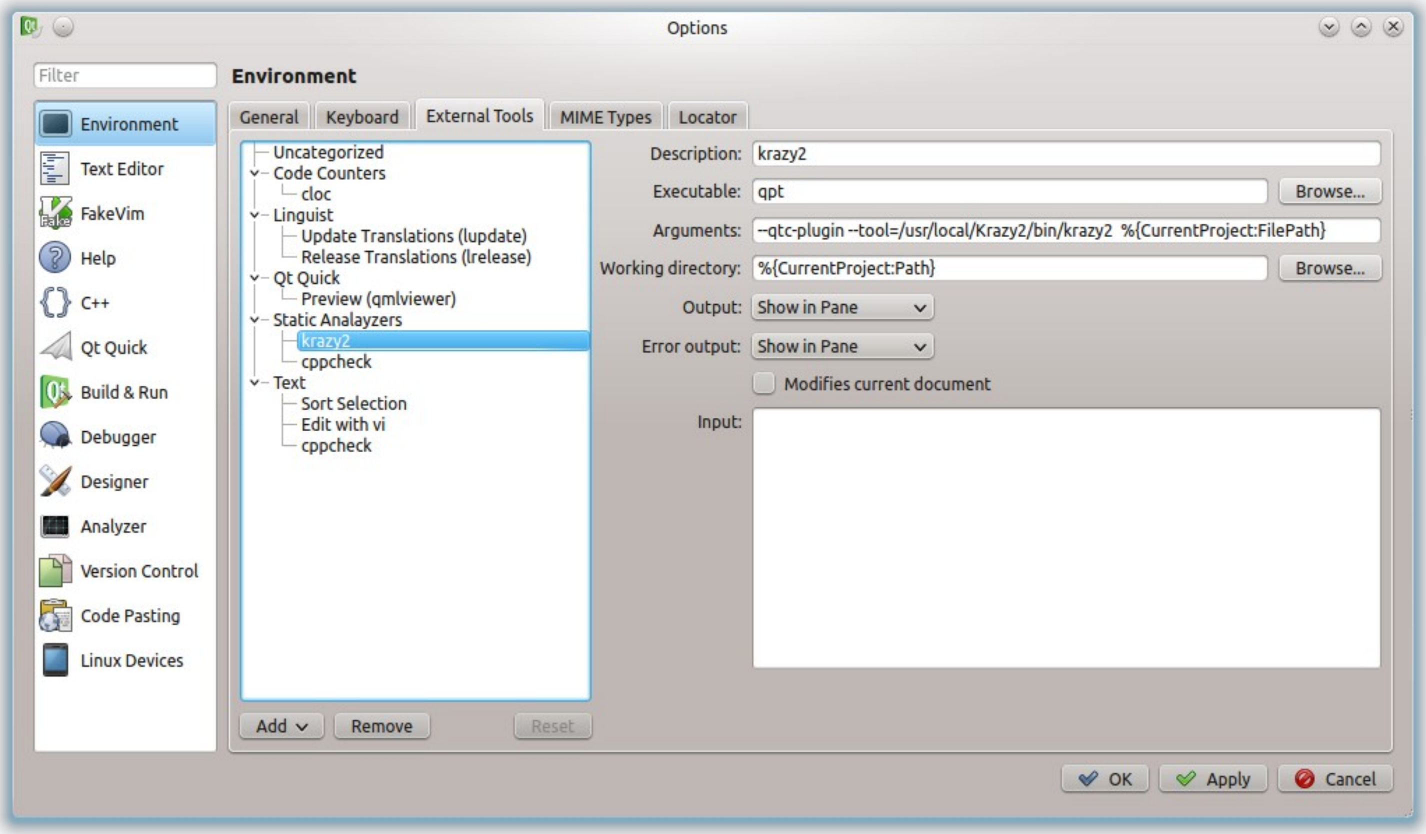
Output: Show in Pane

Error output: Show in Pane

Modifies current document

Input:

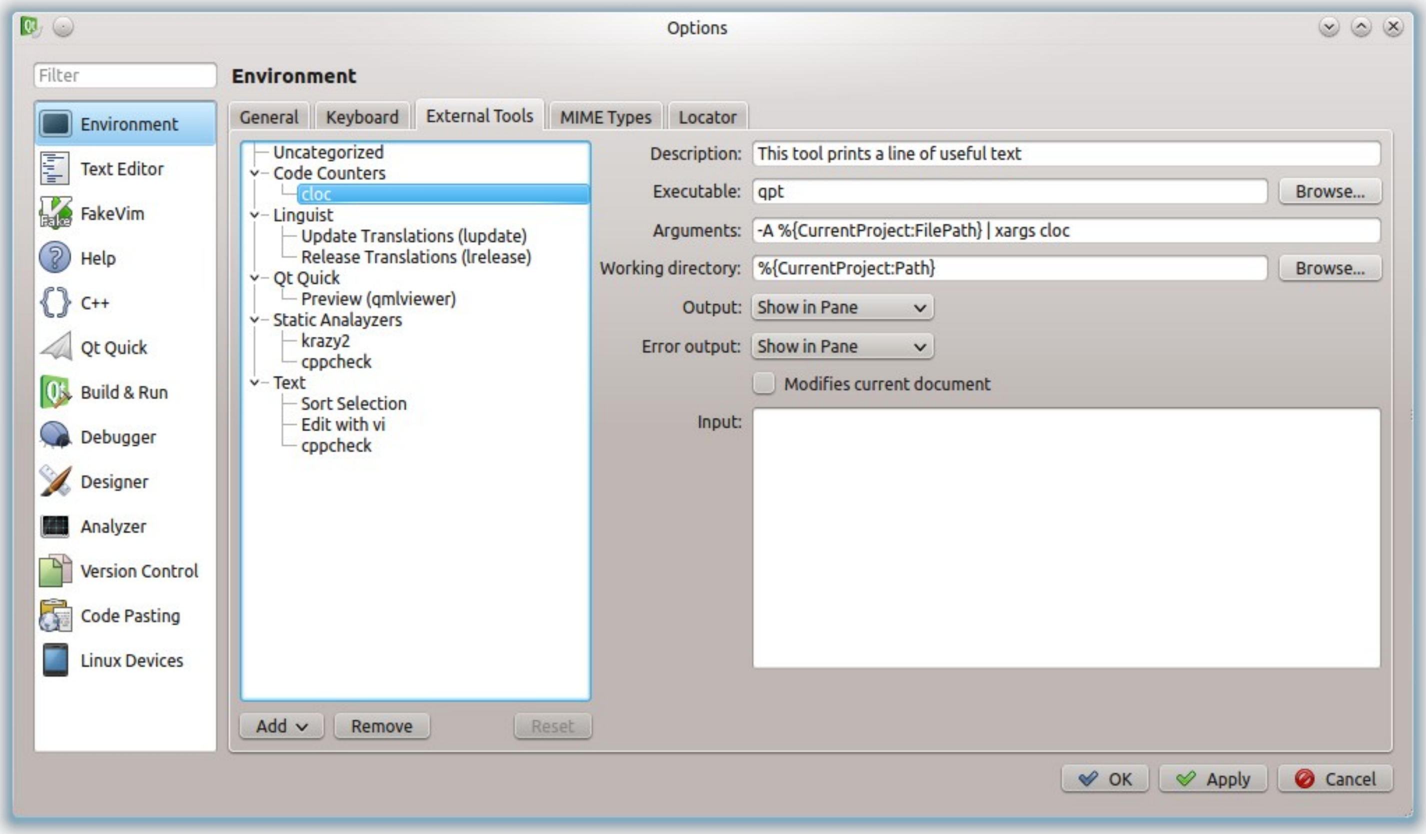
3.1.2 Configurations for krazy2



Environment

Description: krazy2
Executable: qpt
Arguments: --qtc-plugin --tool=/usr/local/Krazy2/bin/krazy2 %{CurrentProject:FilePath}
Working directory: %{CurrentProject:Path}
Output: Show in Pane
Error output: Show in Pane
Modifies current document: ☐
Input:

3.1.3 Configurations for cloc

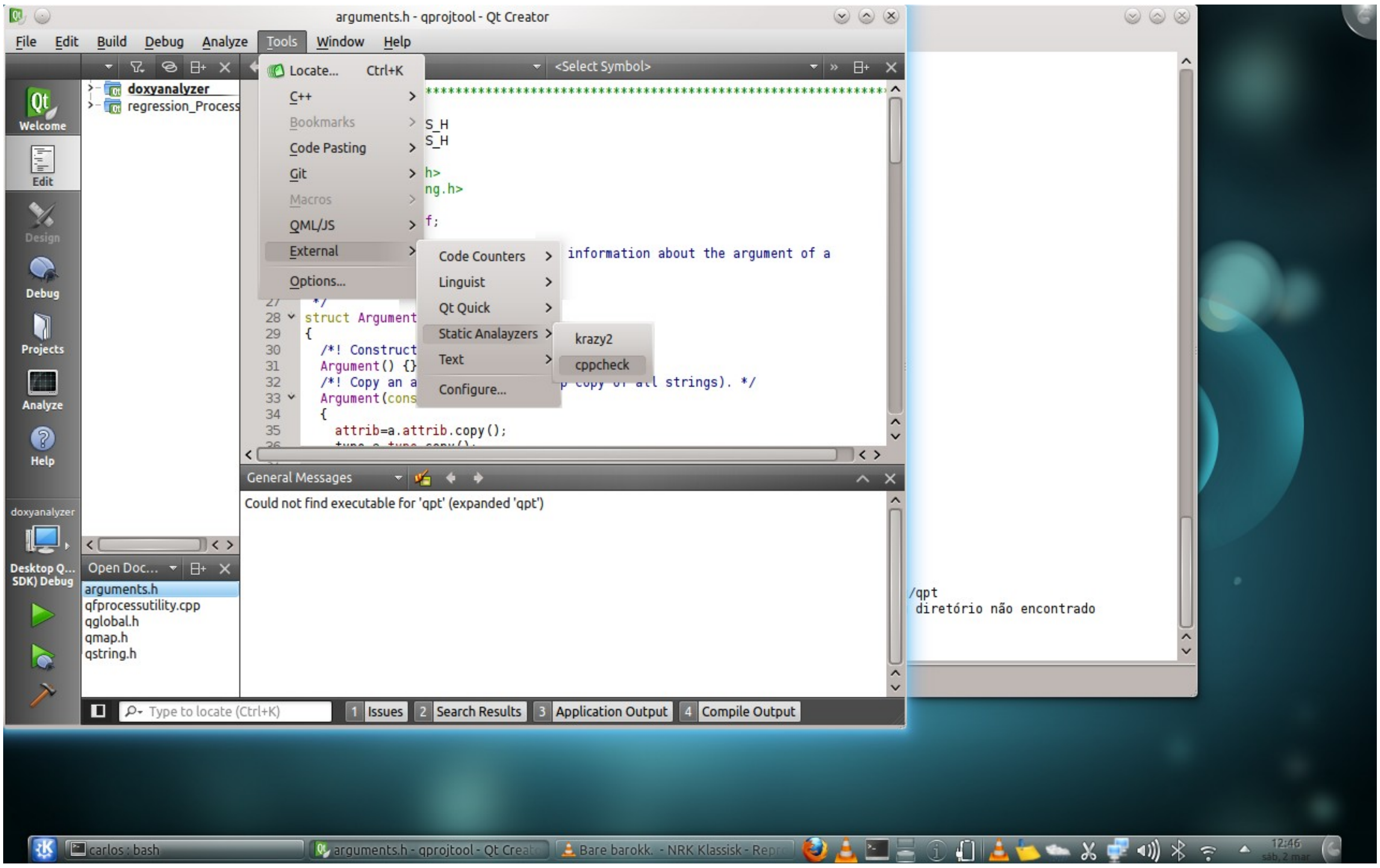


Environment

Description: This tool prints a line of useful text
Executable: qpt
Arguments: -A %{CurrentProject:FilePath} | xargs cloc
Working directory: %{CurrentProject:Path}
Output: Show in Pane
Error output: Show in Pane
Modifies current document: ☐
Input:

3.2 *Using “qpt” as External Tool*

Just use the menus over the current project.



```
S_H
S_H

h>
ng.h>

f;

 information about the argument of a

  */
struct Argument
{
  /*! Construct
  Argument() {}
  /*! Copy an a                p copy of all strings). */
  Argument(cons
  {
    attrib=a.attrib.copy();
```

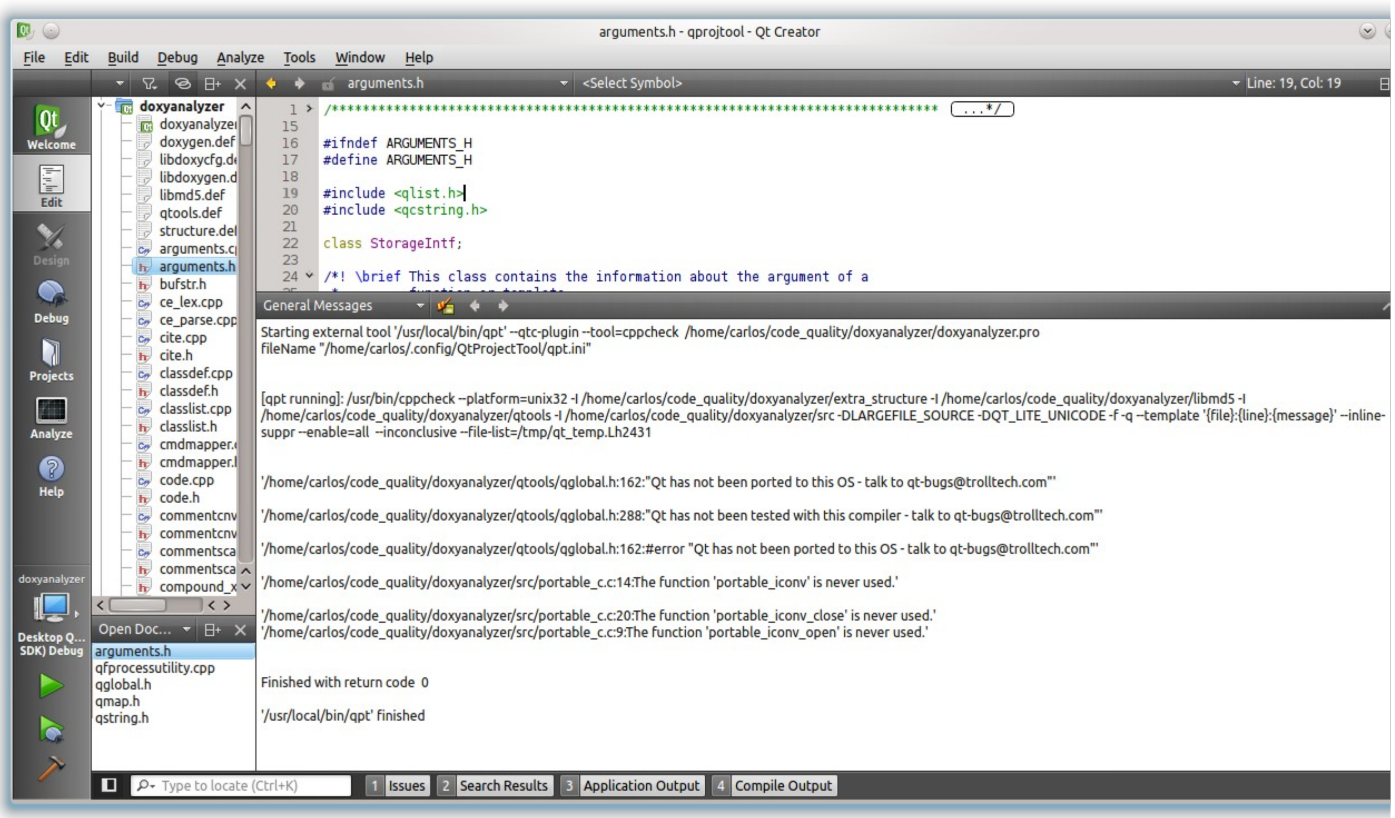
```
Could not find executable for 'qpt' (expanded 'qpt')
```

```
/qpt
 diretório não encontrado
```

3.3 *Cppcheck plugin in QtCreator*

3.3.1 General Messages pane

The normal output from cppcheck goes to “General Messages” pane

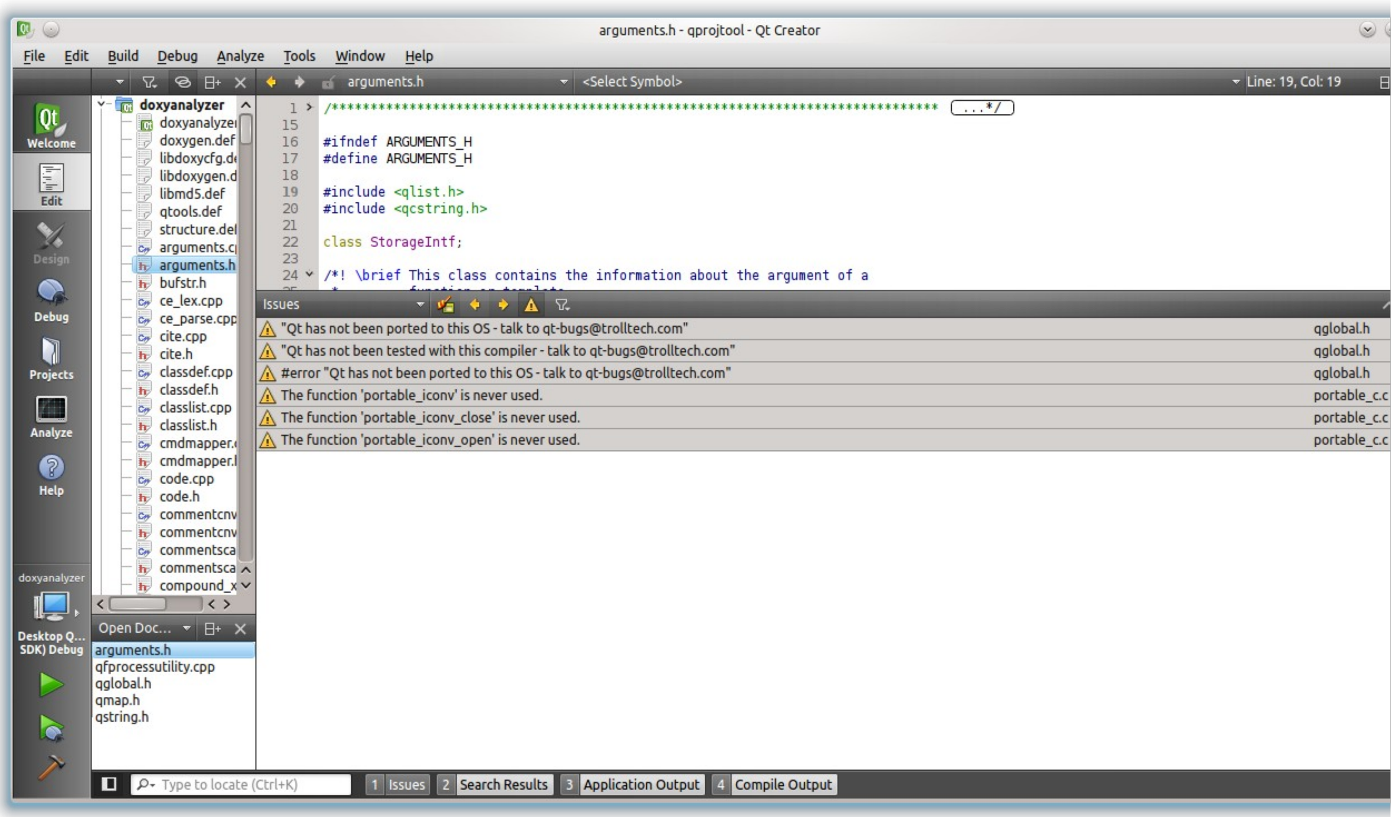


```
/*************************************************************************** ...*/

#ifndef ARGUMENTS_H
#define ARGUMENTS_H

#include <qlist.h>
#include <qcstring.h>

class StorageIntf;

/*! \brief This class contains the information about the argument of a
```

```
Starting external tool '/usr/local/bin/qpt' --qtc-plugin --tool=cppcheck  /home/carlos/code_quality/doxyanalyzer/doxyanalyzer.pro
fileName "/home/carlos/.config/QtProjectTool/qpt.ini"


[qpt running]: /usr/bin/cppcheck --platform=unix32 -I /home/carlos/code_quality/doxyanalyzer/extra_structure -I /home/carlos/code_quality/doxyanalyzer/libmd5 -I
/home/carlos/code_quality/doxyanalyzer/qtools -I /home/carlos/code_quality/doxyanalyzer/src -DLARGEFILE_SOURCE -DQT_LITE_UNICODE -f -q --template '{file}:{line}:{message}' --inline-
suppr --enable=all  --inconclusive --file-list=/tmp/qt_temp.Lh2431


'/home/carlos/code_quality/doxyanalyzer/qtools/qglobal.h:162:"Qt has not been ported to this OS - talk to qt-bugs@trolltech.com"'

'/home/carlos/code_quality/doxyanalyzer/qtools/qglobal.h:288:"Qt has not been tested with this compiler - talk to qt-bugs@trolltech.com"'

'/home/carlos/code_quality/doxyanalyzer/qtools/qglobal.h:162:#error "Qt has not been ported to this OS - talk to qt-bugs@trolltech.com"'

'/home/carlos/code_quality/doxyanalyzer/src/portable_c.c:14:The function 'portable_iconv' is never used.'

'/home/carlos/code_quality/doxyanalyzer/src/portable_c.c:20:The function 'portable_iconv_close' is never used.'
'/home/carlos/code_quality/doxyanalyzer/src/portable_c.c:9:The function 'portable_iconv_open' is never used.'


Finished with return code  0

'/usr/local/bin/qpt' finished
```

3.3.2 Issues pane

By clicking on a issue item, qtcreator allows fast code edition.

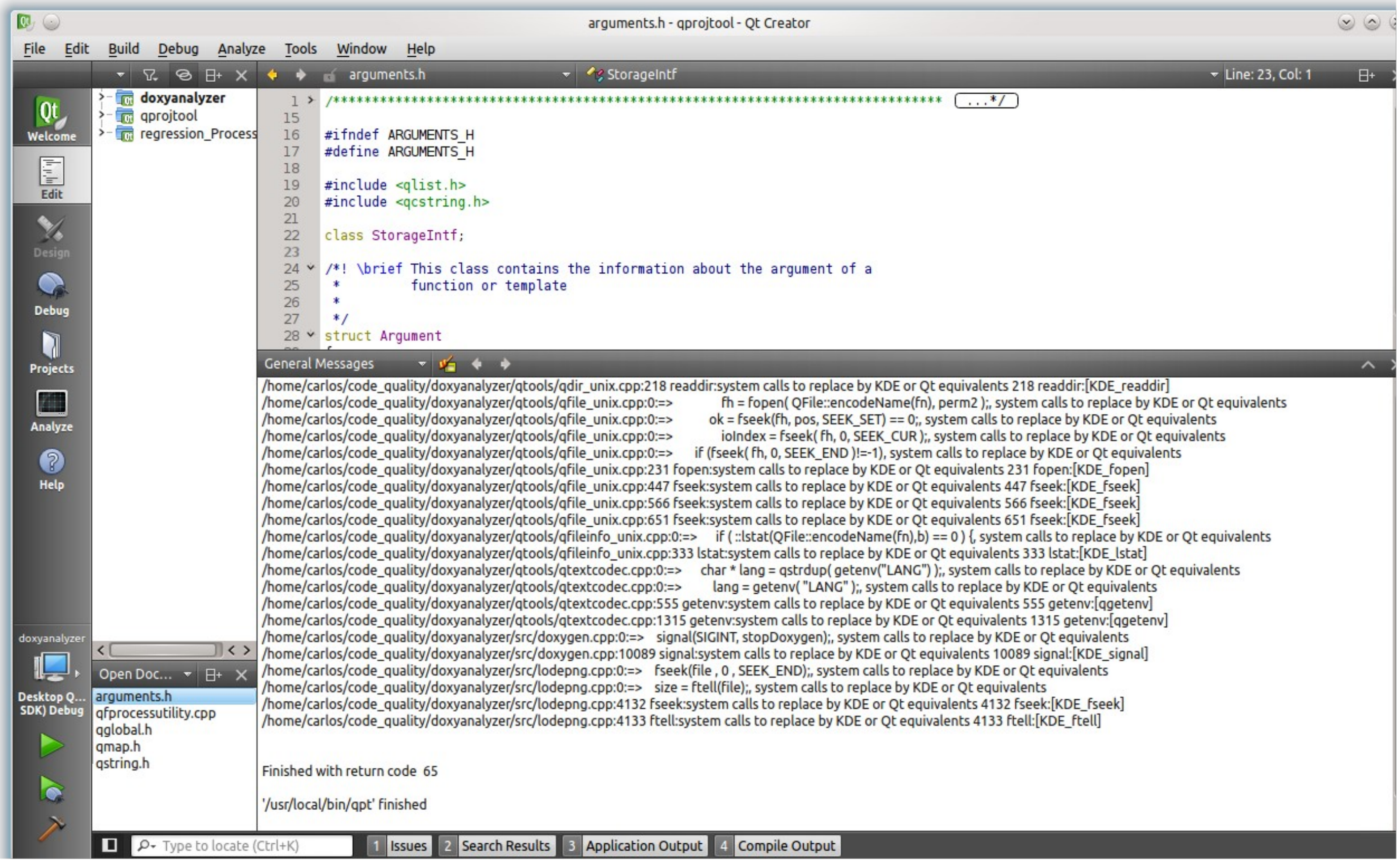


```
"Qt has not been ported to this OS - talk to qt-bugs@trolltech.com"              qglobal.h
"Qt has not been tested with this compiler - talk to qt-bugs@trolltech.com"      qglobal.h
#error "Qt has not been ported to this OS - talk to qt-bugs@trolltech.com"       qglobal.h
The function 'portable_iconv' is never used.                                     portable_c.c
The function 'portable_iconv_close' is never used.                               portable_c.c
The function 'portable_iconv_open' is never used.                                portable_c.c
```

arguments.h - qprojtool - Qt Creator
File Edit Build Debug Analyze Tools Window Help
Issues
1 Issues
2 Search Results
3 Application Output
4 Compile Output

3.4 *Krazy2 plugin in QtCreator*

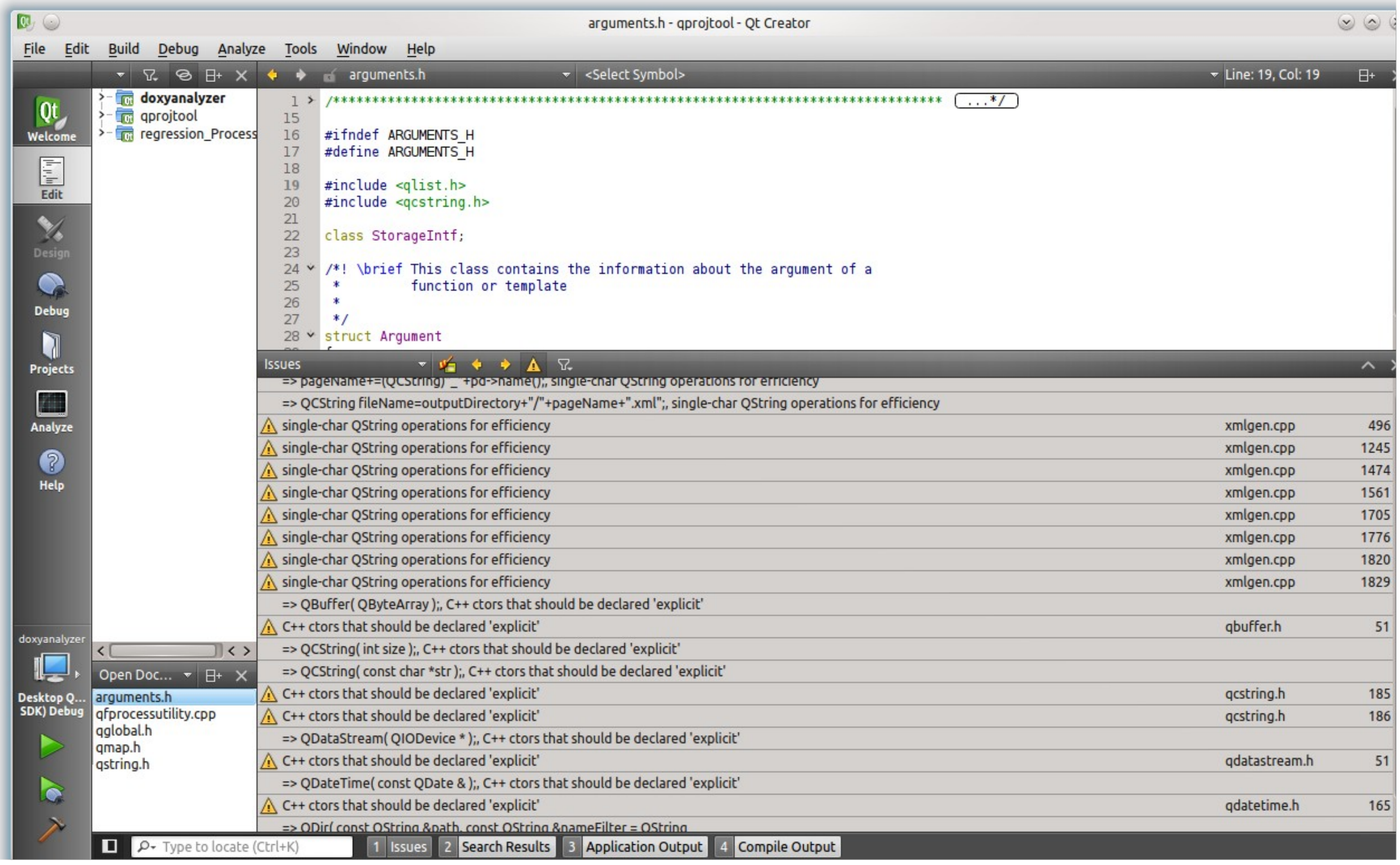
3.4.1 General Messages pane



```
/*************************************************************************** ...*/

#ifndef ARGUMENTS_H
#define ARGUMENTS_H

#include <qlist.h>
#include <qcstring.h>

class StorageIntf;

/*! \brief This class contains the information about the argument of a
 *         function or template
 *
 */
struct Argument
```

```
/home/carlos/code_quality/doxyanalyzer/qtools/qdir_unix.cpp:218 readdir:system calls to replace by KDE or Qt equivalents 218 readdir:[KDE_readdir]
/home/carlos/code_quality/doxyanalyzer/qtools/qfile_unix.cpp:0:=>          fh = fopen( QFile::encodeName(fn), perm2 );, system calls to replace by KDE or Qt equivalents
/home/carlos/code_quality/doxyanalyzer/qtools/qfile_unix.cpp:0:=>        ok = fseek(fh, pos, SEEK_SET) == 0;, system calls to replace by KDE or Qt equivalents
/home/carlos/code_quality/doxyanalyzer/qtools/qfile_unix.cpp:0:=>          ioIndex = fseek( fh, 0, SEEK_CUR );, system calls to replace by KDE or Qt equivalents
/home/carlos/code_quality/doxyanalyzer/qtools/qfile_unix.cpp:0:=>      if (fseek( fh, 0, SEEK_END )!=-1), system calls to replace by KDE or Qt equivalents
/home/carlos/code_quality/doxyanalyzer/qtools/qfile_unix.cpp:231 fopen:system calls to replace by KDE or Qt equivalents 231 fopen:[KDE_fopen]
/home/carlos/code_quality/doxyanalyzer/qtools/qfile_unix.cpp:447 fseek:system calls to replace by KDE or Qt equivalents 447 fseek:[KDE_fseek]
/home/carlos/code_quality/doxyanalyzer/qtools/qfile_unix.cpp:566 fseek:system calls to replace by KDE or Qt equivalents 566 fseek:[KDE_fseek]
/home/carlos/code_quality/doxyanalyzer/qtools/qfile_unix.cpp:651 fseek:system calls to replace by KDE or Qt equivalents 651 fseek:[KDE_fseek]
/home/carlos/code_quality/doxyanalyzer/qtools/qfileinfo_unix.cpp:0:=>     if ( ::lstat(QFile::encodeName(fn),b) == 0 ) {, system calls to replace by KDE or Qt equivalents
/home/carlos/code_quality/doxyanalyzer/qtools/qfileinfo_unix.cpp:333 lstat:system calls to replace by KDE or Qt equivalents 333 lstat:[KDE_lstat]
/home/carlos/code_quality/doxyanalyzer/qtools/qtextcodec.cpp:0:=>     char * lang = qstrdup( getenv("LANG") );, system calls to replace by KDE or Qt equivalents
/home/carlos/code_quality/doxyanalyzer/qtools/qtextcodec.cpp:0:=>       lang = getenv( "LANG" );, system calls to replace by KDE or Qt equivalents
/home/carlos/code_quality/doxyanalyzer/qtools/qtextcodec.cpp:555 getenv:system calls to replace by KDE or Qt equivalents 555 getenv:[qgetenv]
/home/carlos/code_quality/doxyanalyzer/qtools/qtextcodec.cpp:1315 getenv:system calls to replace by KDE or Qt equivalents 1315 getenv:[qgetenv]
/home/carlos/code_quality/doxyanalyzer/src/doxygen.cpp:0:=>   signal(SIGINT, stopDoxygen);, system calls to replace by KDE or Qt equivalents
/home/carlos/code_quality/doxyanalyzer/src/doxygen.cpp:10089 signal:system calls to replace by KDE or Qt equivalents 10089 signal:[KDE_signal]
/home/carlos/code_quality/doxyanalyzer/src/lodepng.cpp:0:=>   fseek(file , 0 , SEEK_END);, system calls to replace by KDE or Qt equivalents
/home/carlos/code_quality/doxyanalyzer/src/lodepng.cpp:0:=>   size = ftell(file);, system calls to replace by KDE or Qt equivalents
/home/carlos/code_quality/doxyanalyzer/src/lodepng.cpp:4132 fseek:system calls to replace by KDE or Qt equivalents 4132 fseek:[KDE_fseek]
/home/carlos/code_quality/doxyanalyzer/src/lodepng.cpp:4133 ftell:system calls to replace by KDE or Qt equivalents 4133 ftell:[KDE_ftell]


Finished with return code  65

'/usr/local/bin/qpt' finished
```

3.4.2 Issues pane



```
/*************************************************************************** ...*/

#ifndef ARGUMENTS_H
#define ARGUMENTS_H

#include <qlist.h>
#include <qcstring.h>

class StorageIntf;

/*! \brief This class contains the information about the argument of a
 *         function or template
 *
 */
struct Argument
```

```
   => pageName+=(QCString) _ +pd->name();, single-char QString operations for efficiency
   => QCString fileName=outputDirectory+"/"+pageName+".xml";, single-char QString operations for efficiency
⚠ single-char QString operations for efficiency                                   xmlgen.cpp       496
⚠ single-char QString operations for efficiency                                   xmlgen.cpp      1245
⚠ single-char QString operations for efficiency                                   xmlgen.cpp      1474
⚠ single-char QString operations for efficiency                                   xmlgen.cpp      1561
⚠ single-char QString operations for efficiency                                   xmlgen.cpp      1705
⚠ single-char QString operations for efficiency                                   xmlgen.cpp      1776
⚠ single-char QString operations for efficiency                                   xmlgen.cpp      1820
⚠ single-char QString operations for efficiency                                   xmlgen.cpp      1829
   => QBuffer( QByteArray );, C++ ctors that should be declared 'explicit'
⚠ C++ ctors that should be declared 'explicit'                                    qbuffer.h         51
   => QCString( int size );, C++ ctors that should be declared 'explicit'
   => QCString( const char *str );, C++ ctors that should be declared 'explicit'
⚠ C++ ctors that should be declared 'explicit'                                    qcstring.h       185
⚠ C++ ctors that should be declared 'explicit'                                    qcstring.h       186
   => QDataStream( QIODevice * );, C++ ctors that should be declared 'explicit'
⚠ C++ ctors that should be declared 'explicit'                                    qdatastream.h     51
   => QDateTime( const QDate & );, C++ ctors that should be declared 'explicit'
⚠ C++ ctors that should be declared 'explicit'                                    qdatetime.h      165
   => QDir( const QString &path, const QString &nameFilter = QString
```