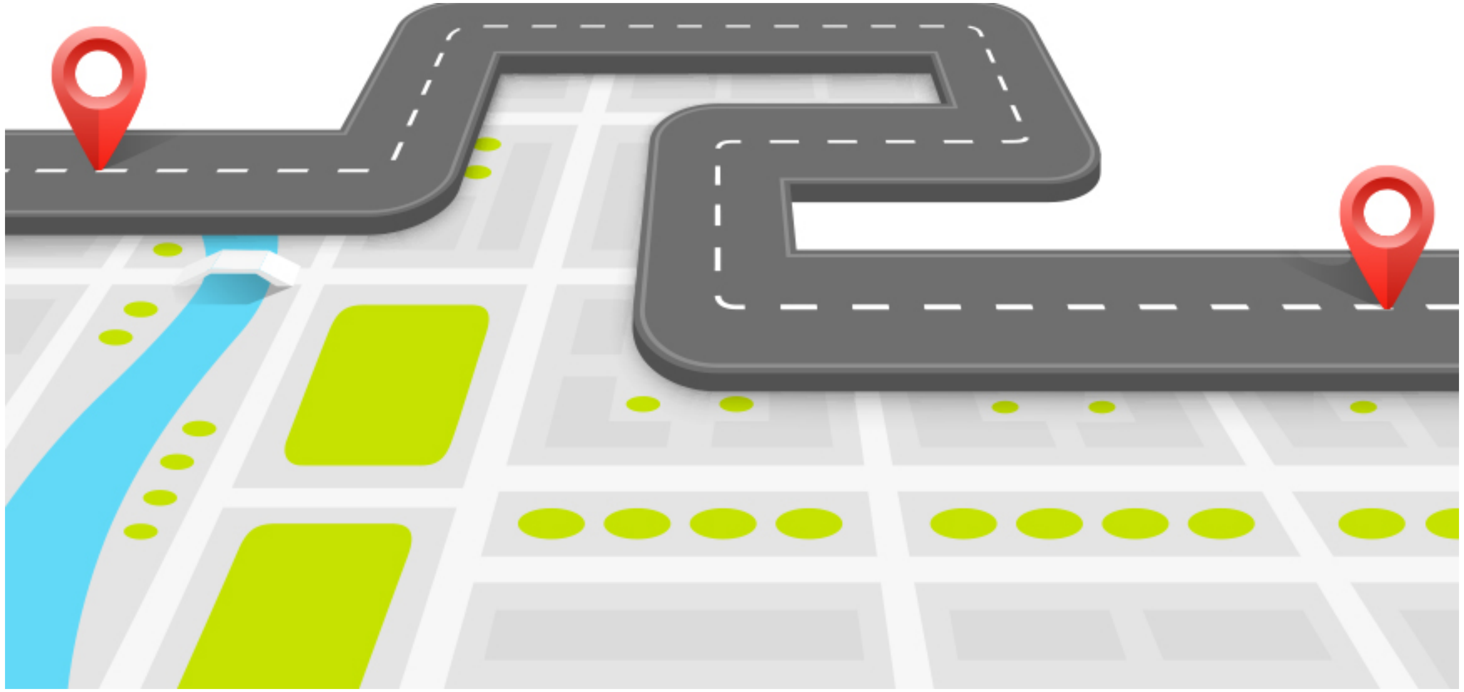
Current roadmap of Qt

2017-2019

up to **Qt 6.0**



April, 2017

Roadmap

- Products
- Graphics and GUI
  - Qt Quick
  - 3D
  - WebGL
- Embedded development
- Other stuff

But first a bit of history

Brief (and simplified) history of The Qt Company, to address the questions about its past and present time.

- 1994: **Troll Tech**. First public release in 1995
- 2008: **Nokia** acquires Troll Tech
- 2012: **Digia** buys Qt from Nokia
- 01.05.2016: **The Qt Company** demerges from Digia

Now The Qt Company is an independent stock listed company.

Products

All the products share the same common core libraries, but there are differences in additional tools and features.

- **Current products**
  - Application Development - regular desktop and mobile applications
  - Device Creation - development for embedded platforms
  - Automotive Suite - digital instrument cluster (IC) and in-vehicle infotainment (IVI)
- **Expected products**
  - Industrial Automation - Industry 4.0, smart factory, machine-to-machine communication
  - Set-Top Box - media-center, smart TVs
  - ...

Graphics and GUI

Cache for OpenGL shaders

Reduce the time of initialization . After the first launch of an application the consequent launches will happen faster (even for drivers with their own implementation of shader cache).

Shared usage of bitmaps

Optimize the RAM consumption in multi-process systems.

Support for other graphical backends

Not only OpenGL - Vulkan, Direct3D, etc.

Qt Quick

Path rendering

Declarative creation of complex shapes out of lines and curves using hardware accelerated QML, which gives a significant performance boost.

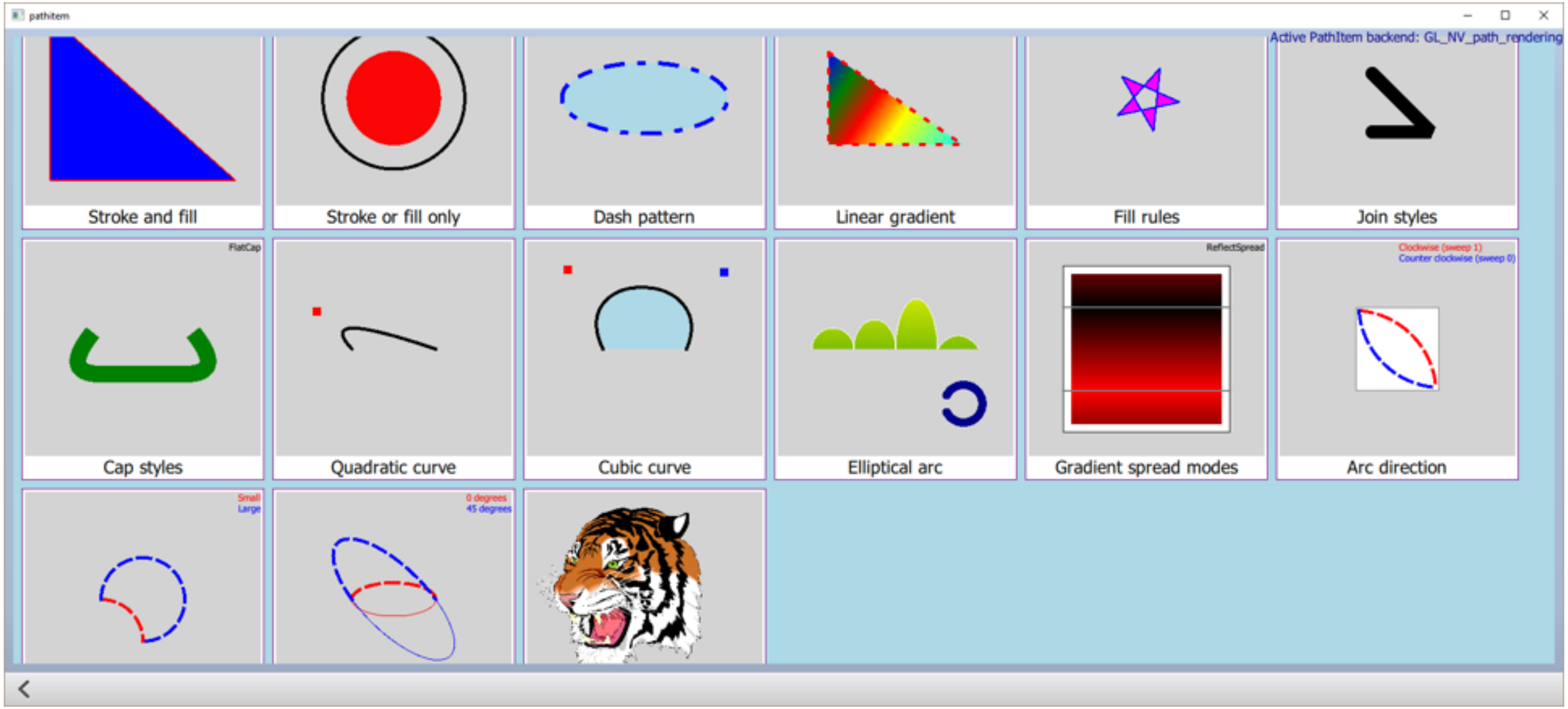
Qt Quick Controls

New controls (TableView, DelayButton, ScrollView, editable ComboBox) and multi-touch support.

Other improvements

Integrated Qt Quick Compiler functionality, reworked garbage collector for QML engine.

Qt Quick path rendering



3D

Currently in development

- using Qt Quick elements as 2D textures in a 3D scene
- mesh morphing
- improving particles and animations (with import of COLLADA animations)
- improved support for external editors and formats (models and materials import)
- further improvements in saving/loading 3D scene and introspection API
- physically based rendering

Qt 3D Studio

Former NVIDIA DRIVE Design Studio. Integration with existing 3D framework, QML API.

Experimental native UI controls

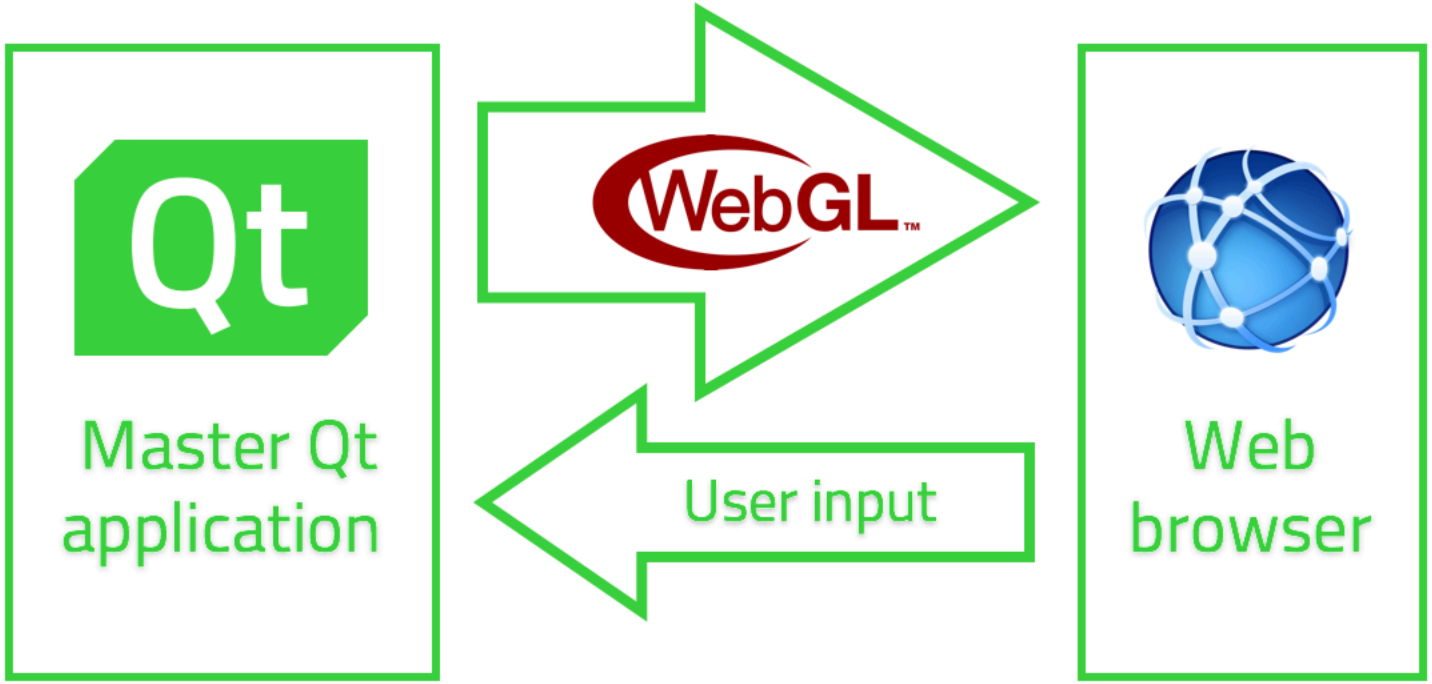
```qml
import QtQml 2.0
import Qt.CommonControls 1.0 as Common
import Qt.AppKitControls 1.0 as AppKit

Window {
    id: window
    visible: true

    Common.Button {
        id: button
        x: 20
        y: 20
        text: "Click me"
        onClicked: text ="You clicked me!"
    }

    Common.TextField {
        id: textField
        placeholderText: "TextField"
        x: 20
        y: button.bottom + 8
    }

    AppKit.SearchField {
        placeholderText: "Search"
        x: 20
        y: textField.bottom + 8
    }
}
```

```cpp
int main(int argc, char *argv[])
{
    QGuiApplication app(argc, argv);
    QCommonWindow window;

    QCommonButton commonButton(QStringLiteral("Click me")
    commonButton.move(20, 20);
    QObject::connect(&commonButton, &QCommonButton::click
        commonButton.setText(QStringLiteral("Clicked!"));
    });

    QCommonTextField textField(&window);
    textField.setPlaceholderText(QStringLiteral("TextFiel
    textField.move(20, commonButton.bottom() + 8);

#if defined(Q_OS_MACOS)
    QAppKitSearchField searchField;
    searchField.setParent(&window);
    searchField.move(20, textField.bottom() + 8);
    NSSearchField *nsSearchField = searchField.nsSearchFi
    nsSearchField.placeholderString = @"Search...";
#endif

    window.showFullScreen();
    return app.exec();
}
```

WebGL streaming

“Remote display” and control for the Qt application by a common web-browser (or another Qt application):

- connection is over the network
- OpenGL calls “streaming”
- user input in return channel
- “out-of-the-box” on browsers

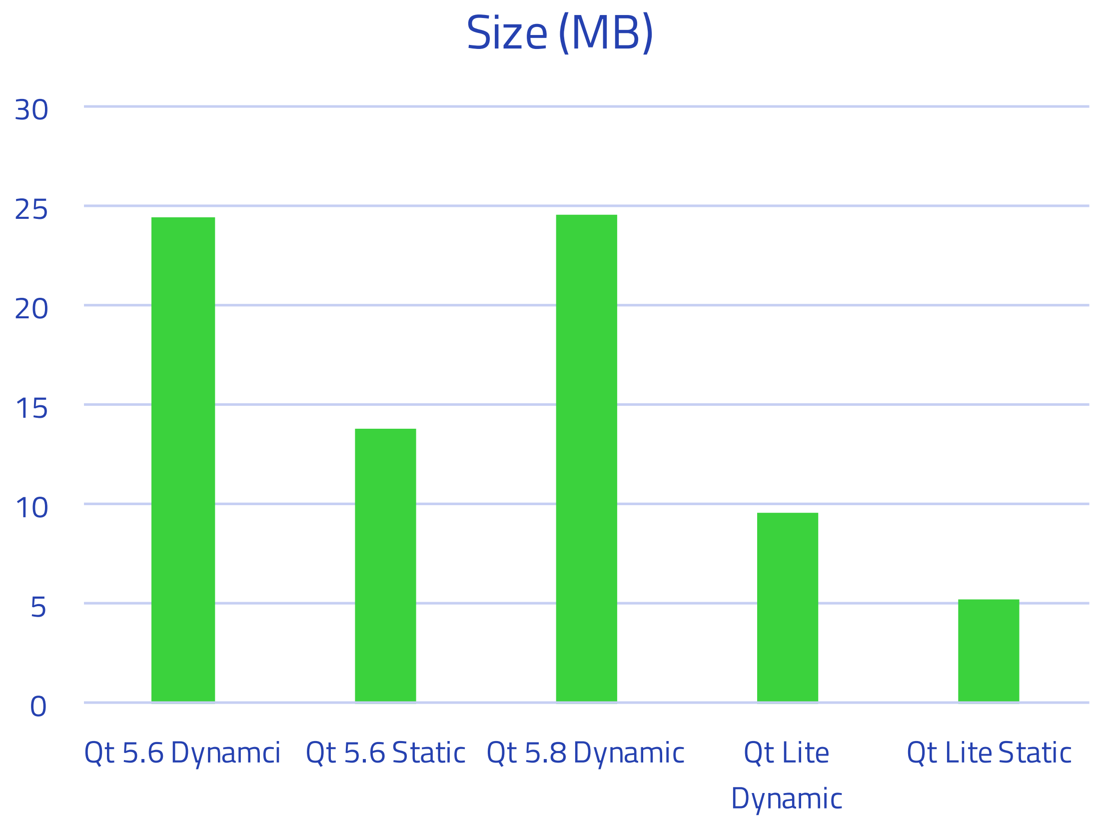


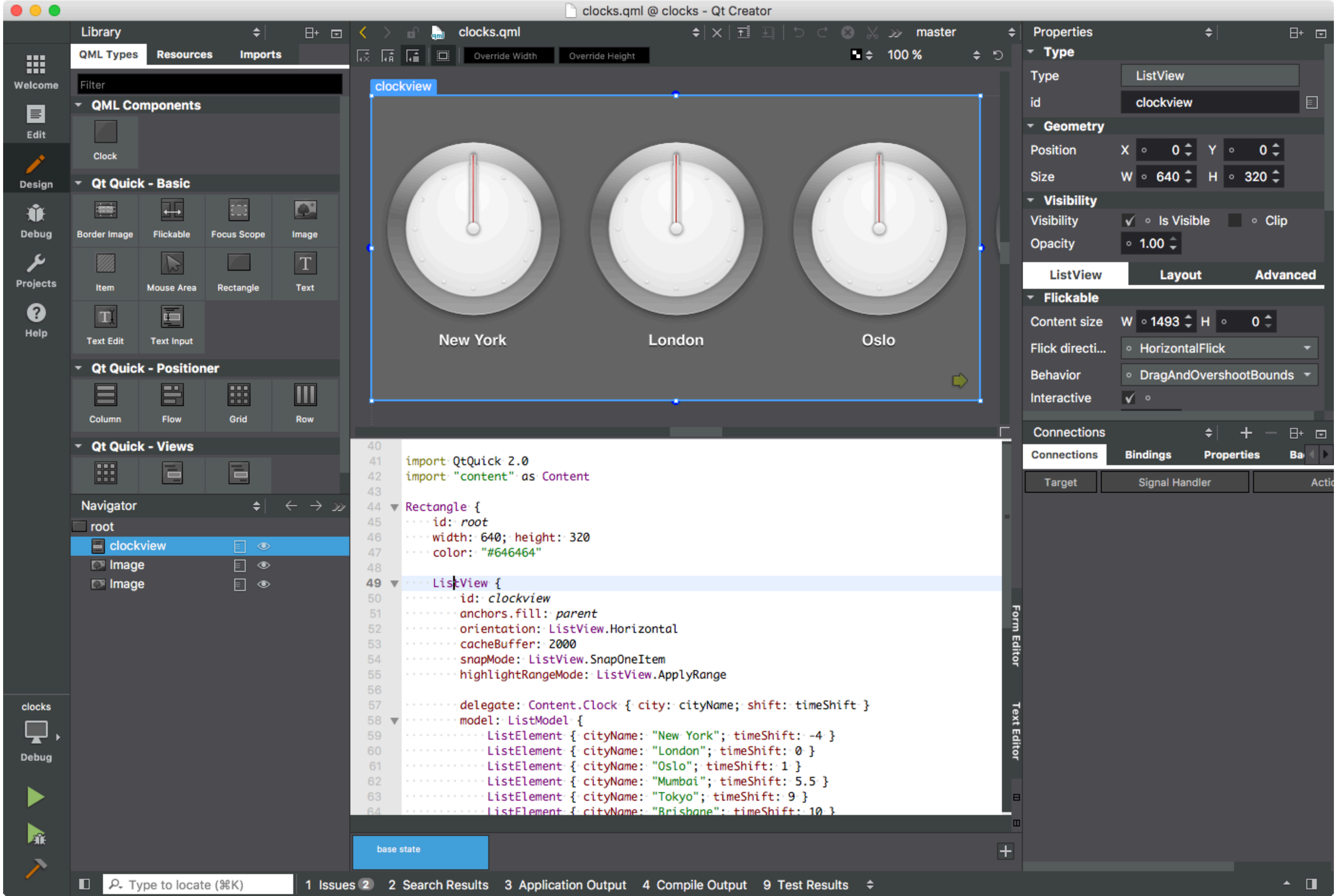
Embedded development

More referenced devices to support, meaning more pre-built Boot to Qt images and toolchains available out-of-the-box.

Further development and optimization for so-called “Qt Lite” - reconfigurability, modularization, reducing binaries size and RAM usage.

QDB - Qt Debug Bridge - a replacement for ADB.

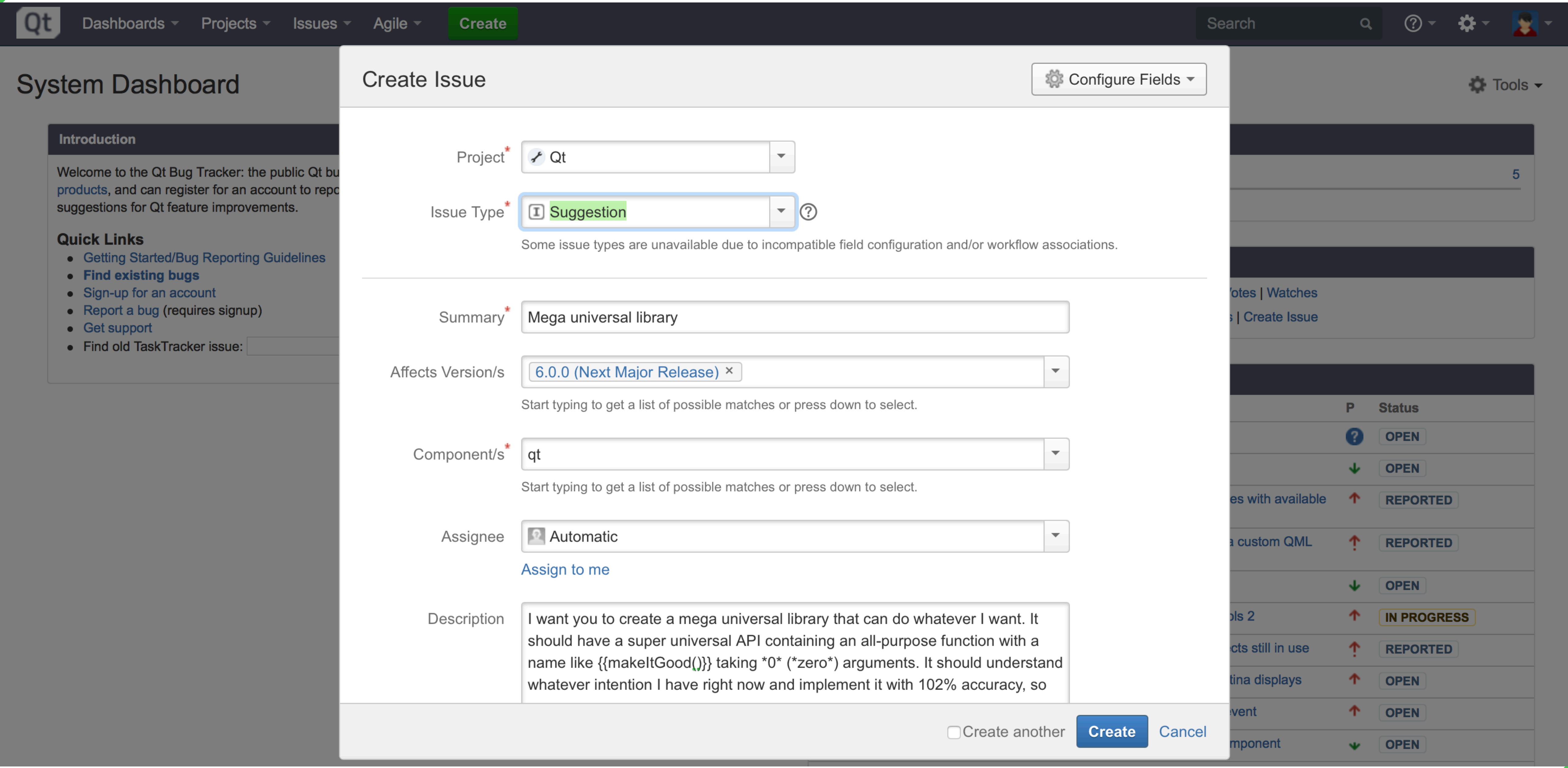



clocks.qml @ clocks - Qt Creator
Library
QML Types
Resources
Imports
Filter
QML Components
Clock
Qt Quick - Basic
Border Image
Flickable
Focus Scope
Image
Item
Mouse Area
Rectangle
Text
Text Edit
Text Input
Qt Quick - Positioner
Column
Flow
Grid
Row
Qt Quick - Views
Navigator
root
clockview
Image
Image
Welcome
Edit
Design
Debug
Projects
Help
clocks
Debug
clocks.qml
master
Override Width
Override Height
100 %
clockview
New York
London
Oslo
import QtQuick 2.0
import "content" as Content
Rectangle {
id: root
width: 640; height: 320
color: "#646464"
ListView {
id: clockview
anchors.fill: parent
orientation: ListView.Horizontal
cacheBuffer: 2000
snapMode: ListView.SnapOneItem
highlightRangeMode: ListView.ApplyRange
delegate: Content.Clock { city: cityName; shift: timeShift }
model: ListModel {
ListElement { cityName: "New York"; timeShift: -4 }
ListElement { cityName: "London"; timeShift: 0 }
ListElement { cityName: "Oslo"; timeShift: 1 }
ListElement { cityName: "Mumbai"; timeShift: 5.5 }
ListElement { cityName: "Tokyo"; timeShift: 9 }
ListElement { cityName: "Brisbane"; timeShift: 10 }
Form Editor
Text Editor
base state
Properties
Type
Type
ListView
id
clockview
Geometry
Position
X
0
Y
0
Size
W
640
H
320
Visibility
Visibility
Is Visible
Clip
Opacity
1.00
ListView
Layout
Advanced
Flickable
Content size
W
1493
H
0
Flick directi...
HorizontalFlick
Behavior
DragAndOvershootBounds
Interactive
Connections
Connections
Bindings
Properties
Target
Signal Handler
Type to locate (⌘K)
1 Issues 2
2 Search Results
3 Application Output
4 Compile Output
9 Test Results

Other stuff

- Combined design-mode and code-mode for Qt Quick Designer
- Further development and improvement of Qt Wayland Compositor (and Application Manager)
- Updated support for RTOS INTEGRITY
- More Apple platforms supported: tvOS and watchOS
- Qt Marketplace (something like NuGet for VS)
- Replacing MinGW with Clang on Windows (*not decided and no concrete timeline whatsoever*)
- Debugging for QJSEngine (*no concrete timeline whatsoever*)
- **Fix all the bugs** (as usual)

And by the way, you can actually influence the roadmap of Qt.



Qt
Dashboards
Projects
Issues
Agile
Create
Search
System Dashboard
Tools
Introduction
Welcome to the Qt Bug Tracker: the public Qt bu
products, and can register for an account to repo
suggestions for Qt feature improvements.
Quick Links
Getting Started/Bug Reporting Guidelines
Find existing bugs
Sign-up for an account
Report a bug (requires signup)
Get support
Find old TaskTracker issue:
Create Issue
Configure Fields
Project*
Qt
Issue Type*
Suggestion
Some issue types are unavailable due to incompatible field configuration and/or workflow associations.
Summary*
Mega universal library
Affects Version/s
6.0.0 (Next Major Release) ×
Start typing to get a list of possible matches or press down to select.
Component/s*
qt
Start typing to get a list of possible matches or press down to select.
Assignee
Automatic
Assign to me
Description
I want you to create a mega universal library that can do whatever I want. It should have a super universal API containing an all-purpose function with a name like {{makeItGood()}} taking *0* (*zero*) arguments. It should understand whatever intention I have right now and implement it with 102% accuracy, so
Create another
Create
Cancel
otes | Watches
| Create Issue
P
Status
OPEN
OPEN
es with available
REPORTED
a custom QML
REPORTED
OPEN
ols 2
IN PROGRESS
cts still in use
REPORTED
tina displays
OPEN
vent
OPEN
mponent
OPEN

Links

- Roadmap: http://blog.qt.io/blog/2017/02/22/qt-roadmap-for-2017/
- New features in Qt 5.9: https://wiki.qt.io/New_Features_in_Qt_5.9
- OpenGL shader cache: http://blog.qt.io/blog/2017/03/23/boosting-performance-shader-binary-caching-qt-5-9/
- Vulkan support: https://blog.qt.io/blog/2016/02/16/the-qt-company-joins-khronos-group-and-promotes-vulkan/
- OpenVG support: http://blog.qt.io/blog/2017/03/31/qt-quick-openvg/
- WebGL streaming: http://blog.qt.io/blog/2017/02/22/qt-quick-webgl-streaming/
- Qt 3D Studio announce: http://blog.qt.io/blog/2017/02/20/introducing-qt-3d-studio/
- Native common controls: https://blog.qt.io/blog/2017/02/06/native-look-feel/
- Qt Debug Bridge development: https://bugreports.qt.io/browse/QTBUG-52857
- Qt Creator 4.3 beta-version: http://blog.qt.io/blog/2017/03/30/qt-creator-4-3-beta-released/
- Apple platforms support: http://blog.qt.io/blog/2017/01/30/whats-new-in-qt-on-apple-platforms/
- Bugtracker: https://bugreports.qt.io/