



# KDEs Weg zu Qt 6

Strategien und Lessons Learned

23.06.2021, betterCode Qt 2021

Nicolas Fella

[nicolas.fella@kde.org](mailto:nicolas.fella@kde.org)  
[@nicofee](https://twitter.com/nicofee)





## Was ist KDE?

- Erstellt Freie Software mit Qt seit 25 Jahren
- Produkte: Plasma Desktopumgebung, Krita, Kdenlive, ...
- Portiert seit Qt 1
- KDE Free Qt Foundation
- Regelmäßige Beiträge zu Qt



## Über mich

- Studiere Master Informatik in Würzburg
- Entwickle Software mit Qt seit 2017
- (Co-) maintainer mehrerer KDE Subprojekte (Plasma, Plasma Mobile, KDE Connect, ...)
- Software Engineer bei KDAB
- Qt Contributor



## Ausgangslage

- KDE hat viel Code: ~15 Mio. LoC
- Fast ausschließlich C++ und Qt5
- Viele unabhängige und diverse Subprojekte
- Verschiedene Codezeitalter
- QtWidgets und QML
- Teils grafikintensive Anwendungen
- CMake als build system



## KDE Frameworks

- 80+ modulare Bibliotheken
- Interessant für externe Benutzer
- API + ABI Stabilität
- Qt 6 Portierung ist auch Zeitpunkt für Redesign



## KDE Frameworks 6 Design Ziele

- Einfache Portierung
- Bessere APIs
- Separierung Widgets/QML
- Weniger Abhängigkeiten
- Besserer cross-Platform support
- Separierung API und Implementierung



## Abschätzungen

- Deprecation Warnungen sind guter Indikator
- Fehlende Module in Qt6
- Neuer Code ist tendenziell einfacher zu Portieren
- High-level code einfacher als low-level
- Ausprobieren!



## Build System

- Anpassungen nötig
- Einfach: s/5/6/
- Kombiniertes support möglich
- Versionslose Targets/Befehle: Qt::Core, qt\_add\_resource
- Vorsicht mit versionslosen Targets in Bibliotheken





## Erste Eindrücke

- Viele trivial behebbare Probleme
- Fehlende includes
- Qt 6 ist strikter in machen Dingen
- Veränderte Typen/Signaturen (override is your friend)
- Verhaltensänderungen schwer abzuschätzen



## QTextCodec

- Neu in Qt6: QStringConverter
- Available in Qt5Compat
- Limitierter Codec support, nur UTF und Latin1
- Breiter Codec support nötig
- QTextStream standardmäßig UTF-8 in Qt6



## QStringRef

- Ersetzt durch QStringView
- QStringView in Qt5, aber mit fehlender API
- (Noch) keine große Portierung in KDE
- QStringRef in Qt5Compat, aber nur begrenzt nützlich ohne `QString::leftRef/midRef/rightRef/splitRef`



## QRegExp

- Ersetzt durch QRegularExpression in Qt5
- Verfügbar in Qt5Compat
- Kleinere Verhaltensänderungen
- Portierung in Qt5 möglich, aber nicht blind!



- Einführung des Rendering Hardware Interface (RHI)
- Low-ish level code benötigt Portierung
- Teils einfach, teils schwierig
- Krita erwartet Probleme wegen fehlendem Angle, keine klare Lösung



- Große Änderungen geplant, Details noch unklar
- Verspricht verbesserte Performance und Tooling
- Portierung wird herausfordernd
  - Keine context properties
  - Kein unqualified lookup
- Überlegung: Abwarten um Bedeutung besser abzuschätzen



## Goodies

- Weiterentwickeltes Qt
- All the small things
- QAction in QtGui
- Neues QtMultimedia
- QtQuick3D
- Fokus auf verbessertes QML



## Zusammenfassung

- Alles wird gut
- Aber nicht ohne Herausforderungen
- Portierung langsamer da interne Änderungen
- Widgets sind überschaubar im Aufwand
- QML wird komplizierter, aber lohnenswert
- Modularität hilft





Fragen?