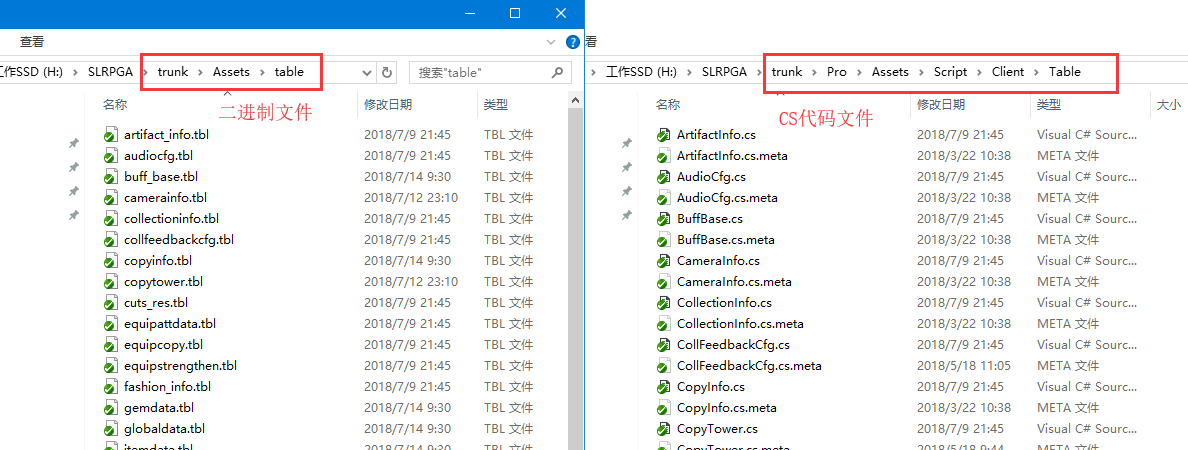
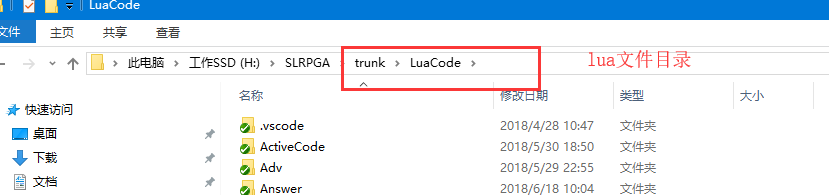
# 正式发布

## 发布前的准备工作

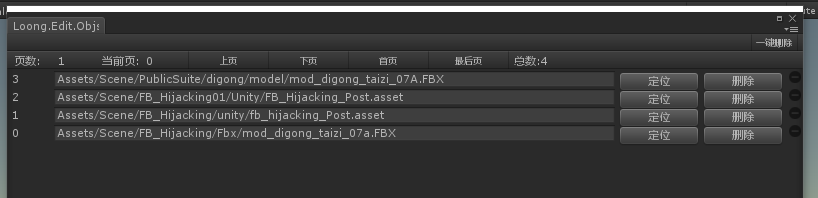
1. 确认已SVN更新到最新版本
2. 确认C#配置提交，要确认二进制文件和CS代码文件都已经提交，检查目录如下：



1. 确认Lua文件提交



1. 检查所有资源是否合法命名
2. ，选择要检查的文件夹，右键菜单【Developer/Loong/资源工具/检查非法字符】
3. ，检查Assets/Pkg和Assets/Scene文件夹
4. ，有非法命名一定要处理掉
5. 检查是否存在多个资源名称和AB名称相同问题
6. ，菜单【Developer/Loong/资源包工具/搜索文件名AB名相同】
7. ，如果存在，弹出窗口中会出现如下类似内容，必须解决：

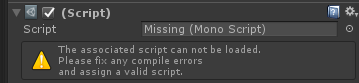


1. 检查是否存在需要设置包名但未设置造成资源冗余问题

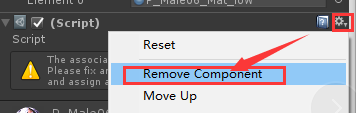
菜单【Developer/Loong/资源包工具/搜索未设置AB资源】

* 1. ，如果存在，弹出窗口中，必须适当解决
  2. ，如果存在资源是属于Standard Assets 目录内，可能要酌情查设置，因为此目录内的资源按理是不参与更新的

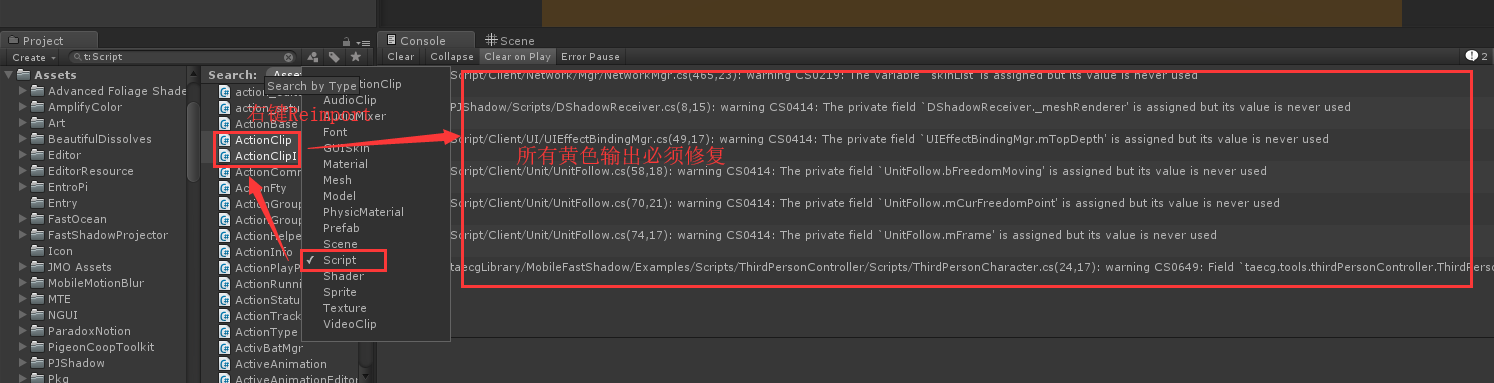
1. 检查是否有丢失脚本，丢失示例：



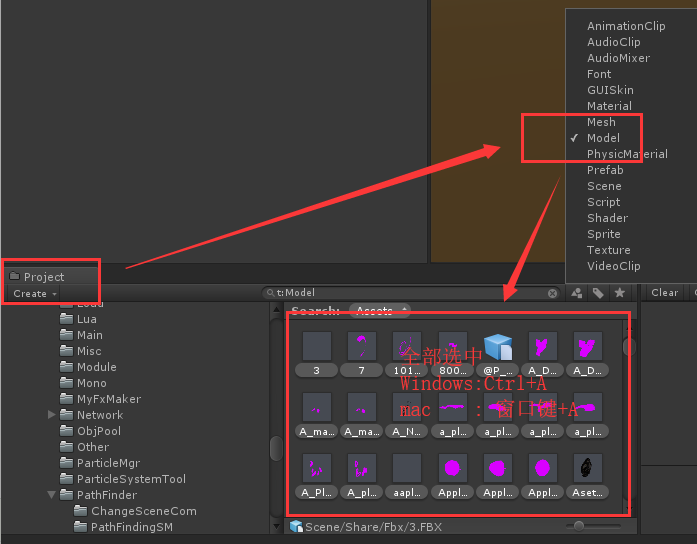
* 1. ，选择所有的prefab，右键【Developer/Loong/脚本工具/丢失工具/打开选择对象的丢失脚本定位窗口】
  2. ，将所有丢失脚本的组件移除：



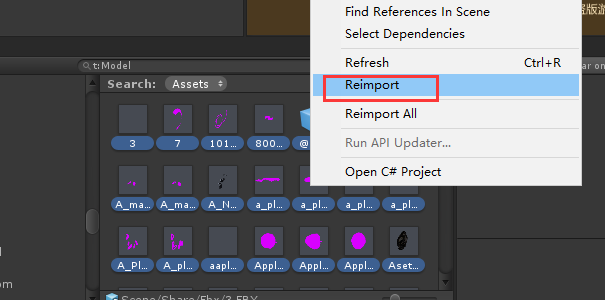
1. 检查日志输出
   1. ，游戏运行状态所有红色输出必须要处理
   2. ，游戏运行状态黄色输出的处理要看其具体含义而定，
   3. ，白/黄色输出如果高频率出现也要处理，如下情况属于高频率：
      1. 1次/秒以上
      2. 点击按钮或者其它情况
   4. ，游戏未运行状态下的黄色/红色输出：



1. 检查所有的FBX/Model是否正确导入
   1. ，必须在发布机上操作
   2. ，清理下控制台
   3. ，选中所有model

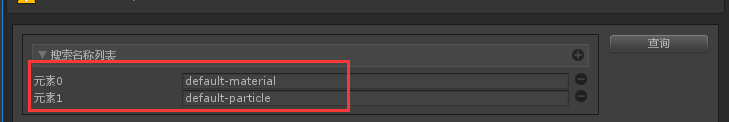


* 1. ，重新导入，等待导入完成



* 1. ，检查控制台是否有黄色/红色输出，如果有必须全部修复，优先找美术负责人处理

1. 确认是否有冗余材质:Default-Material和Default-Particle
   1. ，通过菜单【Developer/Loong/资源工具/材质工具/引用查询】打开编辑器
   2. ，输入搜索名称，选择目录【Assets/Pkg】【Assets/Scene】，点击查询：



* 1. ，如果查询后有内容，所有这些必须找到美术负责人11解决

1. 确认是否有模型勾选了导入材质球
   1. ，通过菜单【Developer/Loong/资源处理器/模型/查询】打开编辑器
   2. ，输入搜索名称，选择目录【Assets/Pkg】【Assets/Scene】，点击查询：



* 1. ，如果查询后有内容，所有这些必须找到美术负责人11解决

1. 确认版本号
   1. ，保证版本号比上一次给到用户的版本号要高
   2. ，若是第一次打包需在打包参数里设置版本号
   3. ，版本号格式:1.2.0.199
      1. 用户版本号 1.2.0

其中次版本号 2 必须比上一次高，一般+1

* + 1. 内部版本号 199

必须比上一次高，至少+10，运营提的

1. 确认场景和Prefab高低配材质是否正确

以防再出现崩溃问题，具体工具可找罗洋

1. 确认所有场景

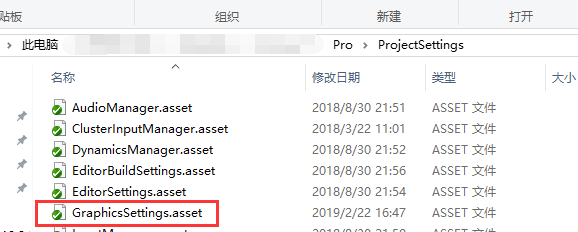
1）图形学设置是否一致

场景位置：Assets/Scene目录，逐一打开所有的场景，11确认：

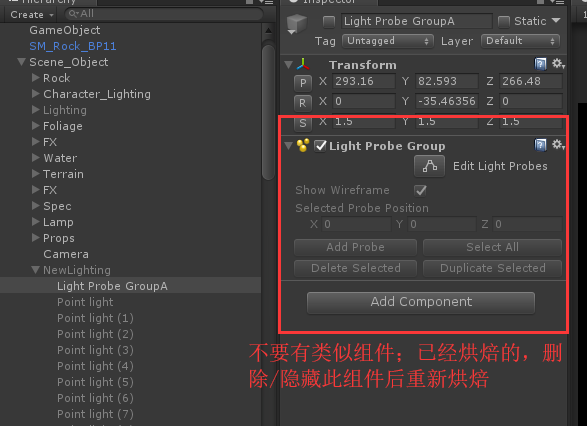
* 1. ，选择菜单【Edit/Project Setting/Graphics】
  2. ，确认图形学设置一致



* 1. ，如有修改，将以下位置文件进行备份：



2）所有场景都得检查打开时控制台是否有GI的错误,以防ysdk的包在TX模拟上的崩溃，若有按照如下方式修复：



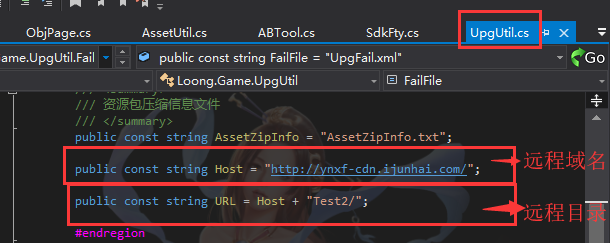
3）依次打开所有打包的场景，右键【Developer/Loong/脚本工具/丢失工具/打开当前场景中的丢失脚本定位窗口】将所有丢失脚本的组件移除：

4）是否有Prefab出现Miss

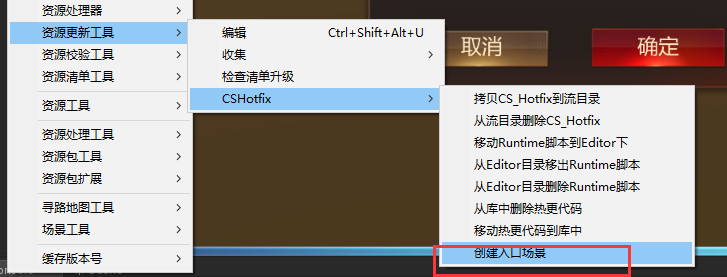
如果出现找相应美术，确定是否可以直接删除或者重新补上，表现如下

1. 确认所有图集设置是否正确(贴图设置，包名设置等),请找对应前端负责人或CTO确认
2. 确认远程目录是否正确，

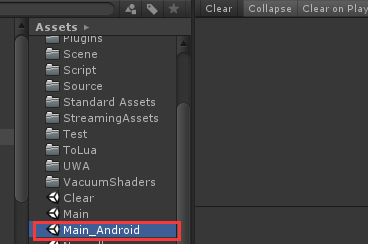
防止测试时故意修改此处而忘记还原



1. 如果安卓启用了Csharp热更，
   1. ，要确认Csharp热更库是最新的，直接找前端负责维护的研发人员
   2. ，要确认Main\_Android是最新的，可以使用工具直接创建最新场景，如下：



创建生成的场景位置如下，并将场景文件提交SVN：



1. 以上内容若出现修改,记得SVN上传
2. 发布debug包内部模拟测试，

参考【自测大纲.docx】

## 发布

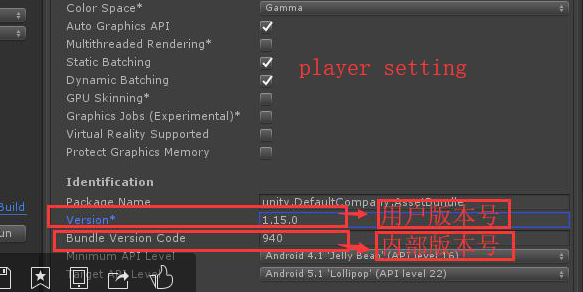
1. 如何发布，可参考【自动发布.docx】
2. 首次发布一定是勾选 Debug的版本
3. 只有在Debug版本通过测试后，才进行Relase版本的发布

## 发布后的善后检查工作

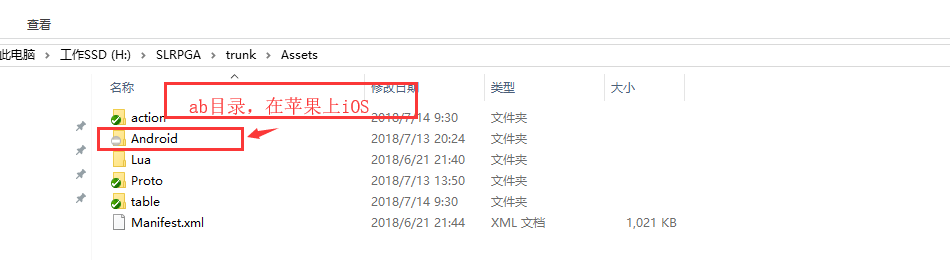
* 1. 发布机账号：所有以phantom开头的账号
  2. 当Debug和Release版本都测试无误后
  3. 将Release包进行SVN备份，并进行文档说明；SVN备份目录和备份文档单独创建仓库；或者在ftp对应稳定版目录进行备份
  4. 将版本号进行文件记录，下次正式发包版本号一定要比上次高，并确认包名和工程内的一致，包名：



工程内：



* 1. 将Assets下的所有资源进行SVN上传并注明当前版本号为0；此时要在SVN上将发布机账号之外的所有人都进行AB目录无访问权限设置（发布机账号除外，以下类似）



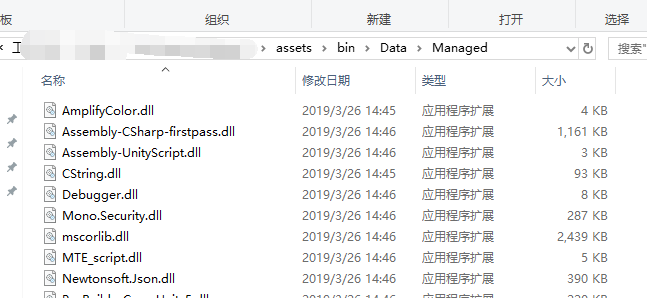
* 1. 将Assets同级目录AssetUpg下搜集的版本号为0(原始)的资源直接上传，进行无访问权限设置
  2. IOS下将Assets同级目录Xcode目录内的Xcode工程上传备份，进行无访问权限设置
  3. 检查不同渠道包的icon和logo设置是否正确
  4. ，爱奇艺：



* 1. ，君海：



* 1. Android如果启用了CS热更，对于APK，要解压查看是否有程序集Assembly-CSHarp.dll，其所在位置如下：



如果有此程序集文件，说明此次打包不正确，找前端负责人进行处理

* 1. 如果经过第三方压缩，要检查第三方是否更改包内清单文件和png.bytes文件

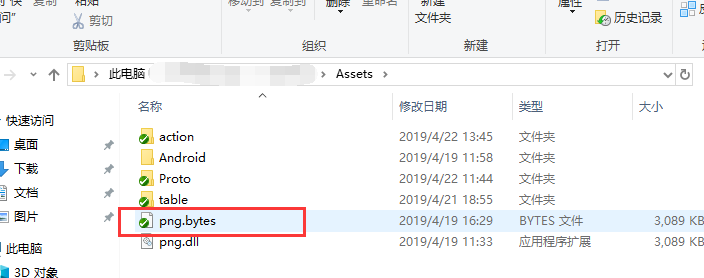
如果第三方修改了此2个文件，直接将这2个文件重新放入包内，可按照如下步骤执行，反之跳过

* 1. ，将压缩安装包后缀名改为zip
  2. ，将当前AssetUpg/Compress下打包最新版本的Manifest.xml放入包内zip包

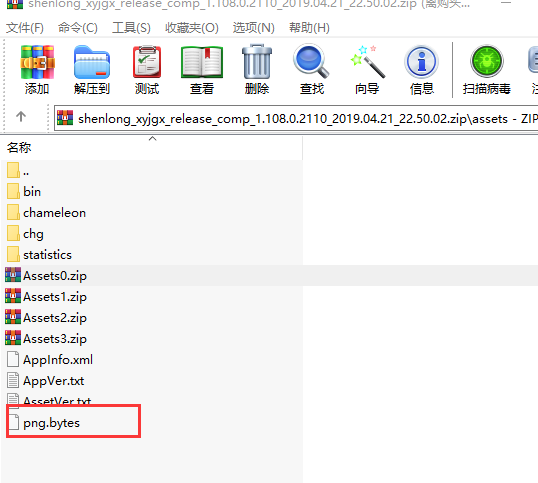
一般在Assets0.zip内

* 1. ，将最新版本的png.bytes放入包内

从这里复制：



拷贝到包内覆盖：



* 1. ，将包后缀名改回.apk, 使用AndroidMultiTool重新签名、
  2. 发布测试包，参考【发布测试包.xlsx】