

Qt 模块简介

原创 Qt 学习 Qt 学习 2020-02-02

点击上方 蓝色 文字，快来 关注 我吧！

在《UI 文件设计与运行机制》一文中，介绍了 Qt 类库是以模块的形式组织各种功能的类，根据项目涉及的功能需求，在项目添加适当的类库模块支持。例如，在 `.pro` 文件中通常可见如下一行：

```
1 QT += core gui
```

其指定项目使用的 Qt 模块。使用 Qt 模块可将相应的头文件包含在内并使其链接到二进制文件。

通常情况下，QT 包含核心类库和 GUI 类库，以确保无需进一步配置即可构建标准 GUI 应用程序。如果要构建没有 Qt GUI 模块的项目，则需要使用 `--` 运算符排除 gui 值。以下行将使构建最少的 Qt 项目：

```
1 QT -= gui // Only the core module is used.
```

如果构建的项目是 Qt Designer 插件，可使用 `uipugin` 来将项目构建为库，使具有 Qt Designer 的特定插件支持。有关更多信息，请参阅构建和安装插件。

本篇目录

1. Qt 基础模块
2. Qt 扩展模块
3. Qt 增值模块
4. 技术预览功能
5. Qt 工具
6. 未来方向

运行环境：

win 10 + Qt 5.12.5 + Qt Creator 4.10

1. Qt 基础模块

基础模块定义了所有平台上 Qt 的基础功能，是 Qt 的核心。基础模块中的基石是 Qt Core，其他所有模块都依赖于此。基础模块在所有受支持的开发平台和经过测试的目标平台上均可使用。除了测试模块将保持源代码兼容之外，基础模块将在整个 Qt 5 中保持源代码和二进制的兼容。

基本模块是所有平台通用的，绝大多数 Qt 应用程序都可用。用于特殊目的的模块被视为附加模块，它建立在基础模块的基础之上，只能在支持这些模块的平台使用。

下面给出 Qt 基础模块及其简介：

■ Qt Core

Qt 5 的核心类库，其他各个模块都建立在 Core 模块之上。相对于 Qt 4，Qt 5 增加了 JSON 支持，并且将对 XML 的支持移

入到 Core 模块，不再是独立的 XML 模块。

- **Qt GUI**

图形用户界面 (GUI) 开发的最基础的类库，包括各种交互事件等；同时这个模块还包括了有关 OpenGL 的内容。

- **Qt Multimedia**

提供对视频、音频、无线电以及摄像头的支持。

- **Qt Multimedia Widgets**

提供基于组件的多媒体功能支持。

- **Qt Network**

提供简便的、可移植的网络编程类库。

- **Qt QML**

提供对 QML 和 JavaScript 语言的支持。

- **Qt Quick**

提供一个用于创建高度动画效果的应用程序的声明式框架。该框架建立在 QML 和 JavaScript 的基础之上。

- **Qt Quick Controls**

提供轻量级 QML 类型，为台式机、嵌入式和移动设备创建高性能的用户界面。这些 QML 类型采用简单且非常有效的桌面风格。

- **Qt Quick Dialogs**

提供一组适用于 Qt Quick 应用程序的系统对话框。

■ Qt Quick Layouts

提供一组适用于 Qt Quick 2 的组件布局。

■ Qt Quick Test

QML 应用程序的单元测试框架，测试用例使用 JavaScript 函数编写。

■ Qt SQL

提供对集成数据库 SQL 的支持。

■ Qt Test

提供供 Qt 应用程序和类库使用的单元测试工具。

■ Qt Widgets

提供针对 Qt GUI 的扩展 C++ 组件。

如果使用 qmake 生成项目，则默认包含 Qt Core 和 Qt GUI 模块。如仅针对 Qt Core 进行链接，请将以下行添加到 `.pro` 文件中：

```
1 QT -= gui // Only the core module is used.
```

在 Windows 系统中，如果不使用 qmake 或其他构建工具（例如 CMake），则还需要链接到 qtmain 库。

2. Qt 扩展模块

Qt 扩展模块是为特定目的添加附加功能，这些模块可能仅在某些开发平台上可用。许多扩展模块要么功能完整，要么为了向后兼容而存在，或者仅适用于某些平台。每个扩展模块都有指定的兼容性说明。

Qt 安装程序包括下载扩展模块的选项，下面 Qt 扩展模块及其简介：

- **Qt 3D**

提供支持 2D 和 3D 渲染的近实时仿真系统的功能。

- **Qt Android Extras**

提供适用于 Android 平台的特定 API。

- **Qt Bluetooth**

提供对蓝牙硬件的访问。

- **Qt Canvas 3D**（弃用）

使用 JavaScript 从 Qt Quick 应用程序中启用类似于 OpenGL 的 3D 绘图调用。

- **Qt Concurrent**

封装了底层线程技术的类库，方便开发多线程程序。

- **Qt D-Bus**

提供利用 D-Bus 协议进行进程间交互的模块。

- **Qt Gamepad**

提供使 Qt 应用程序支持游戏手柄硬件的模块。

- **Qt Graphical Effects**

与 Qt Quick 2 一起，提供一系列用于实现图像特效的模块。

- **Qt Help**

用于将文档集成到应用程序中的模块，类似于 Qt Assistant。

- **Qt Image Formats**

提供支持图片格式的一系列插件，包括 TIFF、MNG、TGA 和 WBMP。

- **Qt Location**

在 QML 应用程序中提供地图、导航和位置搜索等功能。

- **Qt Mac Extras**

为 macOS 平台提供特定的 API 支持。

- **Qt NFC**

提供对 NFC 硬件的访问。

- **Qt OpenGL** （弃用）

提供支持 OpenGL 的模块，不推荐使用 Qt GUI 模块中的 `QOpenGL *` 类。

- **Qt Platform Headers**

提供封装特定平台信息的类，这些类与平台插件给定的运行配置相关联。

- **Qt Positioning**

提供对位置、卫星和区域监视的访问。

- **Qt Print Support**

提供对打印功能的支持。

- **Qt Purchasing**

启用 Qt 应用程序产品的购买。

- **Qt Quick Controls 1** （弃用）

基于可重用的 Qt Quick 的 UI 控件可创建经典样式用户界面。

- **Qt Quick Extras**

提供一组专门的控件，可用在 Qt Quick 中构建界面。

- **Qt Quick Widgets**

提供用于显示 Qt Quick 用户界面的 C++ 小部件模块。

- **Qt Remote Objects**

提供一种易于使用的机制，用于在进程或设备之间共享 QObject 的 API（属性/信号/插槽）。

- **Qt Script** （弃用）

提供 Qt 应用程序的脚本化机制，不推荐使用 Qt QML 模块中的 `QJS *` 类。

- **Qt SCXML**

提供用于从 SCXML 文件创建状态机并将其嵌入到应用程序中的类和工具。

- **Qt Script Tools** （弃用）

为使用了 Qt Script 模块的应用程序提供的额外的组件。

- **Qt Sensors**

提供对传感器硬件和移动手势识别的访问。

- **Qt Serial Bus**

提供对串行工业总线接口的访问。当前，该模块支持 CAN 总线和 Modbus 协议。

- **Qt Serial Port**

提供对硬件和虚拟串行端口的访问。

- **Qt Speech**

提供对可访问性功能（如文本到语音）的支持。

- **Qt SVG**

提供渲染和创建 SVG 文件功能的类，支持 SVG 1.2 Tiny 标准的子集。

- **Qt UI Tools**

用于在运行时动态加载在 Qt Designer 中创建的基于 QWidget 的类。

- **Qt WebChannel**

提供从 HTML 客户端访问 QObject 或 QML 对象的功能，以将Qt应用程序与 HTML / JavaScript 客户端无缝集成。

- **Qt WebEngine**

使用 Chromium 浏览器项目在应用程序中嵌入 Web 内容的类和函数。

- **Qt WebSockets**

提供符合 RFC 6455 的 WebSocket 通信。

- **Qt WebView**

通过使用平台固有的 API 在 QML 应用程序中显示 Web 内容，而无需包含完整的Web浏览器堆栈。

- **Qt Windows Extras**

为 Windows 平台提供特定的 API 支持。

- **Qt X11 Extras**

为 X11 平台提供特定的 API 支持。

- **Qt XML**

SAX 和 DOM 的 C++ 实现。

注意：使用 QDomStreamReader 和 QDomStreamWriter 不需要此模块，它们已经是 Qt Core 的一部分。

- **Qt XML Patterns**

支持 XPath, XQuery, XSLT 和 XML 模式验证。

- **Qt Wayland Compositor**

提供开发 Wayland 合成器的框架。

根据商业许可或 GNU 通用公共许可 v3 提供的扩展组件

- **Qt Charts**

用于显示视觉上令人愉悦的图表，该图表由静态或动态数据模型驱动。

- **Qt Data Visualization**

用于创建 3D 数据可视化。

- **Qt Network Authorization**

为基于 OAuth 的授权提供在线服务支持。

- **Qt Virtual Keyboard**

用于实现不同输入法的框架以及 QML 虚拟键盘，支持本地化的键盘布局和自定义视觉主题。

- **Qt Quick WebGL**

提供一个插件平台，该插件允许 WebGL™ 通过网络流传输 Qt Quick 用户界面。

3. Qt 增值模块

除了作为 Qt 5 的一部分模块发布之外，以下模块和工具还基于 Qt 库提供了附加价值。它们有自己的发布时间表，可以根据商业许可获得。

- **Qt Automotive Suite**

用于开发车载信息娱乐（IVI）系统的软件组件和工具的集合。建立在 Qt for Device Creation 产品之上，并符合 GENIVI 汽车平台架构。

- **Qt for Automation**

用于自动化相关领域的库和工具，例如 KNX，OPC UA 和 MQTT。

- **Qt for Device Creation**

提供用于简便快速且完全集成的嵌入式设备应用程序开发的工具，包含在大多数其他增值解决方案中。

4. 技术预览模块

以下功能仍在开发中，作为技术预览模块：

- **Qt for WebAssembly**

使用 WebAssembly 为 Web 浏览器编译 Qt 应用程序的目标平台。

5. Qt 工具

Qt Tools 在所有支持的开发平台上运行，并促进了应用程序的开发和设计。

- **Qt Designer**

提供扩展 Qt Designer 的模块。

6. 未来方向

邀请 Qt 用户探索 Qt 的其它功能。准备了概述以帮助用户确定要使用的 API，示例演示了如何使用这个 API。

- **Qt Overviews**

有关应用程序开发的主题列表。

- **Examples and Tutorials**

代码示例和教程。

- **Qt Reference Pages**

C++ 和 QML API 清单。

小结

Qt 5 与 Qt 4 最大的一个区别之一是底层架构有了修改。Qt 5 引入了模块化的概念，将众多功能细分到几个模块之中。Qt 4 也有模块的概念，但是是一种很粗的划分，而 Qt 5 则更加细化。本篇参考 Qt 帮助文档，主要对 Qt 5 的模块进行一个简单的介绍，以便大家在需要使用一些功能的时候知道上哪个模块去寻找。

相关阅读：

《信号与槽》

《代码化 UI 设计》

《可视化 UI 设计（设计器 Qt Designer 实现）》

《UI 文件设计与运行机制》

《编写一个 Hello World 程序》



喜欢此内容的人还喜欢

从素人到登上VOGUE，需要几步？

VOGUE



一张上海打工人真实薪资表曝光：抱歉，这才是99%中国人的工资真相

职场充电宝

