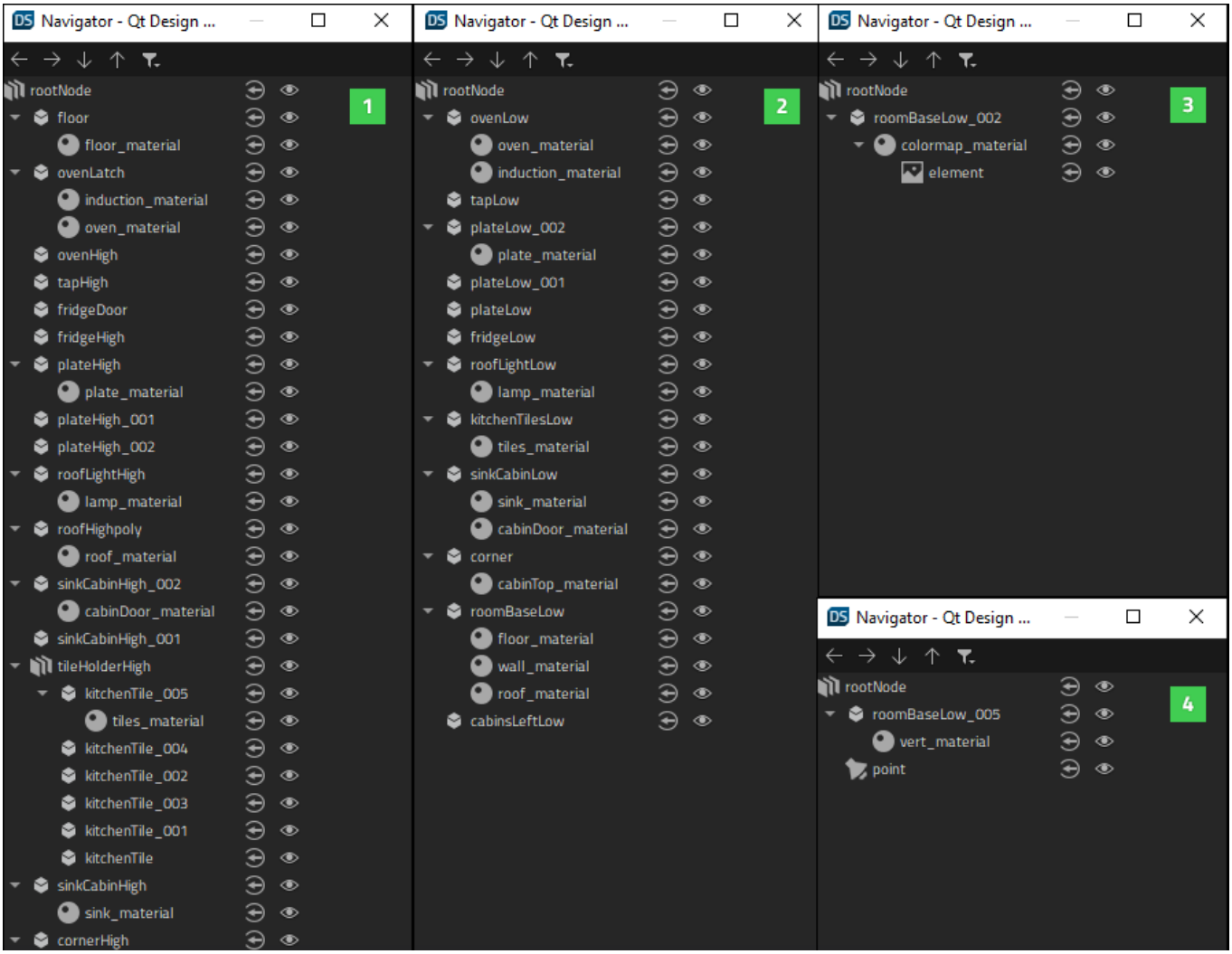
创建优化的 3D 场景

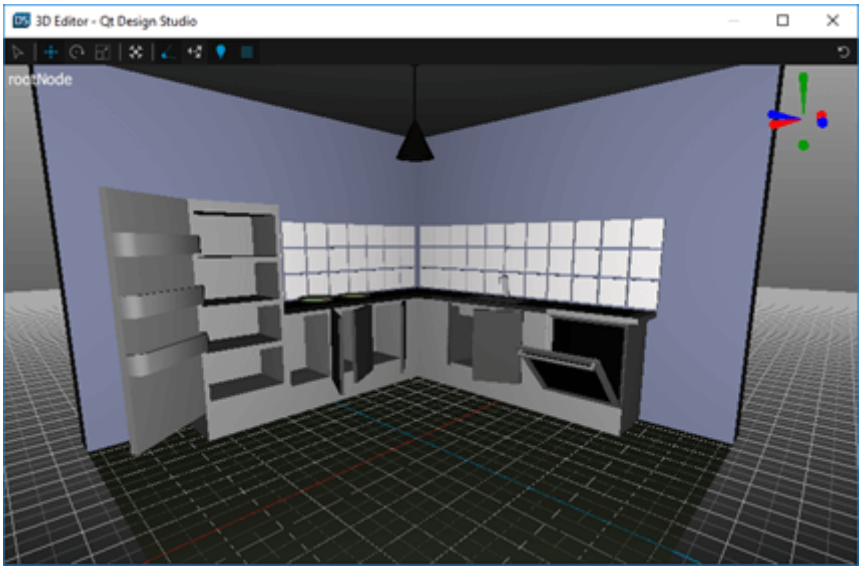
在Qt设计工作室中，您可以使用各种方法来创建3D场景。您选择的策略应始终取决于场景的目标平台。场景内容的创作方式可能会对 UI 的运行时性能产生巨大影响。下面描述的最佳 3D 场景说明了创建场景的四种不同方法。另请参阅优化 3D 场景图形性能的指南和基于用例的 3D 场景最佳实践。

最佳 3D 场景示例

最佳 3D 场景示例包含使用不同策略创建的同一厨房场景的四个版本：高、低、组合和顶点颜色。与其他版本的场景相比，“高场景”包含的对象数量明显更多。“低”、“组合”和“顶点”场景包含的对象较少，因为它们已经过优化，例如，通过将几何体和烘焙材质组合到纹理中。下图显示了 Navigator 中每个场景中的组件，以说明场景之间的结构差异。

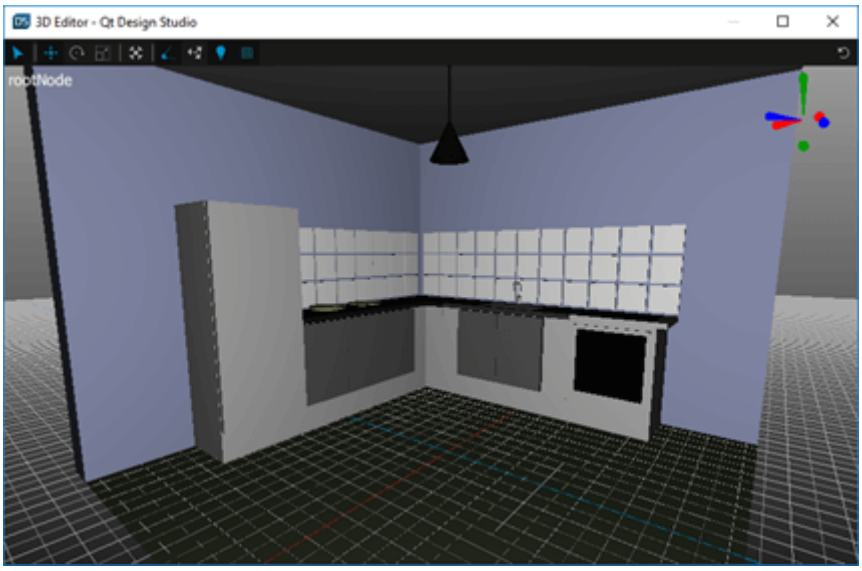


高地场景



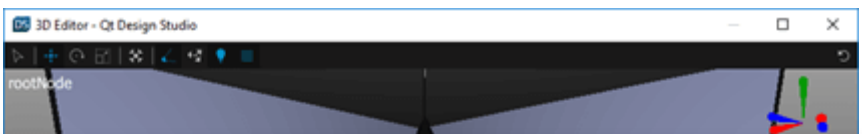
“高”场景 （1） 总共包含 27 个对象。该场景在其他方面类似于“低”场景，但具有更多细节和动画组件，例如可以打开的机柜门和烤箱的闩锁。

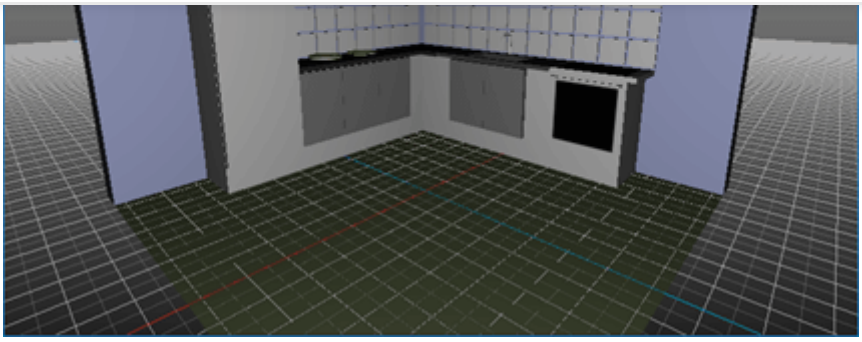
低谷场景



The Low scene (2) includes a total of 12 objects. Each model, for example cabinets, lamps, plates, and the sink, are separate meshes.

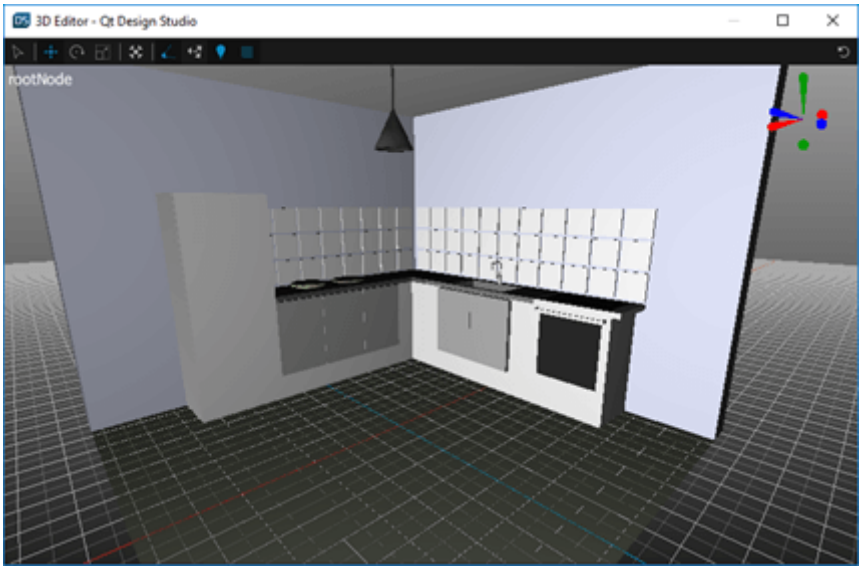
The Combined Scene





The Combined scene (3) has all the meshes combined into a single object. All materials are baked into one single texture.

The Vertex Color Scene



The Vertex Color scene (4) has all the meshes combined into a single object, and colors for the mesh are provided from .mesh file's vertex data, which includes one material and no textures.

Optimizing Your 3D Scene

To optimize the graphics performance of your 3D scene, you should first consider whether you need to animate the 3D objects, lights, camera, or other components in your scene. If not, simplify the scene structure by, for example, combining geometries and baking materials into textures. Ideally, pre-render your 3D image into a 2D image. You should not waste resources by rendering static 3D objects. You should also check if your scene has multiple instances of the same mesh. If so, import only one instance of the mesh, duplicate it in the scene, and use the same material for each duplicate.

See also the following guidelines related to scene graph, asset complexity, CPU specific optimization, and model geometry.

Scene Graph

The scene graph is the hierarchy of nodes that describe the scene to be rendered.

In Qt Design Studio, the scene graph is represented by the tree-like view in **Navigator**. You can also view the hierarchy of nodes in the Code view. By minimizing the size of the scene graph, you can minimize the effort needed when running the scene. In terms of optimization, you should avoid unnecessary groups and complex hierarchy

Asset Complexity

The performance of your UI or scene may sometimes be bottlenecked at render time. Avoid this by simplifying your assets.

For 3D models, you should try to use the minimum number of triangles or vertices to achieve the desired result. Also note that smaller images require less memory and render faster.

Optimizing CPU Performance

Rendering objects requires plenty of processing work from the CPU. Processing each component consumes resources, so reducing the visible object count reduces the amount of work required from the CPU.

To reduce the visible object count:

- Combine objects that are positioned closely and share a texture.

> **Note:** Combining objects that do not share a texture does not increase the CPU performance while running the scene.

- Use fewer materials by combining separate textures into a single texture atlas (like in the Merged scene in the Kitchen Scene Example).
- Minimize the number of factors that cause objects to be rendered multiple times, for example reflections and shadows.

Optimizing Model Geometry

To optimize the geometry of a model:

- Minimize the number of triangles used in the model.
- Minimize the number of UV mapping seams and hard edges created with doubled-up vertices.

Best Practices for 3D Scenes Based on Use Cases

The following table summarizes best practices related to optimizing 3D scenes in the form of use cases.

How to...	Solution
Get best performance from the 3D scene.	If no animation is required, pre-render your 3D scene into a 2D image.
Include animation for some of the objects in the scene (for example, to open doors).	Merge all objects that are static, and export only one of each animated component (for example, each door type).
Get best performance with animated camera.	While using a 2D image is not an option in this case, you should combine all geometry and bake lighting and materials into a single material.
Get best performance with animated light.	While using a 2D image is not an option in this case, you should combine all geometry and bake all materials into a single material.
Get best performance in a 3D scene where all meshes are separated.	Use the same material for each instance of the same mesh.
How to...	Solution

Use baked textures with low memory.	Bake mesh colors into vertex colors (see the Vertex Color Scene in the example project).

‹ Optimizing Designs

Implementing Applications ›