

## Linken zusätzlicher Libraries in ECLIPSE

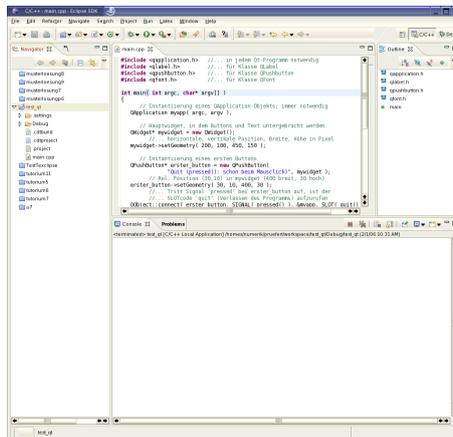
Wir behandeln das Problem am Beispiel Linken der qt-Library. Wir gehen davon aus, das ein managed make Projekt test Qt erstellt ist.

Um Objekte aus der qt-Bibliotheken zu benutzen, müssen diese wie üblich mit

```
#include <qapplication.h>
#include <qlabel.h>
#include <qpushbutton.h>
#include <qfont.h>
```

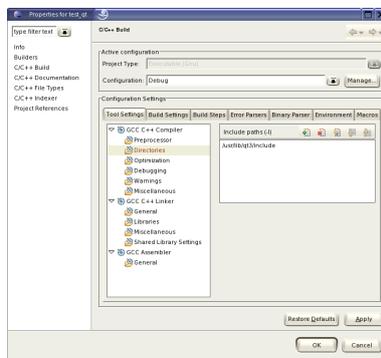
eingebunden werden. Wir müssen ECLIPSE nun mitteilen, wo es die Bibliotheken findet. Diese liegen, je nach Betriebssystem und (bei LINUX) auch Betriebssystem-Variante in den Verzeichnissen c:\qt\include c:\qt\lib (Windows) oder /usr/lib/qt3/include, /usr/lib/qt3/lib, /usr/lib/qt/include, /usr/lib/qt/lib oder /usr/local/qt3/include, /usr/local/qt3/lib. Ein Kommandoaufruf des C++ Compilers unter Linux sähe dann so aus: g++ -I/usr/lib/qt3/include -L/usr/lib/qt3/lib -lqt -o main main.cpp

Einstellen der Pfade: Dazu in ECLIPSE im Fenster C/C++ Projects oder im Navigator-Fenster das Projekt markieren.

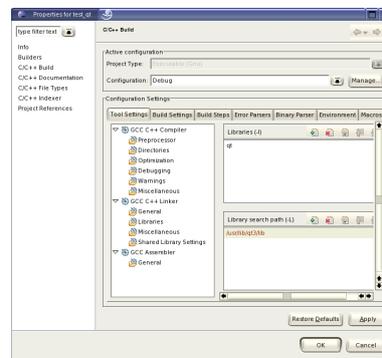


Mit der rechten Maustaste auf das markierte Projekt klicken und den Dialog Properties suchen. Dort im GCC C++ Compiler Dialog bei Directories den Pfad zu den qt Bibliotheken eintragen (grünes Plus öffnet einen Dialog zum Hinzufügen). Apply klicken.

Unter GCC C++ Linker Libraries auswählen und unter Libraries qt sowie unter Library search path den Pfad zum Verzeichnis lib, in der Regel etwa /usr/lib/qt3/lib eintragen. Apply klicken.



GCC C++ Compiler Dialog



GCC C++ Linker Dialog

Falls es schon eine Quelltextdatei mit #include<q\*\*\*> gibt, wird nach dem Apply der Compiler arbeiten und die Fehlermeldungen verschwinden. Mit anderen Libraries sollte eine analoge Vorgehensweise ebenfalls zum Erfolg führen.