

Code once, deploy everywhere.





How can I support different  
mobile platforms efficiently?

# Sebastian Kügler

- Core Developer at KDE
- Alpha-Nerd at Blue Systems
- 10 years Qt experience
- 5 years mobile

# Qt Key Features

- Platform-independent
- Modular, Fast, Mature
- Very extensive
- LGPL and commercial license options
- Makes C++ pleasant



# What exactly?

- Native binary code
- UI done in QtQuick
- OpenGL-ES-accelerated rendering
- Quick development / testing / deployment cycles
- System/device interaction through abstract Qt APIs, or native extensions (C++, Objective-C, JNI, ...)



# Qt on iOS



- Uses Xcode for development and deployment

Supported devices:

- iPhone 3GS and later
- iPod Touch third and later generations
- iPad 2 and later
- iPad Mini

(I haven't tried.)

# Qt on Android



- Test deployment to device via IDE
- Static build (~30MB+ size on device) or
- Shared libraries
- QQuickControls abstract Android UI
- Sensors support much better than on Desktop Linux

Mostly, it *just works*.

# Where it shines...

- Games
- Immersive applications
- Computation heavy applications
- Media





These slides →



- I blog at <http://vizZion.org>
- <https://plus.google.com/+SebastianKügler>