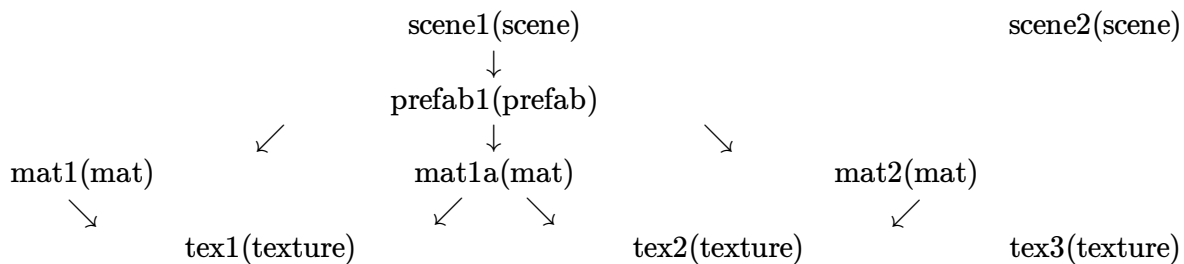


关于 AssetBundle 的一些实践

🕒 Created	@December 22, 2022 2:58 PM
🏷️ Tags	

测试资源引用



打包批次

- **origin** : `prefab` (prefab1) | `mat1` (mat1) | `mat1a` (mat1a) | `mat2` (mat2) | `tex` (tex1,tex2)
- **origin-a** : `prefab` (prefab1) | `mat1` (mat1) | `mat1a` (mat1a) | `mat2` (mat2)
- **origin-b** : `mat1a` (mat1a) | `tex-rename` (tex1,tex2)
- **origin-c** : `tex` (tex3)
- **scene-a** : `scene1` (scene1)
- **scene-a1** : `scene1` (scene1 → scene1-rename) 【这里手动改资源名后打包】
- **scene-a2** : `scene1` (scene1) 【名字没变但是换了路径】
- **scene-b** : `scene1-rename` (scene1)
- **scene-c** : `scene1` (scene2)
- **scene-c1** : `scene1` (scene2 → scene1) 【名字改成和scene1相同，但是放在不同路径】
- **scene-d** : `scene` (scene1,scene2)

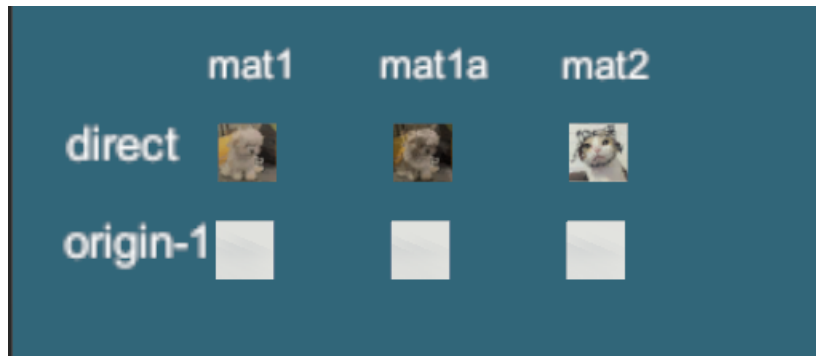
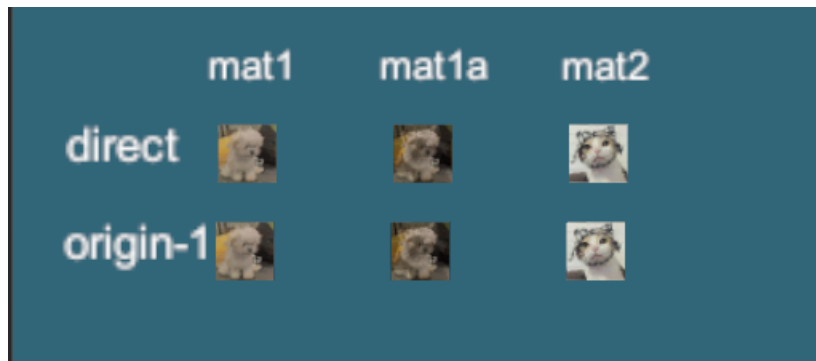


Unity会在同一批次打包从处理依赖关系。

测试

1. origin-1

- 加载包顺序
 - `prefab` (origin) → `mat1` (origin) → `mat1a` (origin) → `mat2` (origin) → `tex` (origin)
 - `prefab` (origin) → `mat1` (origin) → `mat1a` (origin) → `mat2` (origin)
 - `prefab` (origin) → `mat1a` (origin)
- 显示效果



- 输出
无报错。

- 结论

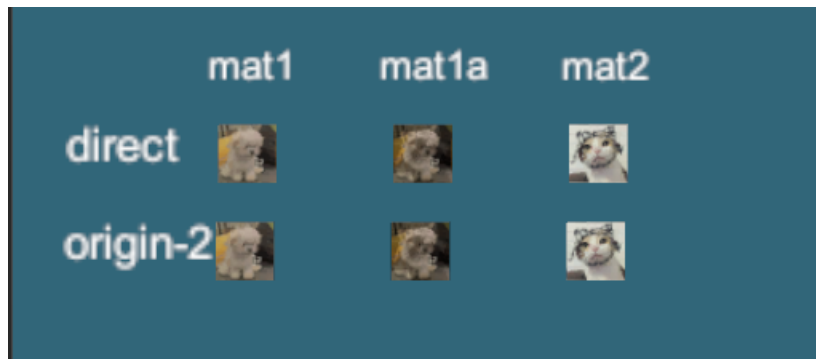
◦ 未加载引用Asset所在的AssetBundle，Asset会Missing，但是不会报错。

2. origin-2

- 加载包顺序

- `prefab (origin-a) → mat1 (origin-a) → mat1a (origin-a) → mat2 (origin-a)`
- `prefab (origin-a) → mat1a (origin-a)`

- 显示效果



- 输出
无报错。

- 结论

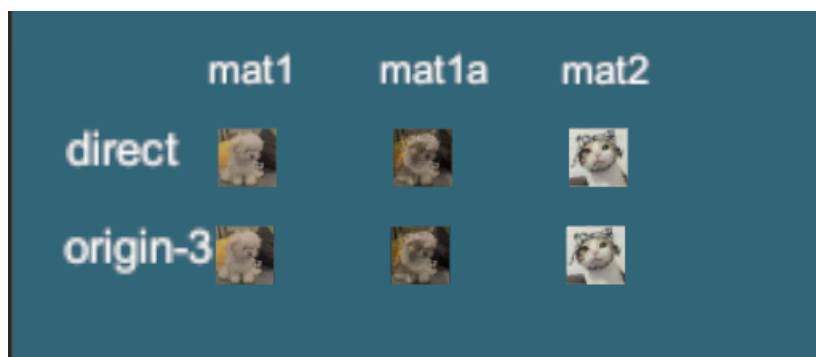
- 未指定AssetBundle的Asset在被引用时会复制进引用其的资源所在的AssetBundle中，可能会复制到多个AssetBundle中，不会报错和冲突。

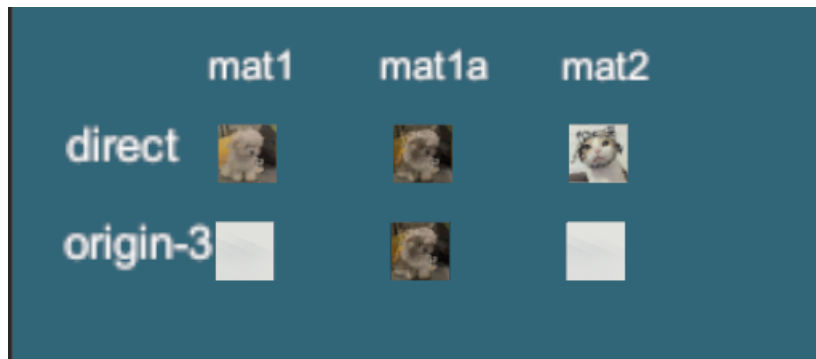
3. origin-3

- 加载包顺序

- `prefab(origin-a)` → `mat1(origin)` → `mat1a(origin-a)` → `mat2(origin)` → `tex(origin)`
- `prefab(origin-a)` → `mat1(origin)` → `mat1a(origin-a)` → `mat2(origin)`

- 显示效果

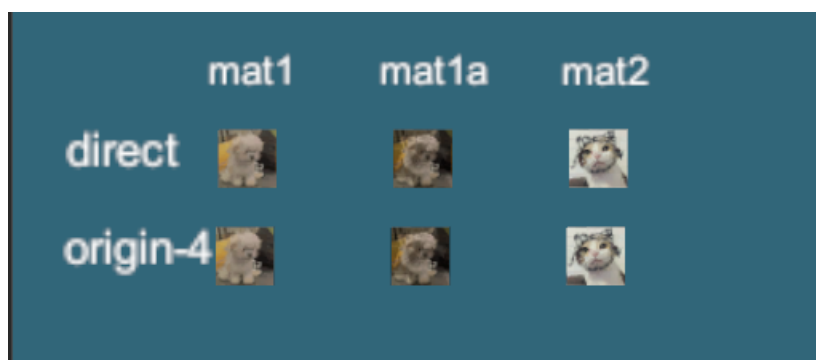


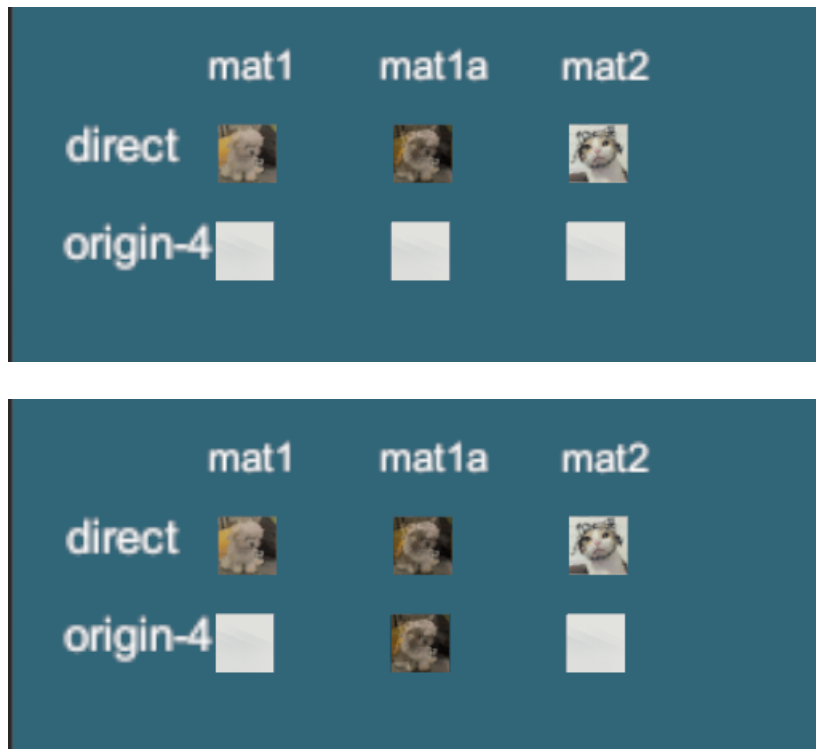


- 输出
无报错。
- 结论
 - 不同批次的AssetBundle间可以互相引用（这是AssetBundle资源更新的基础）。
 - 同一Asset可以既存在在一个显式指定AssetBundle中，又存在在被隐式复制到的AssetBundle中，不会报错和冲突。

4. origin-4

- 加载包顺序
 - `prefab (origin) → mat1 (origin) → mat1a (origin) → mat2 (origin) → tex (origin) → tex (origin)`
 - `prefab (origin) → mat1 (origin) → mat1a (origin) → mat1a (origin-a) → mat2 (origin)`
 - `prefab (origin) → mat1 (origin) → mat1a (origin-a) → mat1a (origin) → mat2 (origin)`
- 显示效果





- 输出

- The AssetBundle 'tex' can't be loaded because another AssetBundle with the same files is already loaded.
- The AssetBundle 'mat1a' can't be loaded because another AssetBundle with the same files is already loaded.
- The AssetBundle 'mat1a' can't be loaded because another AssetBundle with the same files is already loaded.

- 结论

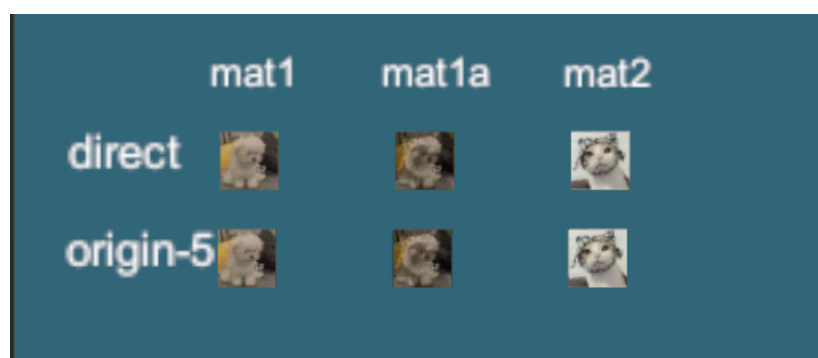
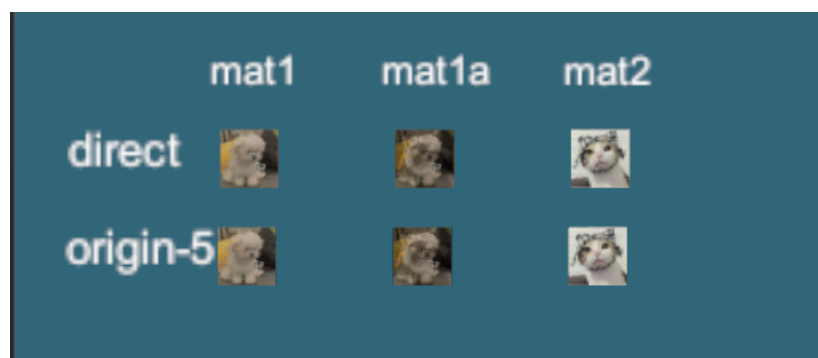
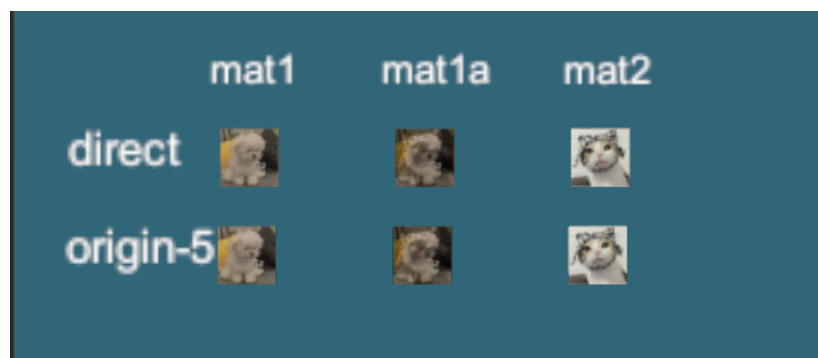
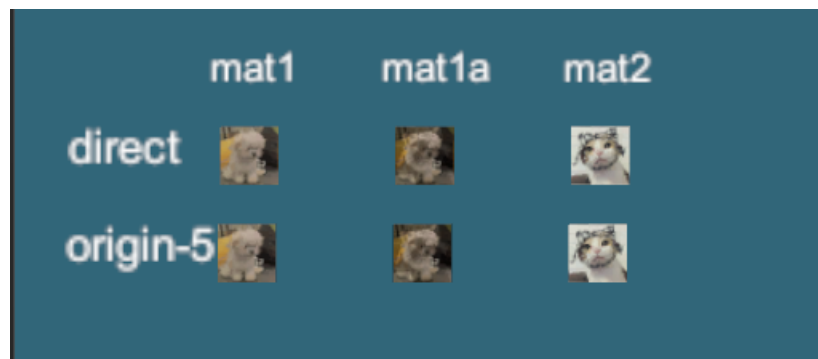
- 包含相同Asset的相同名字的AssetBundle只允许同时加载一次，后面重复的加载会报错并失效。

5. origin-5

- 加载包顺序

- `prefab (origin) → mat1 (origin) → mat1a (origin) → mat2 (origin) → tex (origin) → tex-rename (origin-b)`
- `prefab (origin) → mat1 (origin) → mat1a (origin) → mat2 (origin) → tex (origin) → tex (origin-c)`
- `prefab (origin) → mat1 (origin) → mat1a (origin) → mat2 (origin) → tex (origin) → tex-rename-tex (origin-b)` 【这里打包后手动改文件名】
- `prefab (origin) → mat1 (origin) → mat1a (origin) → mat2 (origin) → tex (origin) → tex-tex-rename (origin-c)` 【这里打包后手动改文件名】

- 显示效果



- 输出
 - 无报错。
 - `The AssetBundle 'tex' can't be loaded because another AssetBundle with the same files is already loaded.`

- 无报错。
- `The AssetBundle 'tex-rename' can't be loaded because another AssetBundle with the same files is already loaded.`
- 结论
 - 相同名字的AssetBundle同时只能加载一次，这个名字指打包时候指定的AssetBundleName，与生成的文件名无关，与AssetBundle中的Asset无关。

6. scene-1

- 加载包顺序
 - `scene1` (scene-a) → `scene1-rename` (scene-b)
 - `scene1` (scene-a) → `scene1` (scene-c)
 - `scene1` (scene-a) → `scene` (scene-d)
- 显示效果

无
- 输出
 - `The AssetBundle 'scene1-rename' can't be loaded because another AssetBundle with the same files is already loaded.`
 - 无报错。
 - `Scene 'Assets/Res/origin/scene1.unity' from AssetBundle 'scene' is already registered with another AssetBundle 'scene1'.`
- 结论
 - 包含相同Scene的AssetBundle即使打包时候指定的AssetBundleName不同，也不能同时加载多次。

7. scene-2

- 加载包顺序
 - `scene1` (scene-a) → `scene1` (scene-a1)
 - `scene1` (scene-a) → `scene1` (scene-a2)
 - `scene1` (scene-a) → `scene1` (scene-c1)
- 显示效果

无
- 输出
 - 无报错。

- The AssetBundle 'scene1' can't be loaded because another AssetBundle with the same files is already loaded.
- The AssetBundle 'scene1' can't be loaded because another AssetBundle with the same files is already loaded.
- 结论
 - 包含相同名字Scene的AssetBundle即使打包时候指定的AssetBundleName不同，也不能同时加载多次。