Qt 5.9 FRAMEWORK

MAKING VISIONS WORK.

Qt

Qt 5.9 FRAMEWORK

Qt Essentials
Qt Essentials define the foundation of Qt on all platforms. They are available on all supported development platforms and on the tested target platforms. They will remain source- and binary-compatible during Qt 5. Essential modules are general and useful for a majority of Qt applications. A module that is used for a special purpose is considered an add-on module even if it is available on all supported platforms.

Qt Core   214
Core non-graphical classes used by other modules.

Submodules:
Animation Classes
Threading Classes
Container Classes
Plugin Classes
Implicitly Shared Classes
State Machine Classes
Input/Output Classes
Event Classes

C++:   #include <QtCore>

Qt GUI   216
Base classes for graphical user interface (GUI) components. Includes OpenGL.

Submodules:
Event Classes
Painting Classes
Rendering in 3D

qmake: QT += gui
C++:   #include <QtGui>

Qt Widgets   204
The Qt Widgets module provides a set of user interface (UI) elements to create classic desktop-style user interfaces.

Submodules:
Basic Widget Classes
Advanced Widget Classes
Abstract Widget Classes
Organizer Widget Classes
Graphics View Classes
Model/View Classes
Main Window and Related Classes
Widget Appearance and Style-Related Classes
Layout Classes

qmake: QT += widgets
C++:   #include <QtWidgets>

Qt Test   3
Qt Test provides classes for unit-testing Qt applications and libraries.

qmake: QT += testlib
C++:   #include <QtTest>

Qt QML   23/16
Classes for QML and JavaScript languages.

qmake: QT += qml
C++:   #include <QtQml>
QML: import QtQml 2.2

Qt SQL   16
Qt SQL is an essential module which provides support for SQL databases.

qmake: QT += sql
C++:   #include <QtSql>

Qt Multimedia   89/27
Classes for audio, video, radio and camera functionality.

qmake: QT += multimedia
C++:   #include <QtMultimedia>
QML: import QtMultimedia 5.8

Qt Quick   45/168
A declarative framework for building highly dynamic applications with custom user interfaces.

qmake: QT += quick
C++:   #include <QtQuick>
QML: import QtQuick 2.5

Qt Quick Controls   34
Reusable Qt Quick-based UI controls to create classic desktop-style user interfaces.

qmake: QT += qml
QML: import QtQuick.Controls 1.4

Qt Network   47
Classes to make network programming easier and more portable.

qmake: QT += network
C++:   #include <QtNetwork>

Qt Multimedia Widgets   4
Widget-based classes for implementing multimedia functionality.

qmake: QT += multimediawidgets
C++:   #include <QtMultimediaWidgets>

Qt Quick Dialogs   5
Types for creating and interacting with system dialogs from a Qt Quick application.

qmake: QT += quick
QML: import QtQuick.Dialogs 1.2

Qt Quick Layouts   5
Layouts are items that are used to arrange Qt Quick based items in the user interface.

qmake: QT += quick
QML: import QtQuick.Layouts 1.3

Deprecated
Modules marked as deprecated.

Enginio   6/4
A Backend-as-a-Service solution to ease the backend development for connected and data-driven applications.

qmake: QT += enginio
C++:   #include <Enginio/Enginio>
QML: import Enginio 1.0

Qt OpenGL   11
OpenGL support classes. Deprecated in favor of the QOpenGL* classes in the Qt GUI module.

qmake: QT += opengl
C++:   #include <QtOpenGL>

Qt Script   13
Classes for making Qt applications scriptable. Deprecated in favor of the QJS* classes in the Qt QML module.

qmake: QT += script
C++:   #include <QtScript>

Qt Script Tools   1
Additional components for applications that use Qt Script.

qmake: QT += scripttools
C++:   #include <QtScriptTools>

Technology Preview Modules
The following modules are still in development but are available as technology previews.

Qt Network Authorization   4
Provides support for OAuth-based authorizationto online services.

qmake: QT += networkauth
C++:   #include <QtNetworkAuth>

Qt Speech   3
Provides support for accessibility features such as text-to-speech.

qmake: QT += texttospeech
C++:   #include <QTextToSpeech>

Qt Remote Objects   10
Provides support for accessibility features such as text-to-speech.

qmake: QT += remoteobjects
C++:   #include <QtRemoteObjects>

Value-Add Modules
In addition to the modules released as part of Qt 5, the following modules and tooling build on top of the Qt libraries to provide additional value. They have their own release schedule and are available under different Qt licences.

Qt for Device Creation
Tools for fast, easy and fully integrated embedded device application development. Includes most other Value-Add features.

Qt Quick Compiler   6
Enables compiling a .qml source files into application binaries, improving load times and security for code assets.

Qt Tools
Qt Tools run on all the supported development platforms and facilitate the development and design of applications.

Qt Designer   21
Classes for extending Qt Designer.

qmake: CONFIG += designer
C++:   #include <QtDesigner>

Qt Help   11
Classes for integrating online documen-tation into applications, similar to Qt Assistant.

qmake: QT += help
C++:   #include <QtHelp>

Qt UI Tools   1
Classes to handle the forms created in Qt Designer.

qmake: QT += uitools
C++:   #include <QtUiTools>

Qt Add-Ons
Qt Add-On modules bring additional value for specific purposes. These modules may only be available on some development platforms. Many add-on modules are either feature-complete and exist for backwards compatibility, or are only applicable to certain platforms. Each add-on module specifies its compatibility promise separately.

Qt Concurrent   5
Classes for writing multi-threaded programs without using low-level threading primitives.

qmake: QT += concurrent
C++:   #include <QtConcurrent>

Qt D-Bus   20
Classes for inter-process communication over the D-Bus protocol.

qmake: QT += dbus
C++:   #include <QDBus>

Qt Print Support   8
Qt provides extensive cross-platform support for printing.

qmake: QT += printsupport
C++:   #include <QPrintSupport>

Qt XML   31
C++ implementations of SAX and DOM.

qmake: QT += xml
C++: #include <QtXml>

Qt XML Patterns   16
Support for XPath, XQuery, XSLT and XML schema validation.

qmake: QT += xmlpatterns
C++:   #include <QtXmlPatterns>

Qt Purchasing   3/3
Enables in-app purchase of products in Qt applications.

qmake: QT += purchasing
C++:   #include <QtPurchasing>
QML: import QtPurchasing 1.0

Qt Canvas 3D   17
Enables OpenGL-like 3D drawing calls from Qt Quick applications using JavaScript.

QML: import QtCanvas3D 1.1

Qt 3D
Qt 3D provides functionality for near-real-time simulation systems with support for 2D and 3D rendering in both Qt C++ and Qt Quick applications.

qmake: QT += 3dcore 3drender 3dinput
        3dlogic 3dextras 3danimation
        3dquick

C++:   #include <Qt3DCore>
        #include <Qt3DRender>
        #include <Qt3DInput>
        #include <Qt3DLogic>
        #include <Qt3DExtras>
        #include <Qt3DAnimation>

QML: import Qt3D.Animation 2.9
        import Qt3D.Core 2.0
        import Qt3D.Extras 2.0
        import Qt3D.Input 2.0
        import Qt3D.Logic 2.0
        import Qt3D.Render 2.0
        import Qt3D.Scene2D 2.9

Qt SVG   4
Classes for displaying the contents of SVG files. Supports a subset of the SVG 1.2 Tiny standard.

qmake: QT += svg
C++:   #include <QtSvg>

Qt Image Formats   26/3
The Qt Image Formats add-on module provides optional support for other image file formats. The Qt Image Formats plugins are located as dynamic libraries in the runtime plugin directory (typically plugins/imageformats), together with the default image format plugins.

Qt Graphical Effects   25
Provides a set of QML types for adding visually impressive and configurable effects to user interfaces.

qmake: QT += quick
QML: import QtGraphicalEffects 1.0

Qt Data Visualization   35/35
UI Components for creating stunning 3D data visualizations.

qmake: QT += datavisualization
C++:   #include <QtDataVisualization>
QML: import QtDataVisualization 1.2

Qt Charts   48/42
UI Components for displaying visually pleasing charts, driven by static or dynamic data models.

qmake: QT += charts
C++:   #include <QtCharts> (namespace QtCharts)
QML: import QtCharts 2.2

Qt Serial Port   2
Provides access to hardware and virtual serial ports.

qmake: QT += serialport
C++:   #include <QtSerialPort/QtSerialPort>

Qt Serial Bus   20
Provides access to serial industrial businterface. Currently the module supports the CAN bus and Modbus protocols.

qmake: QT+= serialbus
C++:   #include <QtSerialBus>

Qt NFC   11/6
Provides access to near-field communication (NFC) hardware.

qmake: QT += nfc
C++:   #include <QtNfc>
QML: import QtNfc 5.2

Qt Bluetooth   26/4
Provides access to Bluetooth hardware.

qmake: QT += bluetooth
C++:   #include <QtBluetooth>
QML: import QtBluetooth 5.2

Qt Sensors   70/24
Provides access to sensor hardware and motion gesture recognition.

qmake: QT += sensors
C++:   #include <QSensor>
QML: import QtSensors 5.0

Qt Positioning   16/11
Provides access to position, satellite and area monitoring classes.

qmake: QT += positioning
C++:   #include <QGeoCoordinate>
QML: import QtPositioning 5.2

Qt Location   39/39
Displays map, navigation and place content in a QML application.

qmake: QT += location
C++:   #include <QGeoRoute>
QML: import QtLocation 5.6

Qt WebEngine   18/19
Provides a QML API to run web applications using the Chromium browser project.

Submodules: QtWebEngineCore
            QtWebEngine
            QtWebEngineWidgets

qmake: QT += webenginecore
            webengine webenginewidgets
C++:   #include <QtWebEngineCore>
            #include <QtWebEngineWidgets>
QML: import QtWebEngine 1.5

Qt WebChannel   2/1
Provides access to QObject or QML objects from HTML clients for seamless integration of Qt applications with HTML/JavaScript clients.

qmake: QT += webchannel
C++:   #include <QtWebChannel/QtWebChannel>
QML: import QtWebChannel 1.0

Qt WebSockets   4/2
Provides WebSocket communication compliant with RFC 6455.

qmake: QT += websockets
C++:   #include <QtWebSockets/QtWebSockets>
QML: import QtWebSockets 1.1

Qt WebView   1/2
Displays web content in a QML application by using APIs native to the platform, without the need to include a full web browser stack.

qmake: QT += webview
C++:   #include <QtWebView>
QML: import QtWebView 1.1

Qt Virtual Keyboard   36
A framework for implementing different input methods as well as a QML virtual keyboard. Supports localized keyboard layouts and custom visual themes.

QML: import QtQuick.VirtualKeyboard 2.3

Active Qt
Classes for applications which use ActiveX and COM.

QAxContainer   7
The QAxContainer module is a Windows-only extension for accessing ActiveX controls and COM objects.

qmake: QT += axcontainer
C++:   #include <QAxContainer>

QAxServer   3
The QAxServer module is a Windows-only static library that you can use to turn a standard Qt binary into a COM server.

qmake: QT += axserver
C++:   #include <QAxServer>

Qt Quick Controls 2   45
Provides lightweight QML types for creating performant user interfaces for embedded and mobile devices.

qmake: QT += quickcontrols2
C++:   #include <QtQuickControls2>
QML: import QtQuick.Controls 2.2

Qt Quick Extras   10
Provides a specialized set of controls that can be used to build interfaces in Qt Quick.

QML: import QtQuick.Extras 1.4

Qt Quick Widgets   1
Provides a C++ widget class for displaying a Qt Quick user interface.

qmake: QT += quickwidgets
C++:   #include <QQuickWidget>

Qt Platform Headers   7
Provides classes that encapsulate platform-specific information, tied to a given runtime configuration of a platform plugin.

Qt Android Extras   9
Provide platform-specific APIs for Android.

qmake: QT += androidextras
C++:   #include <QtAndroidExtras>

Qt Mac Extras   3
Qt Mac Extras provide classes and functions that enable you to use miscellaneous functionality specific to the OS X and iOS operating systems.

qmake: QT += macextras
C++:   #include <QtMacExtras>

Qt Windows Extras   8/9
Provide platform-specific APIs for Windows.

qmake: QT += winextras
C++:   #include <QtWinExtras>
QML: import QtWinExtras 1.0

Qt X11 Extras   1
Provide platform-specific APIs for X11.

qmake: QT += x11extras
C++:   #include <QX11Info>

Qt Wayland Compositor   16/9
Provides a framework to develop a Wayland compositor.

qmake: QT+= waylandcompositor
C++:   #include <QWaylandCompositor>
QML: import QtWayland.Compositor 1.0

Qt SCXML   10/1
Provides functionality to create state machines from SCXML files.

qmake: QT+= scxml
C++:   #include <QScxmlStateMachine>
QML: import Scxml 1.0

Qt Gamepad   3/4
Qt Gamepad is an add-on library that enables Qt app-lications to support the use of gamepad hardware.

qmake: QT+= gamepad
C++:   #include <QtGamepad>
QML: import QtGamepad 1.0

Development Tools

Utilities

makeqpf	Create pre-rendered fonts for embedded devices.
Meta-Object Compiler (moc)	Generate meta-object information for QObject subclasses.
User Interface Compiler (uic)	Generate C++ code from UI files.
Resource Compiler (rcc)	Embed resources into Qt applications during the build process.
(qdbusxml2cpp)	A tool to introspect D-Bus objects and messages. Available from the Qt SDK or the Qt 5 qttools repository.
D-Bus Viewer Qt D-Bus XML compiler	A tool to convert D-Bus interface descriptions to C++ source code.
qmlscene	A utility that loads and displays QML documents even before the application is complete.
windeployqt	Designed to automate the process of creating a deployable folder that contains all libraries, QML imports, plugins and translations that are required to run the application from that folder.

Development platforms
Linux, Windows and Mac OS X

Target platforms

Desktop
Linux/X11
macOS 10.10, 10.11, 10.12
Windows 7, 8.1, 10

Mobile
Android (API Level: 16)
iOS 8, 9, 10
Universal Windows Platform

Embedded
Embedded Linux
QNX 6.6.0, 7.0
INTEGRITY 11.4.x

Legend
The module supports all target platforms when there is no icon visible.

New module available in Qt 5.9

Android mobile operating system

Linux operating system

Windows Desktop / Windows Phone operating system

Commercial module from The Qt Company

Apple desktop operating system

iOS Apple mobile operating system

Title
Brief description   Number of classes or QML types

qmake: Configure option
C++:   Include for the module support
QML: Import for the module support

www.bbv.ch · info@bbv.ch