

Ogre 文件系统分析

盛崇山

<http://antsam.blogone.net>

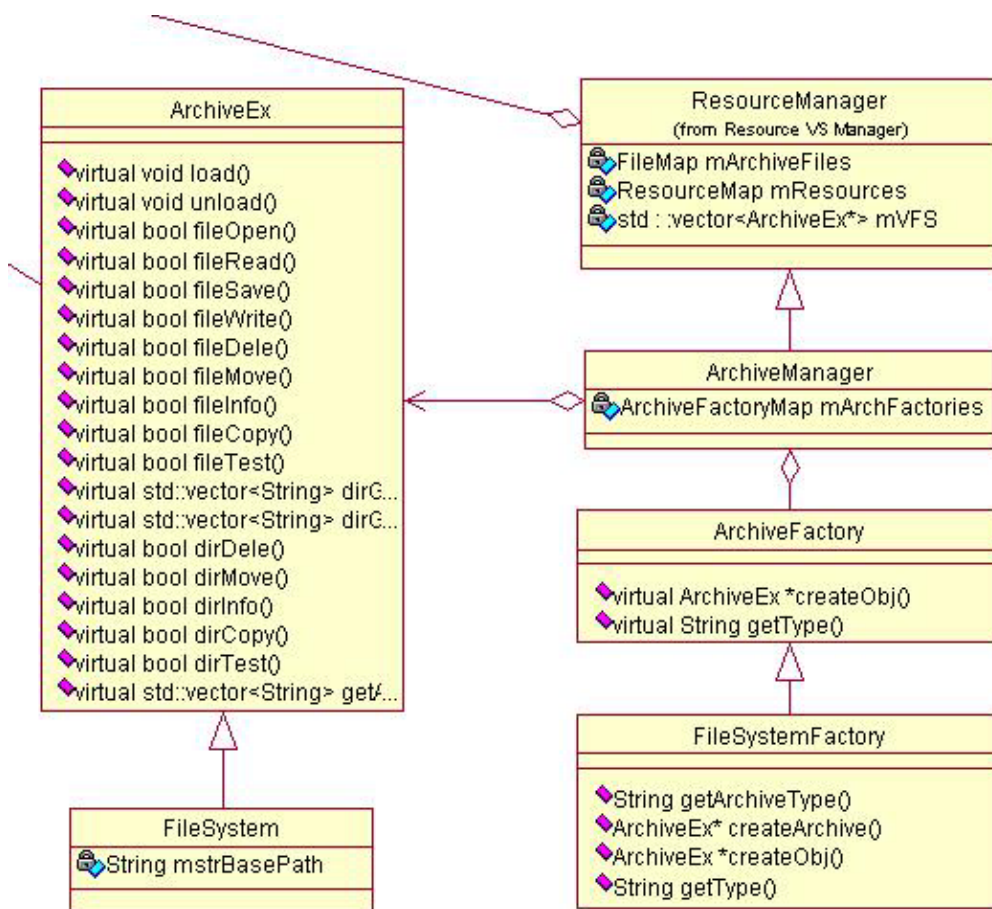
AntsamCGD@hotmail.com

- 为什么要定义文件系统？
- 可以定义那些文件系统？

从一般的角度看好像没有必要定义文件系统，但是游戏的数据为了保密（特别是那些资源文件：游戏图片、游戏音乐、shader 脚本等），可能把所有的文件打包在一个文件中，那么我们读取这个文件的时候，可以把它看成一个文件系统，因为其格式固定，也有文件目录，文件数据等（例如^_^）。

从功能上我们把文件系统分为：自定义文件系统，这类文件系统一般用于读取游戏中的大包文件；另外一种则是普通的文件系统，只是包装了操作系统上的读写功能，用于读写一般文件。下面从代码的角度进一步分析文件系统。

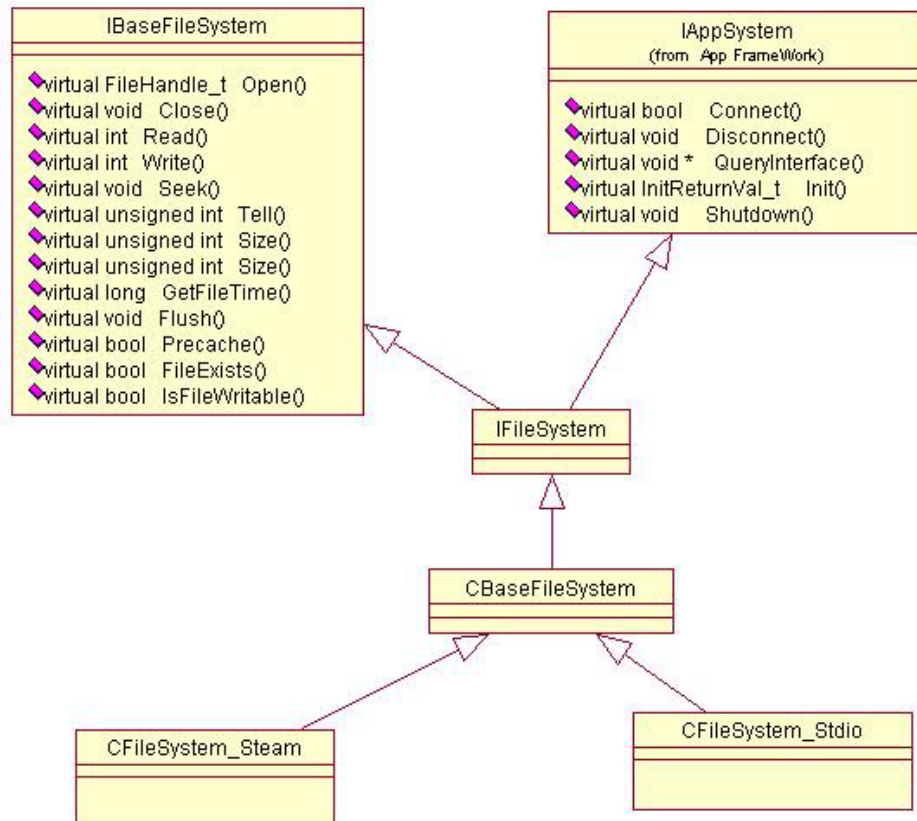
现在让我们看一下文件系统相关类的 UML 图：



图一：文件系统相关类 UML 图

在图中 ,ArchiveEx 是一个文件系统的抽象父类 ,包括了对文件的各种操作 ,ArchiveEx 的子类 File System 是用于读取一般文件的文件系统 ,而且在 Ogre 中文件系统以插件的形式存在。其实在 Ogre 中用到了很多 Factory 设计模式 ,但是有时候比较难区分是 Abstract Factory 还是 Factory Method ,还是两者之间的结合。

关于文件系统的设计我们还可以参考一下 Half Life 的设计 ,下面这幅图是 Half Life2 中文件系统的层次结构 :



可以说只是类的命名方式上的不同。