

QGIS Application - Feature request #313

Support qgis on NetBSD

2006-09-28 09:19 AM - brook-nmsu-edu -

Status: Closed	
Priority: Low	
Assignee: Gary Sherman	
Category: Build/Install	
Pull Request or Patch supplied:	Resolution: fixed
Easy fix?: No	Copied to github as #: 10372
Description	
<p>In order to support building qgis on a [[NetBSD]] platform, I have found that at least the following patches are necessary. They are organized by the version of qgis. The first set is against the 0.7.4 release source code; the second is against the 0.8/preview2 source code. Both have been tested with both gcc v3 and v4. Note that the 0.8/preview2 patches include some compiler-dependent code, because gcc v3 and v4 handle isfinite() differently. Note also, that some of these patches correct errors discovered by gcc v4, which is stricter about enforcing standards than gcc v3.</p> <p>----</p> <p>qgis v0.7.4 patches</p> <p>----</p> <pre>--- acinclude.m4.orig 2005-08-14 00:44:41.000000000 -0600 +++ acinclude.m4 @@ -315,6 +315,15 @@ case "\${host}" in fi ;; + *netbsd*) + QT_LIBS="\$QT_LIB" + if test \$QT_IS_STATIC = yes && test \$QT_IS_EMBEDDED = no; then + QT_LIBS="\$QT_LIBS -L\$x_libraries -lXext -lX11 -lm -lSM -lICE -ldl -ljpeg -lpthread" + else + QT_LIBS="\$QT_LIBS -lpthread" + fi + ;; + *freebsd*) QT_LIBS="\$QT_LIB" if test \$QT_IS_STATIC = yes && test \$QT_IS_EMBEDDED = no; then --- plugins/geoprocessing/qgsdlgpgbuffer.h.orig 2005-06-02 23:02:14.000000000 -0600 +++ plugins/geoprocessing/qgsdlgpgbuffer.h @@ -29,7 +29,7 @@ class [[QgsDlgPgBuffer]]:public [[QgsDlgPgBuff]] //! Constructor [[QgsDlgPgBuffer]]([[QgisIface]] * _qi = 0, QWidget * parent = 0, const char *name = 0); //! Destructor - [[QgsDlgPgBuffer]]::~QgsDlgPgBuffer(); + ~QgsDlgPgBuffer(); //! Set the information label in the dialog void setBufferLabel(QString & lbl); //! Get the buffer distance</pre>	

```

--- plugins/georeferencer/datapointacetate.h.orig 2005-05-01 14:23:17.000000000 -0600
+++ plugins/georeferencer/datapointacetate.h
@@ -13,7 +13,7 @@
class [[DataPointAcetate]] : public [[QgsAcetateObject]] {
public:

- [[DataPointAcetate]]::DataPointAcetate(const [[QgsPoint]]& pixelCoords,
+ [[DataPointAcetate]](const [[QgsPoint]]& pixelCoords,
        const [[QgsPoint]]& mapCoords)
// UGLY!
: [[QgsAcetateObject]](const_cast<QgsPoint&>(pixelCoords)),

--- plugins/grid_maker/graticulecreator.h.orig 2005-05-29 14:04:47.000000000 -0600
+++ plugins/grid_maker/graticulecreator.h
@@ -19,8 +19,8 @@ class [[GraticuleCreator]]
        double theXEndPointDouble,
        double theYEndPointDouble);
~GraticuleCreator() {};
- DBFHandle [[GraticuleCreator]]::createDbf (QString theDbfName ) ;
- SHPHandle [[GraticuleCreator]]::createShapeFile(QString theFileName );
+ DBFHandle createDbf (QString theDbfName ) ;
+ SHPHandle createShapeFile(QString theFileName );
void writeDbfRecord (DBFHandle theDbfHandle, int theRecordIdInt, QString theLabel) ;
void writePoint(SHPHandle theShapeHandle, int theRecordInt, double theXDouble, double y );
//! Writes a WGS 84 .prj file for the generated grid

--- providers/gpx/qgsgpxprovider.h.orig 2005-02-11 16:36:49.000000000 -0700
+++ providers/gpx/qgsgpxprovider.h
@@ -142,7 +142,7 @@ public:
        const std::map<QString, QString>& attrs);

/** Adds one feature (used by addFeatures()) */
- bool [[QgsGPXProvider]]::addFeature(QgsFeature* f);
+ bool addFeature(QgsFeature* f);

/**Returns the default value for attribute @c attr for feature @c f. */
QString getDefaultValue(const QString& attr, [[QgsFeature]]* f);

--- src/qgslabel.h.orig 2005-06-03 03:42:55.000000000 -0600
+++ src/qgslabel.h
@@ -91,7 +91,7 @@ public:
void addRequiredFields ( std::list<int> *fields );

//! Available vector fields
- std::vector<QgsField> & [[QgsLabel]]::fields ( void );
+ std::vector<QgsField> & fields ( void );

//! Pointer to default attributes
[[QgsLabelAttributes]] *layerAttributes ( void );

--- src/qgsproject.h.orig 2005-05-11 21:21:56.000000000 -0600
+++ src/qgsproject.h
@@ -95,9 +95,9 @@ public:

```

```

    write()
*/
//@{
- bool [[QgsProject]]::dirty() const;
+ bool dirty() const;

- void [[QgsProject]]::dirty( bool b );
+ void dirty( bool b );
//@}

```

```

--- src/qgsprojectproperties.h.orig 2005-10-23 18:00:18.000000000 -0600
+++ src/qgsprojectproperties.h
@@ -60,7 +60,7 @@ public:
    /*! Accessor for projection */
    QString projectionWKT();
    /*! Indicates that the projection switch is on */
- bool [[QgsProjectProperties]]::isProjected();
+ bool isProjected();
public slots:
    /*!
**** Slot called when a new button (unit) is selected

```

```

--- src/qgsrasterlayerproperties.h.orig 2005-07-30 23:03:14.000000000 -0600
+++ src/qgsrasterlayerproperties.h
@@ -69,7 +69,7 @@ class [[QgsRasterLayerProperties]] : public
    /** \brief slot executed when the three band radio button is pressed. */
    void rbtnThreeBand_toggled( bool );
    /** \brief slot executed when user wishes to refresh raster histogram */
- void [[QgsRasterLayerProperties]]::pbnHistRefresh_clicked();
+ void pbnHistRefresh_clicked();
    /** Override the SRS specified when the layer was loaded */
    void pbnChangeSpatialRefSys_clicked();

```

```

----
qgis v0.8/preview2 patches
----

```

```

--- acinclude.m4.orig 2006-09-04 21:13:39.000000000 -0600
+++ acinclude.m4
@@ -401,6 +401,15 @@ case "${host}" in
    fi
    ;;

+ *netbsd*)
+   QT_LIBS="$QT_LIB"
+   if test $QT_IS_STATIC = yes && test $QT_IS_EMBEDDED = no; then
+     QT_LIBS="$QT_LIBS -L$x_libraries -lXext -lX11 -lm -lSM -lICE -ldl -ljpeg -lpthread"
+   else
+     QT_LIBS="$QT_LIBS -lpthread"
+   fi
+   ;;

```

```

+
*frebsd*)
  QT_LIBS="$QT_LIB"
  if test $QT_IS_STATIC = yes && test $QT_IS_EMBEDDED = no; then

--- src/plugins/grass/qgsgrassshell.cpp.orig 2006-04-12 09:11:02.000000000 -0600
+++ src/plugins/grass/qgsgrassshell.cpp
@@ -40,8 +40,12 @@ extern "C" {
#ifdef Q_OS_MACX
#include <util.h>
#else
+#ifdef +NetBSD+
+#include <util.h>
+#else
#include <pty.h>
#endif
+#endif
#include <sys/types.h>
#include <sys/stat.h>
#include <sys/ioctl.h>

--- src/providers/wms/qgsprovider.cpp.orig 2006-09-06 15:25:11.000000000 -0600
+++ src/providers/wms/qgsprovider.cpp
@@ -20,2 +20,6 @@

+#define _GLIBCPP_USE_C99 1
+#include <cmath>
+#undef _GLIBCPP_USE_C99
+
#include "qgslogger.h"
@@ -31,3 +35,2 @@

-#include <cmath>
#include <q3url.h>
@@ -1949,4 +1952,9 @@ bool [[QgsWmsProvider]]::calculateExtent()
    //make sure extent does not contain 'inf' or 'nan'
+#if +GNUC+ >= 4
+  if(!isfinite(extent.xMin()) || !isfinite((int)extent.yMin()) || !isfinite(extent.xMax()) || \
+    !isfinite((int)extent.yMax()))
+#else
    if(!std::isfinite(extent.xMin()) || !std::isfinite((int)extent.yMin()) || !std::isfinite(extent.xMax()) || \
      !std::isfinite((int)extent.yMax()))
+#endif
    {

```

History

#1 - 2006-09-28 10:01 AM - brook-nmsu-edu -

To support building qgis when multiple versions of qt are installed on the system (as is common when using pkgsrc on [[NetBSD]] or other systems), I suggest the following in addition to the patches above.

- Move the file create_qm_files.sh to create_qm_files.sh.in (i.e., create an autoconf template for this file)

- Apply the following patch to the file to allow substitution by configure.

```
--- create_qm_files.sh.in.orig
+++ create_qm_files.sh.in
@@ -9,8 +9,8 @@
 # create_qm_files.sh,v 1.4 2004/07/14 18:16:00 gsherman Exp
 echo Creating qmake project file
 # force the output name to be qgis_qm.pro
-qmake -project -o qgis_qm.pro
+@QTDIR@/bin/qmake -project -o qgis_qm.pro
 echo Creating qm files
-lrelease -verbose qgis_qm.pro
+@QTDIR@/bin/lrelease -verbose qgis_qm.pro
 echo Removing qmake project file
 rm qgis_qm.pro
```

- Apply the following patch to allow configure to substitute in this new template.

```
--- configure.in.orig 2006-09-12 22:06:03.000000000 -0600
+++ configure.in
@@ -448,6 +448,7 @@ AC_SUBST(DEBUG_QGIS)
 AC_CONFIG_FILES([
   Makefile
   Makefile.win.rules
+ create_qm_files.sh
   doc/Makefile
   doc/images/Makefile
   doc/install_guide/Makefile
```

#2 - 2006-09-29 11:24 PM - anonymous -

- Resolution set to fixed

- Status changed from Open to Closed

These patches have been applied in SVN commit:358800d7 (SVN r5888)

Thanks

#3 - 2006-09-30 04:57 AM - anonymous -

The changes to do with isfinite/std::isfinite have caused problems for other users (see developer mailing list). I have implemented a configure check for std::isfinite/isfinite and altered qgswmsprovider.cpp accordingly (SVN commit:b906049b (SVN r5891)). However, I don't have a BSD system to test this on, so please try it and see. I'm unsure if what I've done in qgswmsprovider will work as I intended, so any fixes will be welcome.

#4 - 2009-08-22 12:46 AM - Anonymous

