

# Building Quantum GIS mit Microsoft Visual C++ 2008 Express Edition auf Windows XP Professional

Dieses Tutorial erklärt wie man auf Windows XP Professional Quantum GIS in der aktuellen Version als Projekt in Microsoft Visual C++ 2008 Express Edition einbinden und im Folgenden installieren kann.

Vielen Dank an Tim Sutton, David Willis und Jürgen Fischer die mit dem MSVC Teil des Tutorials „Compiling QGIS From Source“ die Grundlage für dieses Tutorial geliefert haben.

## Schritt 1: Microsoft Visual C++ 2008 Express Edition und Microsoft Platform SDK for Windows Server 2003 R2

Microsoft Visual C++ 2008 Express kann hier heruntergeladen werden:

<http://www.microsoft.com/germany/express/download/downloaddetails.aspx?p=vcpp>

Die Installation erfolgt fast von alleine; MSDN und SQL Server müssen nicht installiert werden.

Microsoft Platform SDK for Windows Server 2003 R2 kann hier heruntergeladen werden:

<http://www.microsoft.com/downloads/details.aspx?FamilyId=0BAF2B35-C656-4969-ACE8-E4C0C0716ADB&displaylang=en#filelist>

Wählt aus folgenden Dateien die entsprechende aus:

PSDK-amd64.exe
PSDK-ia64.exe
PSDK-x86.exe

Wählt bei der Installation unbedingt das Standardverzeichnis und wählt im Folgenden diese Optionen:

Configuration Options	
+ Register Environmental Variables	(Yes)
Microsoft Windows Core SDK	
+ Tools	(Yes)
+ Tools (AMD 64 Bit)	(No unless this applies)
+ Tools (Intel 64 Bit)	(No unless this applies)
+ Build Environment	
+ Build Environment (AMD 64 Bit)	(No unless this applies)
+ Build Environment (Intel 64 Bit)	(No unless this applies)

+ Build Environment (x86 32 Bit)	(Yes)
+ Documentation	(No)
+ Redistributable Components	(Yes)
+ Sample Code	(No)
+ Source Code	(No)
+ AMD 64 Source	(No)
+ Intel 64 Source	(No)
Microsoft Web Workshop	(Yes)
+ Build Environment	(Yes)
+ Documentation	(No)
+ Sample Code	(No)
+ Tools	(No)
Microsoft Internet Information Server (IIS) SDK	(No)
Microsoft Data Access Services (MDAC) SDK	(Yes)
+ Tools	
+ Tools (AMD 64 Bit)	(No)
+ Tools (AMD 64 Bit)	(No)
+ Tools (x86 32 Bit)	(Yes)
+ Build Environment	
+ Tools (AMD 64 Bit)	(No)
+ Tools (AMD 64 Bit)	(No)
+ Tools (x86 32 Bit)	(Yes)
+ Documentation	(No)
+ Sample Code	(No)
Microsoft Installer SDK	(No)
Microsoft Table PC SDK	(No)
Microsoft Windows Management Instrumentation	(No)
Microsoft DirectShow SDK	(No)
Microsoft Media Services SDK	(No)
Debuggin Tools for Windows	(Yes)

In Microsoft Visual C++ Express müssen nun folgende Einstellungen unter Extras -> Optionen -> Projekte und Projektmappen -> VC++-Verzeichnisse getroffen werden:

Ausführbare Dateien:
C:\Programme\Microsoft Platform SDK for Windows Server 2003 R2\Bin
Includedateien:
C:\Programme\Microsoft Platform SDK for Windows Server 2003 R2\Include
C:\Programme\Microsoft Platform SDK for Windows Server 2003 R2\Include\atl

```
C:\Programme\Microsoft Platform SDK for Windows Server 2003
R2\Include\mfc
Bibliotheksdateien:
C:\Program Files\Microsoft Platform SDK for Windows Server 2003
R2\Lib
```

Schließt nun Microsoft Visual C++ Express und wechselt in den folgenden Ordner:

```
C:\Programme\Microsoft Visual Studio 9.0\Common7\Tools
```

Editiert hier die Datei vsvars32.bat und fügt folgende Zeilen nach den letzten Zeilen die mit „@set“ beginnen ein:

```
@set PATH=C:\Programme\Microsoft Platform SDK for Windows Server 2003
R2\Bin;%PATH%
@set INCLUDE=C:\Programme\Microsoft Platform SDK for Windows Server
2003 R2\Include;%INCLUDE%
@set INCLUDE=C:\Programme\Microsoft Platform SDK for Windows Server
2003 R2\Include\mfc;%INCLUDE%
@set LIB=C:\Programme\Microsoft Platform SDK for Windows Server 2003
R2\Lib;%LIB%
@set LIB=%LIB%;C:\OSGeo4W\lib
```

## Schritt 2: Wichtige Abhängigkeiten

Nun müssen wir uns um alle Abhängigkeiten von Quantum GIS kümmern. Glücklicherweise kann uns hier das OSGeo4W Package helfen, dieses kann hier heruntergeladen werden:

<http://download.osgeo.org/osgeo4w/osgeo4w-setup.exe>

Wählt die Fortgeschrittene Installation und bleibt auch hier beim Standardverzeichnis C:\OSGeo4W. Folgende Pakete habe ich installiert (es können mit Sicherheit einige Pakete weggelassen werden welche noch von vorher aktivierten Web-Paketten stammen, aber dies ist nun mal meine Konfiguration):

```
Commandline_Utilites
curl, gdal, gdal_dev, gdal16, libgeotiff, libpq, msys, netcdf, proj,
python, shapelib
Desktop Utilities
grass, qgis, qgis-dev
Libs
curl, expat, fcgi, fftw, freetype-mingw, fribidi, gdal, gdal-dev,
gdal16, gdal16-grass, geos, grass-devel, gsl-devel, gsl-libs, hdf4,
```

```
hdf5, iconv, libgeotiff, libjpeg, libjpeg-devel, libjpeg12, libmysql,
libpng, libpq, libtiff, libxml2, mingw-libs, msvcrt, netcdf, ogdi,
openssl, pdcurses, pdflib, proj, pyopengl, pyqt4, python-numpy,
python-qtgis, python-win32, qtgis-devel, qtgis-georef-gislab, qt4-devel,
qt4-doc, qt4-libs, shapelib, sip, sqlite3, tcltk, wxpython, xerces-c,
zlib
```

**Web**

Keine Pakete

**Web\_Applications**

Keine Pakete

Die Installation kann einige Zeit in Anspruch nehmen.

Zusätzlich müssen nun noch Flex und Bison heruntergeladen werden:

Flex: <http://gnuwin32.sourceforge.net/downloads/flex.php>

Bison: <http://gnuwin32.sourceforge.net/downloads/bison.php>

Bei der Installation kann entweder das Verzeichnis C:\OSGeo4W angegeben werden oder das Standardverzeichnis. Bei der Installation ins Standardverzeichnis müssen im Nachhinein folgende Ordner in C:\OSGeo4W kopiert werden:

```
bin
include
lib
share
```

Alle Abhängigkeiten müssen nun noch in die Umgebungsvariablen eingetragen werden. Öffnet dazu mit Rechtsklick auf den Arbeitsplatz das Fenster Eigenschaften. Auf dem Reiter „Erweitert“ findet ihr die Schaltfläche „Umgebungsvariablen“. Unter Systemvariablen müssen nun folgende Variablen eingetragen werden (wenn diese bereits vorhanden sind, müssen nur die fehlenden Einträge ergänzt werden):

```
INCLUDE
C:\Programme\Microsoft Platform SDK for Windows Server 2003
R2\Include\.

LIB
C:\Programme\Microsoft Platform SDK for Windows Server 2003 R2\Lib\.

LIB_DIR
C:\OSGeo4W
```

```
Path
```

```
C:\Programme\Microsoft Platform SDK for Windows Server 2003
R2\Bin\.;C:\Programme\Microsoft Platform SDK for Windows Server 2003
R2\Bin\WinNT\.; C:\Programme\Microsoft Visual Studio
9.0\VC\bin;C:\Programme\Microsoft Visual Studio
9.0\Common7\IDE;C:\Programme\Microsoft Visual Studio
9.0\Common7\Tools;C:\OSGeo4W\bin
```

### Schritt 3: Weitere Einstellungen in Microsoft Visual C++ Express

Öffnet das bereits bekannte Fenster unter Extras -> Optionen -> Projekte und Projektmappen -> VC++-Verzeichnisse und fügt folgende Einträge hinzu:

```
Ausführbare Dateien:
  C:\OSGeo4W\bin

Includedateien:
  C:\OSGeo4W\include
  C:\OSGeo4W\include\Qt
  C:\OSGeo4W\include\QtCore
  C:\OSGeo4W\include\QtGui
  C:\OSGeo4W\include\QtNetwork
  C:\OSGeo4W\include\QtSvg
  C:\OSGeo4W\include\QtXml
  C:\OSGeo4W\include\Qt3Support

Bibliotheksdateien:
  C:\OSGeo4W\lib
```

### Schritt 4: Python

Python in der Version 2.5.1. kann hier heruntergeladen werden:

<http://python.org/ftp/python/2.5.1/python-2.5.1.msi>

Für die Installation sind keine besonderen Einstellungen nötig.

Ladet euch nun SIP unter folgendem Link runter und entpackt es in den Ordner C:\dev\cpp:

<http://www.riverbankcomputing.co.uk/software/sip/download>

Öffnet nun im MSVC über Extras -> Visual Studio 2008 Command Prompt die Eingabekonsole, wechselt in den Ordner in den ihr SIP extrahiert habet und gebt folgendes ein:

```
c:\python25\python configure.py -p win32-msvc2005
nmake
nmake install
```

## Schritt 5: CMake & SVN

CMake 2.6.4. kann hier heruntergeladen werden:

<http://www.cmake.org/cmake/resources/software.html>

Die Installation birgt keine Probleme.

SVN kann hier heruntergeladen werden und muss in C:\Programm\svn entpackt werden:

<http://subversion.tigris.org/downloads/1.4.5-win32/apache-2.2/svn-win32-1.4.5.zip>

Fügt nun noch die beiden Verzeichnis zur Umgebungsvariable PATH hinzu (siehe Ende Schritt 2):

```
C:\Programme\CMake 2.6\bin
C:\Programme\svn\bin
```

## Schritt 6: Quantum GIS Checkout

Öffnet über Start -> Ausführen eine „cmd.exe“, wechselt in das Verzeichnis C:\dev\cpp und checkt den aktuellen QGIS Quellcode mit folgendem Befehl aus:

```
svn co https://svn.osgeo.org/qgis/trunk/qgis
```

Die eventuelle nachfolgende Nachricht könnt ihr mit einem “p” bestätigen.

## Schritt 7: Building und Installation

Führt nun in der Eingabeaufforderung folgende Schritte durch:

```
cd qgis
mkdir build
cd build
cmakesetup ..
```

Nach dem Klick auf „Configure“ müsst ihr nur noch Microsoft Visual Studio 9.0 auswählen und der Rest des CMakeSetups sollte automatisch laufen. Individuelle Anpassungen an den Einstellungen können nach der Konfiguration vorgenommen werden.

Jetzt kann wieder MSVC geöffnet werden und dort über Datei -> Öffnen -> Projekt/Projektmappe die soeben erstellte Datei „qgisX.X.X.sln“ (je nach aktueller Version) geöffnet werden.

Wechselt in der Toolbar nun noch von „Debug“ auf „Release“. Im Projektmappen-Editor auf der linken Seite kann nun mit Rechtsklick auf „ALL\_BUILD“ der Buildvorgang über „Erstellen“ gestartet werden. Nachdem hier alles ohne Fehler verlaufen ist, wird mit Rechtsklick -> Erstellen auf „INSTALL“ die Quantum GIS Version installiert (standardmäßig in C:\Programme\qgisX.X.X).

Quantum GIS kann nun wie gewohnt gestartet werden.

**Viel Spaß beim Entwickeln!**

**Florian Hillen**

**Tipp zum Schluss:** Alle Projekte aus dem Projektmappen-Editor können einzeln „erstellt“ werden. Wenn ihr also Änderungen an Plugins vorgenommen habt, reicht es diese einzeln zu builden und danach über INSTALL die Änderungen zu übernehmen.