

OGRE 分析之文件系统（四）

Mythma

<http://www.cppblog.com/mythma>

Email: mythma@163.com

四、Scripts

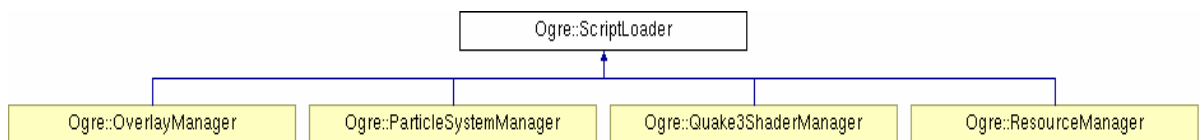
为了便于设置，OGRE 的许多方面都使用了脚本（Scripts）。Scripts 只是简单的文本文件，编辑修改十分方便。

OGRE 中可以作为脚本的主要有如下几种（配置文件也可以看作是一种 Scripts）：

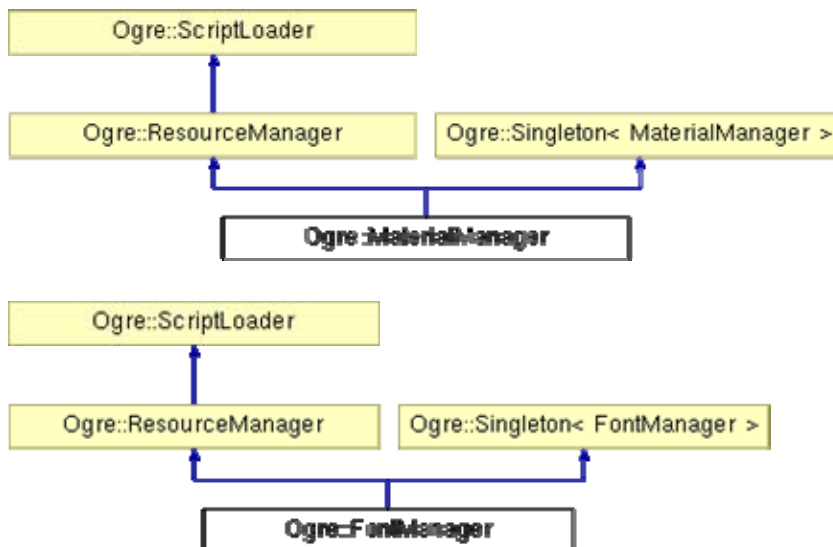
- Material Scripts
- Particle Scripts
- Overlay Scripts
- Font Definition Scripts

关于各种脚本的格式，OGRE 手册有详细的介绍。

对脚本的操作，OGRE 有一个 `Ogre::ScriptLoader` 抽象类，其子类如下：



由于 Material 和 Font 属于资源的一种，所以操作 Material Scripts 和 Font Definition Scripts 的功能使用由 `ResourceManager` 的子类 `Ogre::MaterialManager` 和 `Ogre::FontManager` 完成的，如下：

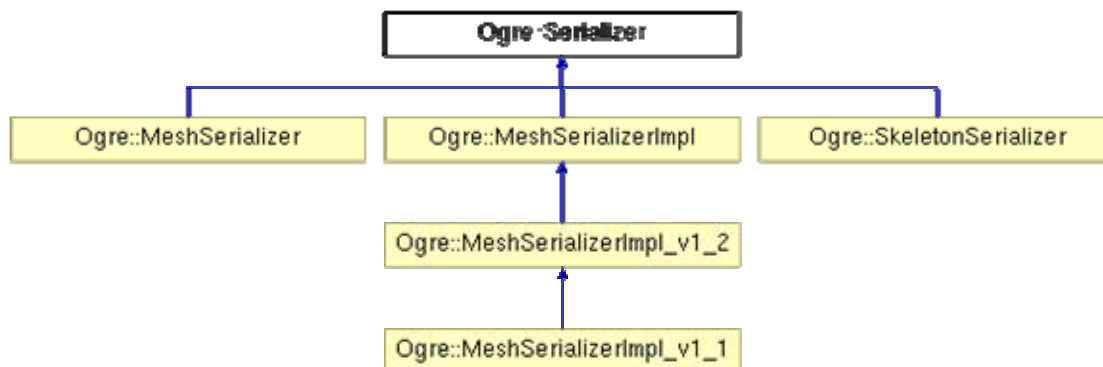


至于 OGRE 是如何读取各种脚本的，在此不作分析。

五、序列化

前面分析的 `Archive` 和 `DataStream` 主要作为通用读数据的工具。对于二进制文件的读写，OGRE 提供了专门的序列化类——`Ogre::Serializer`。

此类文件，OGRE 主要有.mesh 和.skeleton 两种，对这两种文件的序列化，OGRE 提供如下类：



可见，针对 mesh 文件的不同版本，OGRE 提供了不同的内部实现。MeshSerializer 和 MeshSerializerImpl 之间关系可看成是**桥接模式**。只不过 MeshSerializer 中存放的是不同版本的 MeshSerializerImpl 表：

```
typedef std::map<String, MeshSerializerImpl* > MeshSerializerImplMap;
MeshSerializerImplMap mImplementations;
```

对 Material，OGRE 是看成是脚本资源来管理，即为文本文件。对其读写类 OGRE 用了个序列化的名字——Ogre::MaterialSerializer，但是并没有从 Ogre::Serializer 继承，显得有点不伦不类。

六、log 管理

日志可以用来输出程序的运行状态情况。简单的日志操作就是书写文件，而 OGRE 的日志管理却不仅仅如此。OGRE 可以把信息输出到几个 log 文件中，甚至可以重新定向到 GUI。

OGRE 与 log 相关的类有三个：

Ogre::Log：负责文件的读写

Ogre::LogManager：Log 管理。它维护了一个 Log 的表和 LogListener 的表。

Ogre::LogListener：用于 Log 输出的重定向。

草稿

2005.12.03

