Q搜索

Qt设计工作室手册 > 将 QML 模块与插件结合使用

将QML模块与插件结合使用

OML 模块可以使用C++插件将C++中定义的组件公开给 OML 应用程序。

要创建 QML 模块并使其显示在"组件"视图中, 请执行以下操作:

- 1. 创建自定义组件并将所有文件放在专用于模块的目录中。例如: 。.qmlimports\asset_imports
- 2. 对于Qt Quick UI项目(.qmlproject),请在应用程序的.qmlproject文件中指定包含模块的目录的路径,其中要将模块用作变量的值。例如。importPathsimportPaths: ["imports", "asset_imports"]
- 3. 为模块创建一个文件并将其放在模块目录中。有关详细信息,请参阅模块定义 qmldir 文件。qmldir
- 4. 按照生成类型描述文件中的说明创建文件。qmltypes

Qt设计工作室手册3.8.0 Topics >

件为例。.metainfodesignerqtquickcontrols2.metainfo

7. 使用与Qt Design Studio相同的Qt版本和编译器构建模块。有关更多信息,请参阅在设计模式下运行 QML 模块。

您的模块现在应显示在"组件"中。如果存在有效文件,则组件应显示在组件的子部分中。.metainfo

生成类型描述文件

注册 QML 类型时,请确保 QML 模块具有文件。理想情况下,它应该与文件位于同一目录中。该文件包含由模块插件导出的组件的描述,并在导入模块时由Qt Design Studio加载。plugins.qmltypesqmldirqmltypes

有关详细信息,请参阅类型说明文件。

自动转储插件

如果带有插件的模块缺少文件,Qt Design Studio会尝试通过在后台运行程序来生成临时文件。但是,这种自动转储是一种具有许多故障点的回退机制,您不能依赖它。gmltypesgmldump

导入 QML 模块

默认情况下,Qt Design Studio 将查找 Qt for QML 模块的 QML 导入路径。

如果您使用 qmake 并且您的应用程序添加了 Qt Design Studio 应该使用的其他导入路径,请在应用程序的文件中指定它们:QML_IMPORT_PATH.proQML_IMPORT_PATH += path/to/module

如果使用 CMake,请将以下命令添加到 CMakeLists.txt 文件中以设置 QML 导入路径:

Qt DOCUMENTATION

导入路径会影响 CMake 项目生成的所有目标。

在设计模式下运行 QML 模块

QML 仿真层(也称为 QML Puppet)在**设计**模式下用于渲染和预览图像以及收集数据。为了能够从QML模块正确渲染自定义组件,必须使用与QML模块相同的Qt版本和编译器构建仿真层。

在Windows上,选择**Help**>About Qt Design Studio以检查构建插件所需的**Qt**版本和编译器。例如:。Based on **Qt** 5.15.2 (MSVC 2019, 64 bit)

在 macOS 上,选择 Qt Design Studio>**About Qt Design Studio**以查看类似以下内容: Based on Qt 5.15.2 (Clang 10.0 (Apple), 64 bit)

插件的行为应该有所不同,具体取决于它是由仿真层还是应用程序运行。例如,不应在"设计"模式下运行动画。您可以使用环境变量的值来检查插件当前是否正在由应用程序运行或在设计模式下编辑。QML_PUPPET_MODE

如果要在**设计**模式下使用与实际应用程序不同的模块,例如C++项的模型,则可以在文件中使用(对于 qmake 项目),或在产品中声明和设置属性(对于 Qbs 项目)。导入路径中定义的模块将仅在**设计**模式下使用。有关示例,请参阅Qt 快速控件 - 联系人列表。

QML_DESIGNER_IMPORT_PATH.proqmlDesignerImportPathsQML_DESIGNER_IMPORT_PATH

〈模拟动态系统 动态行为〉











联系我们

 关于我们
 条款和条件

 投资者
 开源

 编辑部
 常见问题

职业

办公地点

 支持服务
 支持中心

 专业服务
 下载

 合作伙伴
 Ot登录



社区

为Qt做贡献

论坛

维基

下载

市场

© 2022 Ot公司

反馈 登录