使用Unity3D也有一段时间了，由于团队一直使用SVN进行版本管理，现总结一下：

(1) Unity3D的二进制资源必须加锁进行版本控制，因为它没办法merge；

(2) Unity3D自己产生的C#工程相关文件，包括VS产生或者Mono产生的，都不需要归入版本控制；

(3) 在新加入项时，Unity3D会产生一个同名的.meta文件，这个文件记录了此项一个唯一的64位的guid，这个文件必须上传，否则会出现各种缺失的异常情况（比如脚本Miss等）

(4) 对于Library文件夹，有以下几个.Asset文件需要进行版本管理：

               TimeManager

               TagManager

               QualitySettings

               ProjectSettings

               NetworkManager

               InputManager

               EditorBuildSettings

               DynamicsManager

               AudioManager

即便如此，在使用Editor进行开发时，由于需要经常改动场景的.unity 文件，而这个是二进制的，所以需要得锁且不能Merge。感觉并行开发的难度还是比较大的，Unity3D还是比较适合个人开发者和较小团队开发，对于大型团队开发，无论是从版本管理还是代码结构上，都觉得不太方便和自然。

<http://blog.csdn.net/jjiss318/article/details/7186990>