Qt DOCUMENTATION

Qt 设计工作室手册 3.7.0

搜索

Topics >

将 Qt 桥接用于 Adobe XD

组织资产

要在使用适用于 Adobe XD 的 Qt 桥将设计从 Adobe XD 导出到 Qt 设计工作室时获得最佳效果，请在使用 Adobe XD 时遵循以下准则：

> 将您的设计排列到*画板中*，并将其组织成组和图层，这些组和图层作为单独的文件导入到Qt Design Studio中。

> 使用描述性和唯一 ID，以避免在生成的 UI 中重复资产名称和 ID。

> 使用 XD 组件和实例重用 UI 元素。

注意： 尽管适用于 Adobe XD 的 Qt 桥保留了 XD 组件和实例的关系，但尚不支持覆盖和状态。

要在 Adobe XD 中使用的字体也用于 Qt 设计工作室，您需要将它们作为资源导入到 Qt 设计工作室。Qt 设计工作室会在您预览 UI 时将它们部署到设备上。有关详细信息，请参阅使用自定义字体。

支持的设计元素

您可以使用适用于 Adobe XD 的 Qt 桥接导出设计的以下部分：

> 画板

> 群

> 形状（矩形、椭圆形、多边形、线条、笔）

> 发短信

以下设计元素可能无法按预期导出。

> 组件状态

> 组件覆盖

> 原型

> 重复网格

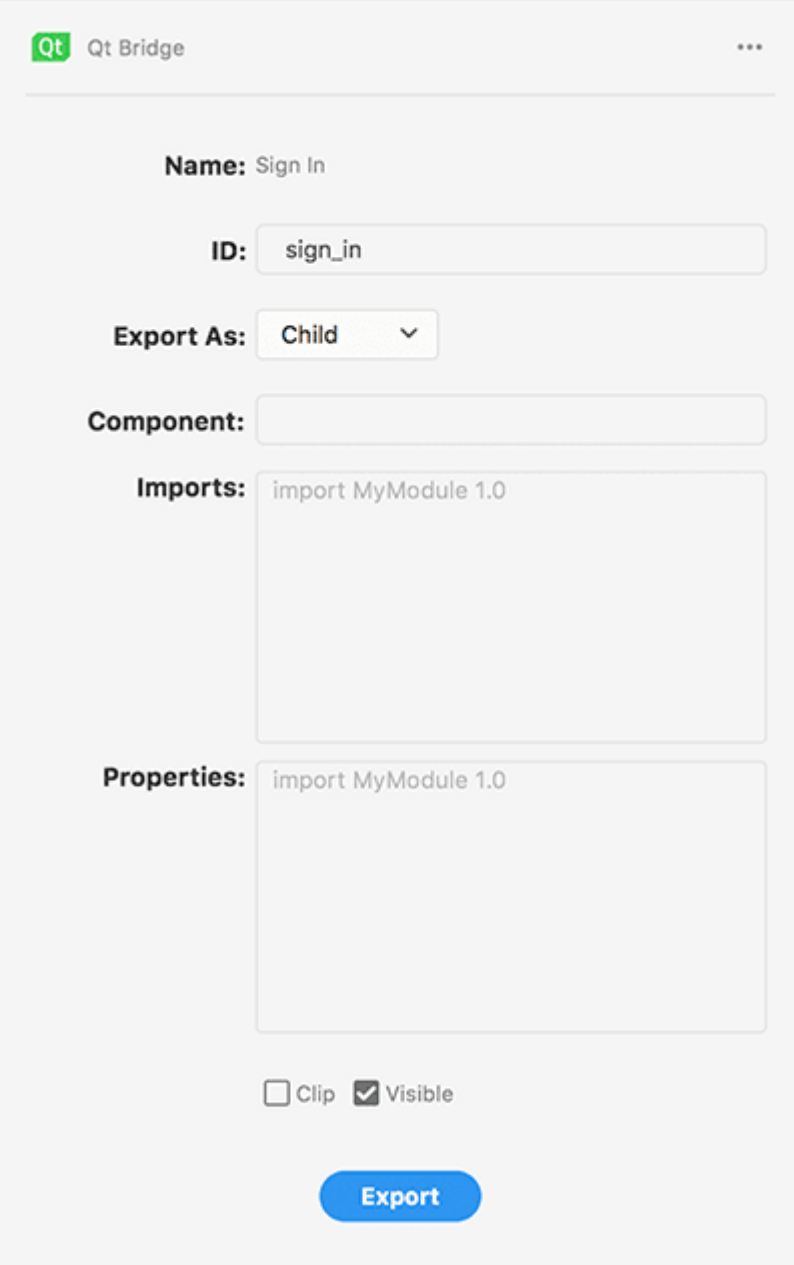**注意：** Adobe XD 的插件 API 支持不完整。具体来说，对 XD 组件的支持是有限的。将来可能会更改此设置，适用于 Adobe XD 的 Qt 桥可能会扩展 XD 组件支持。

Qt DOCUMENTATION

当您从 Adobe XD 导出设计并将其导入 Qt 设计工作室时，画板和图层之间的层次结构关系将保留。这些关系是项目在Qt设计工作室中生成的代码中组织的基础。

画板只能作为组件导出或跳过。组件将作为包含美工板上的图稿和文本的单独文件导入。

注释图层以进行导出

借助适用于 Adobe XD 的 Qt 桥接，可以对图层进行批注，以提示必须如何导出每个图层或组。{**主页**} 面板显示并允许导出图层注释：

1. Qt Bridge for Adobe XD automatically proposes identifiers for all groups and layers that you can change in the **ID** field. Use unique and descriptive IDs to avoid duplicate IDs when the layer and the respective artwork is imported into Qt Design Studio. Even though the importer in Qt Design Studio is capable of fixing duplicate IDs, doing so will generate warnings. It is recommend that you should manually check all the IDs to make them unique and descriptive.

2. In the **Export As** field, select the export type for the group or layer:

   > **Component** exports the layer as a separate UI file that contains all the exportable artwork and text in it. Only Artboards can be exported as components.

Qt DOCUMENTATION

> **Merged** merges the rendition of the selected groups and layers into the rendition of parent artboard or group as one item.

> **Skipped** completely skips the selected layer.

3. In the **Component** field, specify the component or Qt Quick Studio Component to morph this layer into. The component that is generated during import will be of this type. For example, if you drew a rectangle, you can export it as a Rectangle component. You can provide the import statement of the module where the component is defined in the **Imports** field.

> **Note:** The implicit properties except position and size are not applied when the **Component** is defined. For example, all text properties will be ignored if **Component** is defined for a text layer, but explicit properties defined in the **Properties** field will be applied.

4. Select the **Render Text** check box to render the text layer as an image

5. In the **Imports** field, enter additional import statements to have them added to the generated code file. For example, to use Qt Quick Controls 2.3, you need the import statement and to use Qt Quick Studio Components 1.0, you need the import statement . You can also import a module as an alias.`QtQuick.Controls 2.3``QtQuick.Studio.Components 1.0`

6. In the **Properties** field, specify properties for the component. You can add and modify properties in Qt Design Studio.

7. Select the **Clip** check box to enable clipping in the component generated from the layer. The generated component will clip its own painting, as well as the painting of its children, to its bounding rectangle.

8. Select the **Visible** check box to determine the visibility of the layer in the generated UI in Qt Design Studio.

9. Select **Export** to launch the export dialog to export the document into a .qtbridge archive.

> **Note:** XD Components can not be skipped and Text layers can only be merged when **Render Text** is selected.

Export Defaults

Qt Bridge for Adobe XD assigns the following defaults to the layers:

By default:

> Artboards and XD Components are exported as *components*.

> Component instances, Text layers and immediate children of an Artboard are exported as *child*.

> Any layer not falling under the aforementioned criteria is exported as *merged*.

> Images are exported as PNGs by default with no Hi-DPI images.
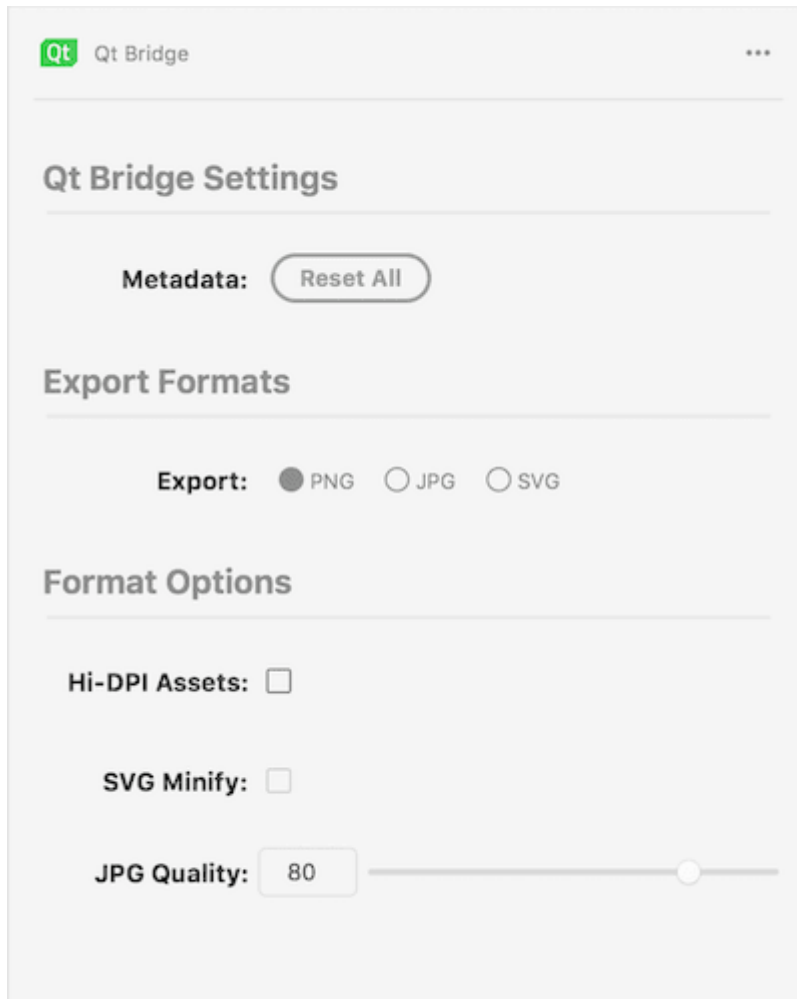
> **Visible** is set to .`true`

All the assets and metadata are copied to the directory you specified. This might take a little while depending on the complexity of your project.

You can now create a project in Qt Design Studio and import the assets to it, as described in Creating Projects and Importing Designs.

Qt Bridge for Adobe XD Settings

Qt **DOCUMENTATION**

Home
✓ Settings
License
Help
About Qt-Bridge

Manage Plugin

Qt Qt Bridge   • • •

Qt Bridge Settings

Metadata: (Reset All)

Export Formats

Export: ● PNG   ○ JPG   ○ SVG

Format Options

Hi-DPI Assets: ☐

SVG Minify: ☐

JPG Quality: [80] ────────────────○──────

1. Select **Reset All** to remove all of the Qt Bridge data stored in the document. Use **Edit** > **Undo** to restore the data if you accidentally removed it.

2. You can export images into PNG, JPG or SVG format. In the section **Export Formats**, select the image format to export.

3. Depending on the image format selected for export, the **Format Options** allows fine tuning the exported image.

    > Select **Hi-DPI Assets** to generate Hi-DPI images alongside normal scale images. Being a vector format, this option is not available for SVG format.

    > Select **SVG Minify** to minify the SVG.

    > Select **JPG Quality** to specify the JPG compression quality in the range [1, 100].

Suggestions and Tips

Qt DOCUMENTATION

example, you can specify the component or Qt Quick Studio Component to use for a component or layer. If you have drawn an arc that you mean to animate, you can export it as an Arc component to avoid having to replace the arc image with an Arc component in Qt Design Studio. Or you could export a button as a Qt Quick Controls Button component.

› Name the layers in exactly the same way as your IDs, to be able to find artwork later, especially as the export files can grow very large and complicated as they approach the level of a complete UI project.

› Make sure to skip all artboards that you don't want to be part of the final UI to avoid cluttering the Qt Design Studio project. You can select multiple artboards on a page and then select **Skip** in Qt Bridge for Adobe XD to skip them.

› Store all assets in the scalable vector graphics (SVG) format to be able to easily rescale them for different screen sizes and resolutions. You can export assets into JPG, PNG, or SVG format and select options for optimizing them during the export.

The Qt Company

Contact Us

Company

About Us

Investors

Newsroom

Careers

Office Locations

Licensing

Terms & Conditions

Open Source

FAQ

Support

Support Services

Professional Services

Partners

Training

For Customers

Support Center

Downloads

Qt Login

Contact Us

Customer Success

Community

Qt DOCUMENTATION

Wiki

Downloads

Marketplace

Feedback   Sign In