框架说明：

架构说明：

GameFramework说明：

GameFramework代码UML说明：

代码规范说明：

**一．架构说明：**

GameFramework三层架构：

1.底层：

说明：各个核心功能的实现，表现为XXManager相关的实现，

具体见GameFramework说明：**https://github.com/EllanJiang/GameFramework**

命名空间：GameFramework

是一个DLL

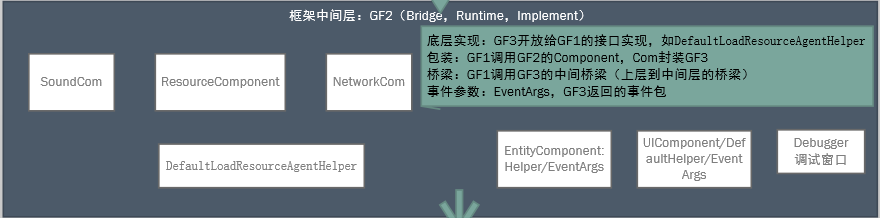


2.中间层：

对底层的封装，上层业务层不能直接调用底层XXManager，必须通过XXComponent中转调用，这里还包括对底层使用的接口的实现类，比如DefaultLoadResourceAgentHelper完成ResourceManager的最终资源加载实现。这里开放给用户修改或添加实现类

命名空间：UnityGameFramework.Runtime

目录：Assets/Scripts/GF\_Bridge



3.业务层：

GameFramework相关：通过调用XXComponent调用底层的核心功能完成业务逻辑。

放在GF\_Logic

其他业务逻辑：放在Assets/Scripts/Logic里

命名空间：Qin

目录：Assets/scripts/Logic/GF\_Logic；Assets/scripts/Logic 

**二．代码规范说明：（业务层）**

类：前缀c; cSkillManager, cCharacter

枚举：前缀e；ePlayer\_Job

结构体：前缀s；sItemInfo

Partial：代码多用Partial，尽量在一个类的管理下，避免代码之间过于零散无关联不好

管理；比如PlayerManager分为PlayerManager, PlayerManager.Net, PlayerManager.Load

命名空间：代码过多，需要按命名空间组织代码，注意层次

三．目录结构

RawRes: 原始资源目录，先按模块分，再按内容分，最后按类型分；

比如Character和Windows

Character包含character001，character002；

character001包含Model,Animation,Texture,Material等；

Windows包含LoginUI，RoleUI；

LoginUI包含：Texture，Material, Effect, Animation等

Scene包含SelectScene, GameScene

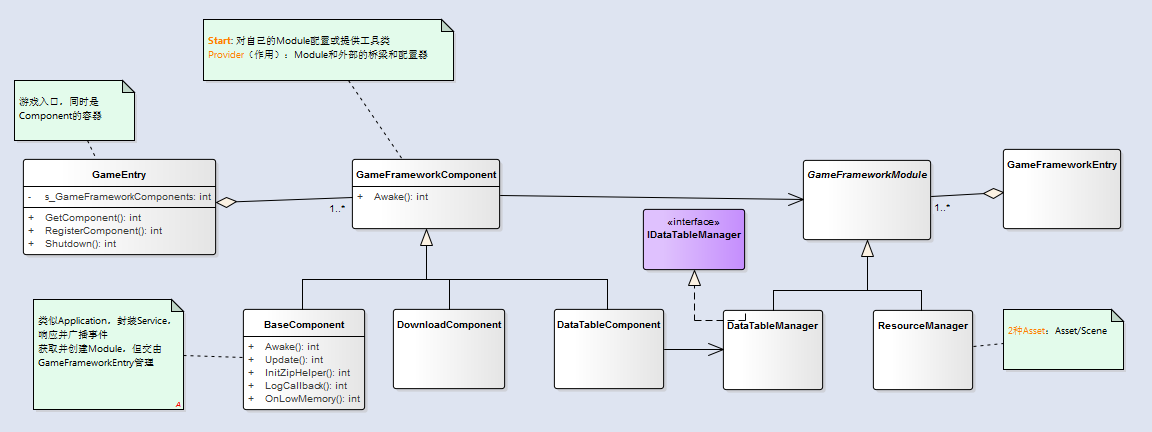
GameScene包含：Model，Texture，Material，Scene文件等

Prefab：Prefab文件

Scripts：逻辑代码；Common,Logic,GF\_Bridge

**三．GameFramework代码UML图：（放大）**

1. 主框架：



1. 资源加载时序图



3.其他：见EA图