

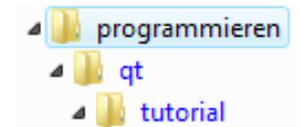
## Erste Schritte mit dem Qt Creator

Diese Beschreibung wendet sich an Anfänger, die noch nie mit der Entwicklungsumgebung (IDE) Qt Creator gearbeitet haben. Sie ist nicht für Programmierer gedacht, die schon mit einigen IDEs gearbeitet haben und sich deshalb im Qt Creator schnell alleine zurechtfinden. Fangen wir also klein an. Diese Dokumentation wurde auf Windows Vista erstellt. Andere grafische Systeme werden etwas andere Darstellungen bieten. Hier werden auch keine Konsolenkommandos eingesetzt.

Hinweis: Mit dem Qt Creator kann man auch Konsolenanwendungen erstellen.

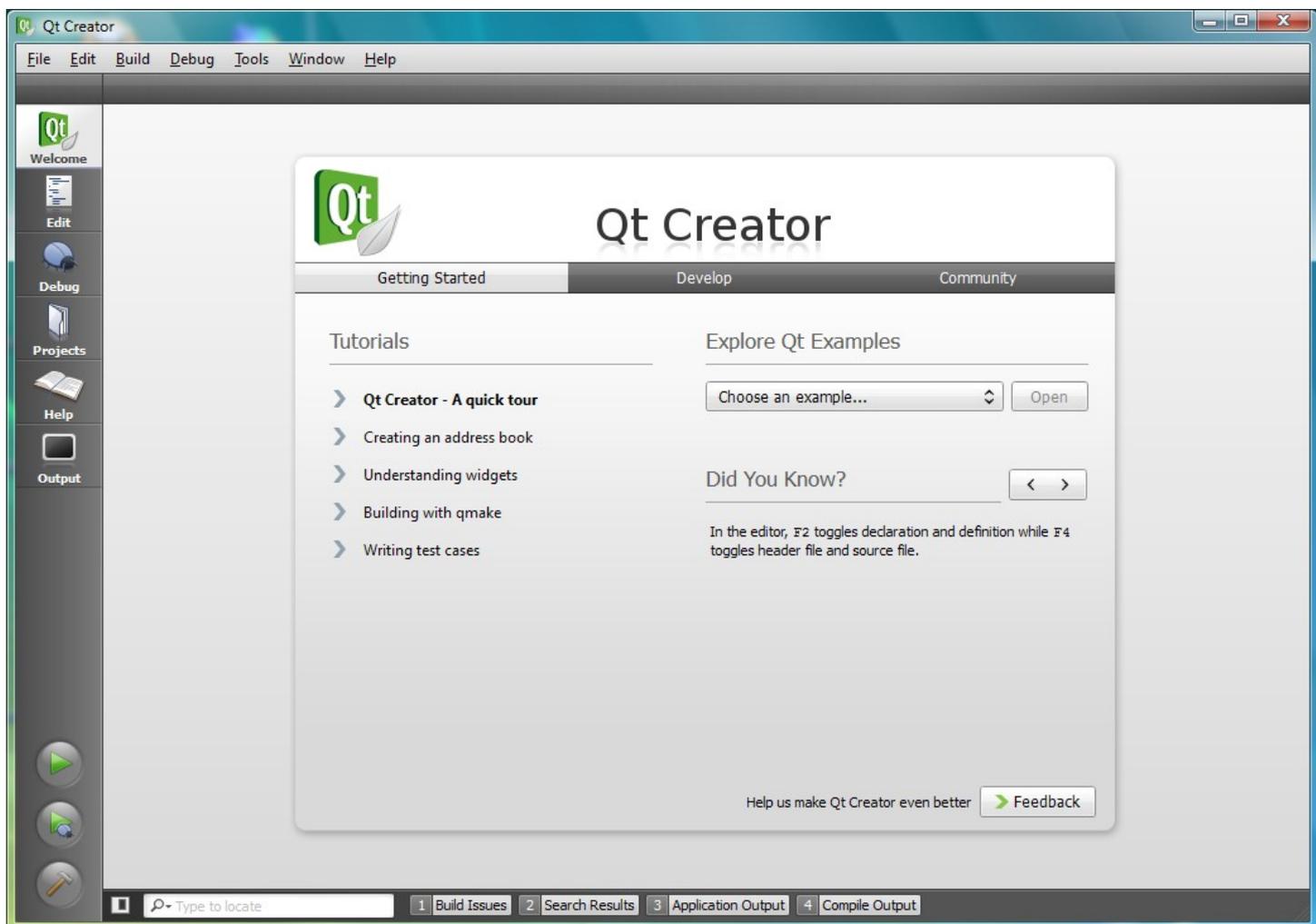
### Vorbereiten eines Teildateibaums

Damit Sie die Dateien zu den zu erstellenden Anwendungen leicht finden, sollten Sie zuerst auf Ihrer Festplatte einen Teildateibaum anlegen. Rechts ist ein Beispiel dazu abgebildet. Der Pfad zum Ordner tutorial wird im weiteren verwendet.



### Starten des Qt Creator

Es öffnet sich das folgende Anfangsfenster.



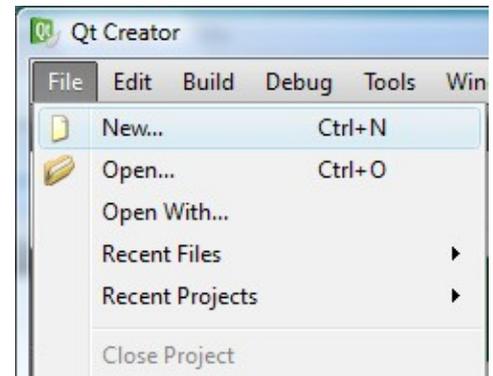
Wenn Sie bereits einige kleine GUI-Anwendungen erstellt haben, können Sie sich in den hier angebotenen Tutorials umsehen. Für echte Neulinge bietet sich das aber noch nicht an. Hier soll der Einstieg an Hand des Anfänger-Tutorials gezeigt werden.

Das Anfänger-Tutorial ist im WWW zu finden unter <http://doc.trolltech.com/4.2/tutorial.html>.

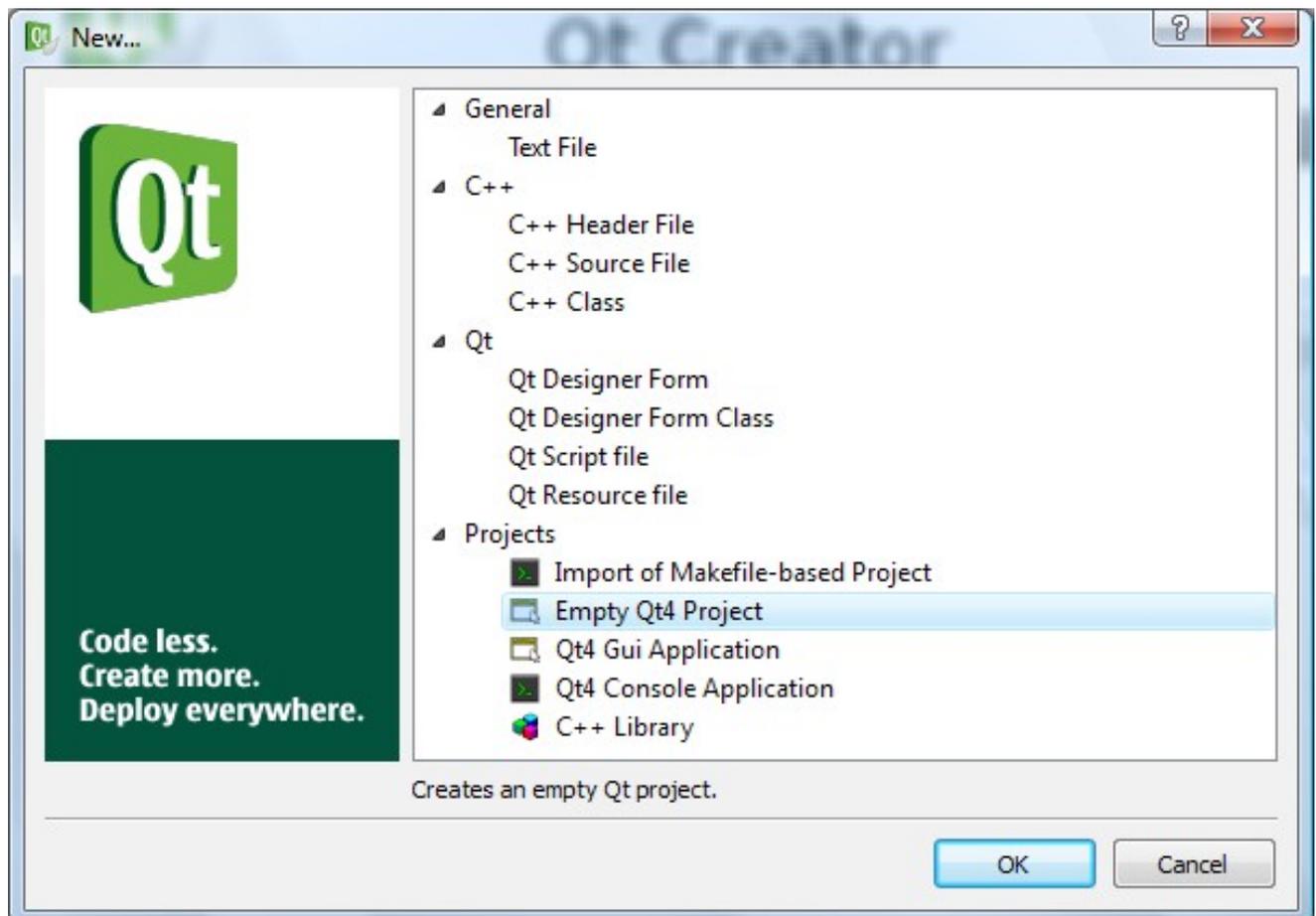
## Das erste Projekt „t1“ des Anfänger-Tutorials

Das erste Projekt des Tutorials wird dort „Hello World!“ genannt, ein durchaus gebräuchlicher Name. Das Projekt soll aber nicht „HelloWorld!“ sondern „t1“ genannt werden, das zweite Projekt nennen wir dann „t2“...

Sie lassen die im Qt Creator angebotenen Tutorials an dieser Stelle unbenutzt und legen sofort ein zunächst leeres Projekt „t1“ an. Dazu verwenden Sie das Hauptmenü oben im Fenster. Neue Dinge legt man mit **File/New...** an, siehe Abbildung rechts.



Danach öffnet sich ein Dialogfenster, in welchem auszuwählen ist, von welcher Art das neu Anzulegende sein soll. Hier brauchen Sie ein leeres Qt4 Projekt: „Empty Qt4 Project“



Mit OK geht es weiter zum nächsten Dialog. Darin tragen Sie hinter „Name“ den Projektnamen (t1) ein. Hinter „Create in“ notieren Sie den Pfad zum vorhin angelegten Tutorial-Ordner oder suchen diesen im Dateibaum mit Browse...

Name:

Create in:

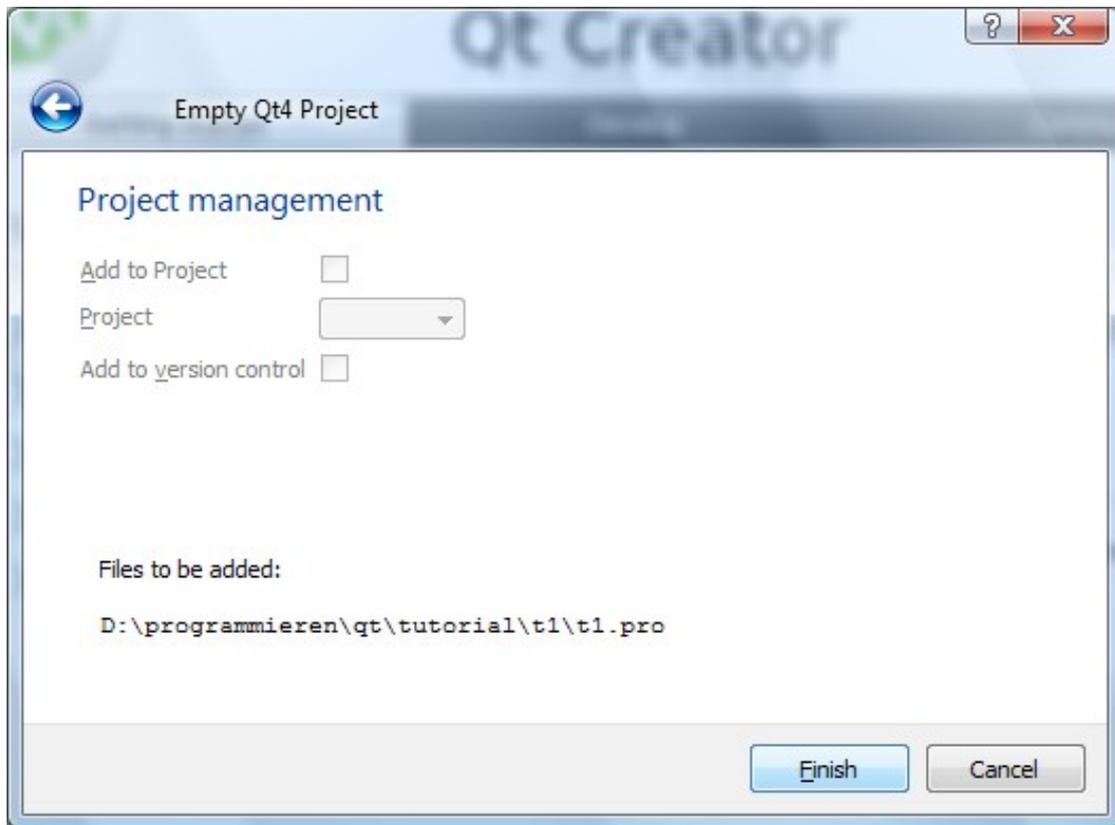
Wenn beide Eintragungen richtig vorliegen, klicken Sie auf den Button „Next“!

Dann öffnet sich ein letzter Dialog (s. Folgeseite), in welchem an dieser Stelle nichts geändert werden kann und auch nichts zu ändern ist. Es bleibt nur zu prüfen, ob der Pfad für die hinzuzufügende Datei richtig ist.

Files to be added:

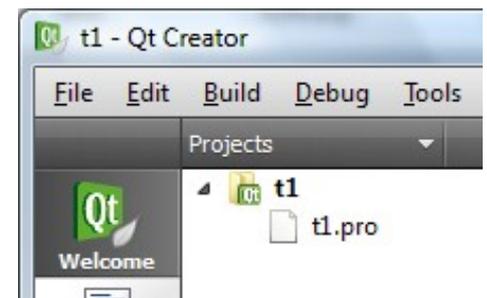
Bsp.: D:\programmieren\qt\tutorial\t1\t1.pro

„t1.pro“ ist die Beschreibungsdatei für unser erstes Projekt.



Nach einem Klick auf den Button „Finish“, wird das Projekt angelegt. Im Qt Creator Fenster finden Sie nun unter „Projects“ oben links das eingetragene Projekt. Es ist noch leer. Um die erste GUI-Anwendung zu erstellen, fehlt noch der Quelltext, auch „Sourcecode“ genannt.

Der Sourcecode wird in einer Quelltextdatei (Source File) gespeichert, die Sie als nächstes anlegen und dem leeren Projekt hinzufügen werden.



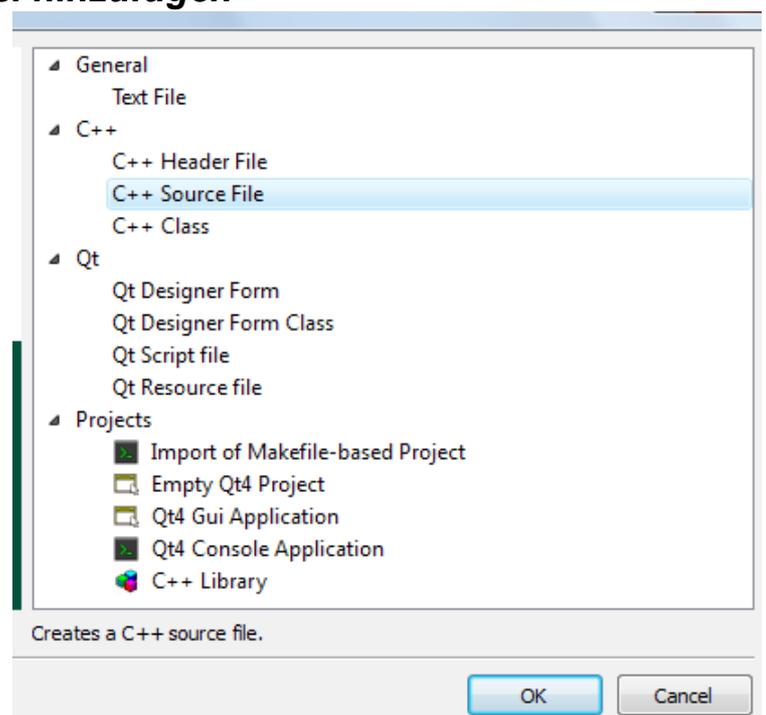
### **Dem Projekt eine C++ Quelltextdatei hinzufügen**

Hierzu verwenden Sie wieder das Menü mit File/New...

Im sich öffnenden Dialog wählen Sie „C++ Source File“ und bestätigen mit OK!

Nach File/New... erhalten Sie immer dieses Dialogfenster, in dem Sie die Art der anzulegenden Datei auswählen müssen. Sie sollten also immer wissen, was Sie gerade neu anlegen lassen wollen.

Vielleicht werden Sie versucht sein, eine „Qt4 Gui Application“ anzulegen. Dann tun Sie es einfach! Vermutlich werden Sie von den Texten, die Ihnen danach begegnen, etwas verwirrt sein. Dann werden Sie aber wohl verstehen, warum wir mit einem leeren Projekt begonnen haben. Ein nicht gewünschtes Projekt kann leicht wieder entfernt werden.



Nach OK öffnet sich der nächste Dialog. Dort tragen Sie den Namen der neuen Quelltextdatei ein! Die Endung „.cpp“ können Sie weglassen, sie wird automatisch hinzugefügt. Überprüfen Sie den Pfad! Falls dieser nicht wie gewünscht ist, ändern Sie ihn ab!

### Choose the location

Name:

Path:

Achten Sie darauf, dass der Pfad zum gewünschten Projektordner führt. Sonst wird es früher oder später ziemlich chaotisch. Am besten ist es, wenn jedes Projekt in einem eigenen Projektordner liegt.

Nach einem Klick auf den Button „Next“ öffnet sich ein neuer Dialog. In diesem können Sie die neue Quelltextdatei einem Projekt zuordnen. Wenn Sie das Häkchen hinter „Add to Project“ entfernen, wird diese Datei keinem Projekt zugeordnet, was hier unsinnig wäre.

Wenn Sie mehrere Projekte in Ihrer Session verwalten, können Sie hinter „Project“ das gewünschte Projekt auswählen, dem die neue Datei hinzugefügt werden soll. In Ihrem Fall wird vermutlich ausschließlich „t1.pro“ verfügbar sein.

Überprüfen Sie bitte den Pfad unter „Files to be added:“! Sollte er nicht wie gewünscht sein, brechen Sie das Anlegen der Datei ab und beginnen die Prozedur noch einmal!

Wenn alles in Ordnung ist, klicken Sie auf den Button „Finish“! Danach erscheint der Name der neuen Quelltextdatei (t1.cpp) im Projekt t1 – links oben unter „Projects“.

### Project management

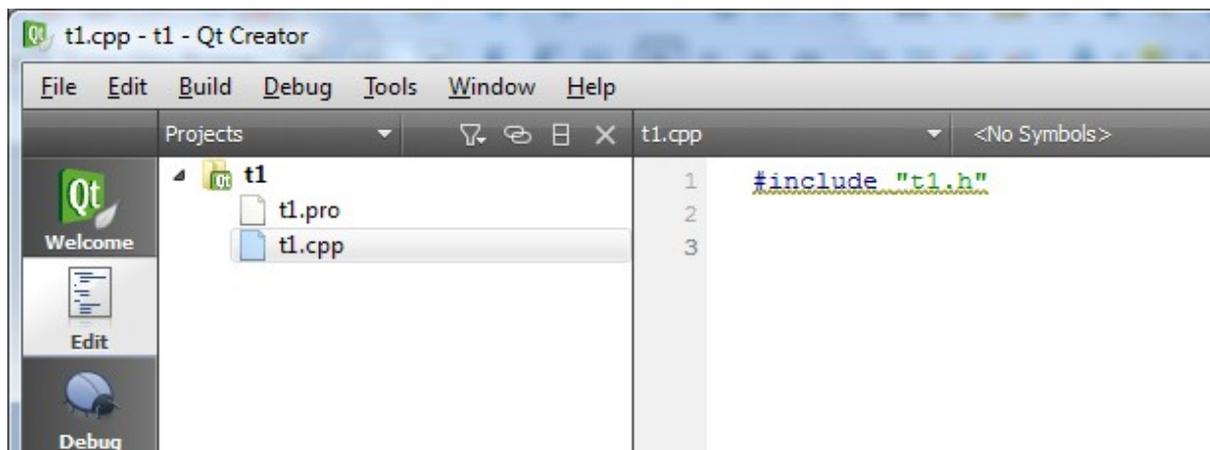
Add to Project

Project

Add to version control

Files to be added:

D:\programmieren\qt\tutorial\t1\t1.cpp



Im rechten Teilfenster ist der aktuelle Inhalt dieser Quelltextdatei dargestellt. Sie finden hier eine einzelne #include-Anweisung, die nicht gebraucht wird. Entfernen Sie diese Anweisung! In das nun leere Editierfenster tragen Sie den Quelltext aus dem Tutorial zu „Hello World!“ ein!

Faule Menschen neigen dazu, hier mit copy & paste (kopieren und einkleben) zu arbeiten. Sie haben aber mehr davon, wenn Sie den Quelltext akribisch abschreiben ;-) - statt des Textes „Hello World!“ können Sie selbstverständlich auch andere Texte einsetzen. Bevor Sie weitere Experimente darin anstellen, sollte die kleine Anwendung zum Laufen gebracht werden.

## Eine geschriebene Anwendung zum Laufen bringen



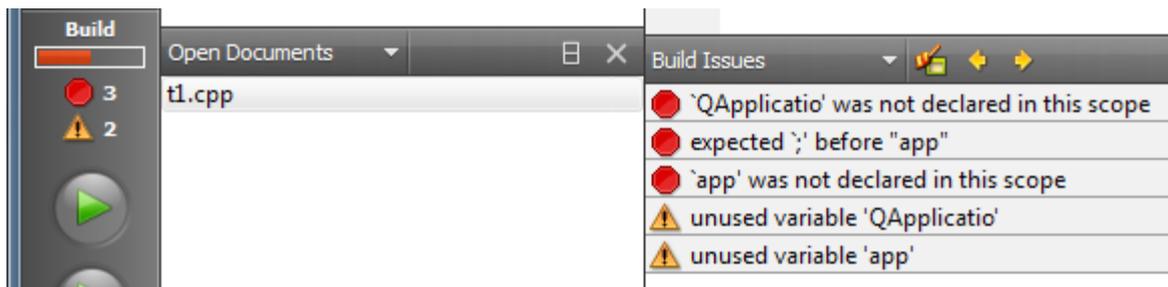
In der IDE gibt es unten links drei runde Anknickbuttons. Die Bedeutungen dieser Buttons steht hier links davor.

Ein kurzer Klick auf Erstellen (build all) bringt Compiler und Linker zum Übersetzen und Linken des Projektes. Das gelingt auch, wenn das Projekt – wie sonst üblich – aus mehreren Dateien und Ressourcen besteht. (Bei einem langen Klick erscheint eine kleine Auswahlbox, in der Sie zwischen einer Debug- und einer Release-Version umstellen können.)

In der Fortschrittsanzeige zeigt ein Balken den Fortschritt des Erstellens an. Bei Erfolg wird der Balken am Ende grün.

Nach dem erfolgreichen Erstellen können Sie die Anwendung starten oder debuggen.

Falls das Erstellen auf Grund von Fehlern nicht gelingt, werden Ihnen diese Fehler angezeigt, wie dies der folgende Ausschnitt anzeigt. (Hier wurde als Fehler bei QApplication das n am Ende weggelassen.) Der Fortschrittsbalken ist rot statt grün und rechts erscheinen einige Fehlermeldungen. Diese verschwinden beim nächsten erstellen, wenn der kleine Fehler beseitigt ist.



Wenn alles gut gegangen ist und die Anwendung gestartet wird, erscheint das kleine rechts abgebildete GUI-Fenster. Sie können seine Größe verändern, es auf die Taskleiste reduzieren und es mit dem Schließfeld schließen. Nur das Anklicken des Buttons bewirkt, außer gedrückt zu erscheinen, noch gar nichts.



Das wird sich im zweiten Projekt des Tutorials ändern.

## Fortsetzen

So, jetzt sollten Sie nach und nach die nächsten Projekte (t2, t3, ...) des Tutorials anlegen, die Quelltexte eingeben und studieren, sowie die Anwendungen zum Laufen bringen.

## Wichtiger Hinweis

Die erstellten Anwendungen können in MS Windows nicht ohne weiteres außerhalb des Creators zum Laufen gebracht werden. Hierfür werden mehrere dll-Dateien (Dynamic Link Library) benötigt, welche dem Windows-System nicht zur Verfügung stehen.

## Zweites Projekt hinzufügen

Im Qt Creator werden Projekte in Sitzungen (Sessions) organisiert. In einer Session können Sie neue Projekte anlegen, bereits vorhandene hinzufügen und solche entfernen. Anders als beispielsweise in MS Visual Studio ist ein Projekt nicht einer Session fest und eindeutig zugeordnet. Ein Projekt kann hier zu mehreren Sessions gehören. Die Projekte liegen unabhängig von Sessions irgendwo im Dateibaum.

Wenn Sie bisher noch keine Session angelegt haben, wird die so genannte „default“ Session verwendet. In dieser sollen Sie nun das zweite Projekt hinzufügen. Dies gelingt wie bisher mit **File/New...**

Danach wählen Sie wieder „**Empty Qt4 Project**“ aus! Nach OK erhalten Sie den Dialog für den Projektnamen und dessen Speicherort (location).

Geben Sie hinter „Name:“ „**t2**“ ein!

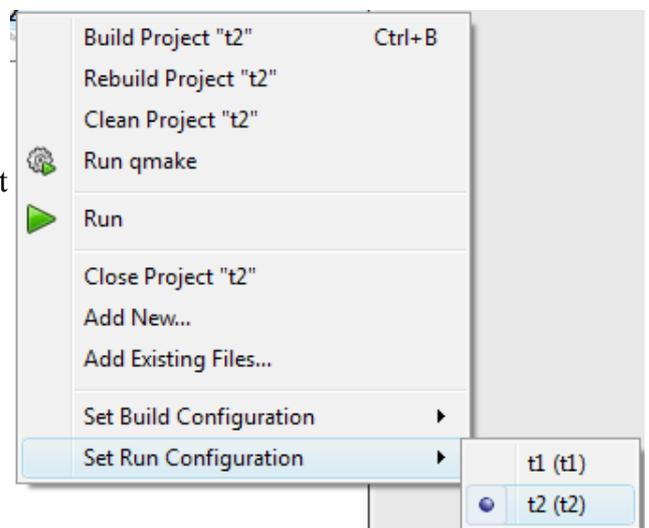
Hinter „Create in:“ steht leider der Pfad des zuletzt angelegten bzw. aktiven Projekts. Ändern Sie diesen ab, indem Sie hier „\t1“ am Ende entfernen! Dann wird das neue Projekt in einem Unterverzeichnis „t2“ angelegt. Das ist zwar etwas umständlich, der Qt Creator verhält sich aber so.

Nach dem Klick auf „Next“ überprüfen Sie bitte im folgenden Dialog den Pfad unter „Files to be added:“! Hier sollte am Ende „t2\t2.pro“ stehen. Wenn der Pfad stimmt, klicken Sie auf „Finish“! Danach erscheint das neue, noch leere Projekt unter „Projects“ oben links, wie es die Abbildung rechts zeigt.



Nun sollten Sie dieses Projekt „t2“ zum aktiven Projekt machen. Das heißt im Creator zwar nicht so, bewirkt aber, dass ein Run die später erzeugte Anwendung starten wird. Außerdem, und das ist an dieser Stelle hilfreich, wird bei der neu anzulegenden C++-Datei dieses Projekt bereits vorausgewählt sein. Dies schützt Sie ein wenig vor einem Zuordnungsfehler.

Um dies zu erreichen, klicken Sie mit der rechten Maustaste auf eines der Projektsymbole (t1 oder t2). Es öffnet sich ein kontextsensitives Menü. Darin schieben Sie den Mauszeiger auf „Set Run Configuration“ und dort auf den kleinen Pfeil rechts! Nun öffnet sich ein Untermenü, welches alle in der Session eingetragenen Projekte anzeigt (hier nur t1 und t2). Darin klicken Sie auf t2.



Vor dem aktiven Projekt erscheint der Punkt. In der Projektverwaltung ist der Name des aktiven Projekts immer fett dargestellt. Anschließend werden also „t1“ normal und „**t2**“ fett dargestellt sein.

## Dem zweiten Projekt eine Quelltextdatei hinzufügen

Leiten Sie dies wieder mit File/New... ein! Im folgenden Dialog wählen Sie wie gehabt „C++ Source File“ aus! Im nächsten Dialog ist der Pfad des Ordners zum aktiven Projekts bereits eingetragen.

Notieren Sie hinter „Name:“ den Namen der C++-Datei, hier „t2“!

### Hinweis:

Es ist zwar üblich für solche C++-Dateien, welche die main-Funktion beinhalten den Dateinamen main.cpp zu verwenden. Die Namen t1, t2, ... erleichtern aber den Dateitransfer, weil sich im Namen der Projektname wieder spiegelt. In umfangreicheren Projekten sollte dies anders als hier gehandhabt werden.

### Choose the location