Bilden der Qt Library am Beispiel von Qt 4.8.5 und Visual Studio 2013
(leider momentan ohne Webkit):

* „qt-everywhere-opensource-src-4.8.5.zip“ runterladen und int c:\(QtDir) entpacken
* „strawberry-perl-5.18.1.1-32bit.msi“ runterladen und installieren
* „ActivePython-2.7.5.6-win32-x86.msi“ runterladen und installieren
* Unter c:\(Qtdir)\mkspecs\win32-msvc2012\qmake.conf die Zeile:
QMAKE\_COMPILER\_DEFINES += \_MSC\_VER=1700 WIN32
zu:
QMAKE\_COMPILER\_DEFINES += \_MSC\_VER=1800 WIN32
ändern
* Volle Berechtigung auf den Ordner c:\(Qtdir)\ geben und bin Ordner löschen
* Qt Library „patchen“ die Datei unter „(Qt install Dir)/src/3rdparty/javascriptcore/JavaScriptCore/wtf/MathExtras.h“ modifizieren:
	+ Ca. Zeile 99:
	ist : #if COMPILER(MSVC) || COMPILER(RVCT)
	soll : #if (COMPILER(MSVC) && \_MSC\_VER < 1800) || COMPILER(RVCT)
	+ Ca. Zeile 124:
	ist : #if COMPILER(MSVC)
	soll : #if COMPILER(MSVC) && \_MSC\_VER < 1800
	+ Ca. Zeile 128:
	#if \_MSC\_VER < 1800
	inline bool signbit(double num) { return \_copysign(1.0, num) < 0; }
	#endif
* Path Systemvariable und QTDIR ändern, damit sie auf bin von neu zu erstellendem Qt verweist
* Command Promp öffnen (cmd)
* Perl Systempfad anfügen #set PATH=c:\strawberry\perl\bin;%PATH%
* Python Systempfad anfügen #set PATH=c:\python27;%PATH%
* Vcvarsall.bat ausführen:
	+ #cd C:\Program Files\Microsoft Visual Studio 12.0\VC
	+ #vcvarsall.bat
* Nach Qt dir wechseln:
	+ #cd c:\(Qtdir)\
* Configuration setzen und nmake starten:
	+ #configure.exe -mp -opensource -no-webkit -qt-zlib -qt-libpng -qt-libjpeg -qt-libtiff -qt-libmng -nomake demos -nomake examples -platform win32-msvc2012
	+ Lizenztyp wählen und ‚y‘ für akzeptieren eingeben
	+ #nmake //Achtung dauert extrem lange

Quellnachweis für configure Flags:
http://qt-project.org/doc/qt-4.8/configure-options.html