资源管理框架

资源管理框架都实现了那些功能？

（热更新）从打包到加载，打包的话我们会把这个差异包所包含信息的这个清单文件一起上传到服务器，然后当这个客户端进行启动的时候，首先会检查资源更新，会跟服务器进行一个对比，包括版本号，名称和时间，最后如果是确定之后，再进行一个资源的MD5码的一个对比，把需要下载的资源放到下载队列里面，进行下载，下载完成之后进行一个资源的替换，再进行主游戏逻辑，

（资源管理）资源管理这方面的话，当需要加载目标资源的时候，每个资源都会有一个清单列表，包含着它所依赖的这些资源信息，首先会把依赖资源全部加载出来，同时我们对它的管理采用的是引用计数这种方式，把它的引用计数进行更新，这是加载，当它卸载的时候，会把它的引用计数--，当它的引用计数为0的时候，把它放到待删除队列里面，待删除队列它有一个容量，它的管理我们采用的是LRU的一种思想，当这个容量满的时候，我们会将前面先添加进去的进行删除掉，但是我们可能会进行再次使用，所以我们会再有一个时间进行判断，如果超出这个时间的话，我们就会进行卸载掉，如果再使用的话，再从这个待删除队列里面拿出来复用，这样就灵活一点。

下载，比对资源变更那一块，用了MD5，MD5值存在了清单文件里，每一个资源都会有一个对应的清单文件

那本地已经下载的资源的MD5值是怎么取出来的？

本地里面也会有一个清单文件，下载更新完成之后，本地的清单文件也会进行更新，更新成最新版本

本地和服务器都会各有一个清单文件

AssetBundle资源打包这一块是什么策略？

依赖打包这种形式，UI的话，我们是分模块进行打包，比如背包，商城，排行榜分别打成一个包，游戏里面这种角色模型，把它的材质，贴图打成一个包，特效再单独打成一个包，音效单独打成一个包，场景的话不是很大，就单独打成了一个包。

资源卸载一般会在场景切换的时候会进行资源卸载，当场景切换的时候，待卸载队列也会进行卸载，场景切换的话，原来缓存的也会进行卸载

AssetBundle在什么时候卸载？

加载完之后就卸载了，如果AssetBundle里面有多个资源，只用了里面一个两个资源，也是加载完之后直接卸载，大多数跟打包策略有关，打包策略也是大程度减少了AB包加载完，再进行卸载反复重复的这种几率，也是提高了性能，从打包策略上配合一下

卸载Unload传的参数为false

切场景卸载资源用的用的Unload（true）进行卸载的

AssetBundle打包可能会出现资源冗余，在打包之前，看这个资源他被那些资源依赖了，如果这个计数>=2,我们就拿出来单独打包，防止资源冗余。

资源之间的依赖关系处理，也就是循环遍历处理，判断他那个依赖计数，如果他的依赖计数>=2的时候就进行打公共包

AssetBundle压缩方式？

LZ4，LZMA

LZMA压缩率可能大一点，整体如果是LZMA的话它的包体可能会小一点，LZ4包体会大一点，LZMA是一个整体打包，如果我们只需要AB包中的一个资源的话，就要进行整体解压，LZ4的话，是进行分块的打包的，进行资源获取的时候可以找到对应的块把它解压出来，虽然说LZ4的压缩率没有LZMA的高，但是可以单独解压一个资源而不用解压全部，而且LZ4还是一种无损压缩算法，压缩速度为每核心40MB/s，所以我们当时用的是LZ4



