**Core Engine自动化测试简要介绍**

**[文件夹结构介绍]**

1. arbiter目录是我们写的测试代码所在目录,后续新增测试类也放到此处
2. CoreServices\EventStructures\PlatformServices和"\*\*API"几个目录是Thrift接口文件生成的目录,不用理会
3. thrift目录是python的Thrift依赖库目录,不用理会
4. configuration.cfg是配置文件目录。测试程序运行时的参数或默认配置都是从这里面读取。这个比较重要，请慎重改动
5. main.py是测试程序的主函数模块，通过执行它运行测试程序
6. core-test-debug.log、core-test-info.log是测试程序执行后生成的日志文件。

**[该程序会测试什么]**

1. 通过Thrift接口向Arbiter添加一个名为"MegaPixel"，host是10.101.0.181(可通过配置文件进行配置)的AMTK类型的设备，并检测是否正确添加到数据库和DS
2. 视频直播是否正确，通过调用beginStreamSession查看DS返回的liveview的URL是否正确
3. 设置视频存储大小是否正确，通过setChunkSize设置视频大小
4. 视频存储是否正确
5. 图片存储是否正确
6. 视频存储事件是否正常触发
7. 更新设备(默认将设备的host由10.101.0.181改为10.101.0.182)。查看数据库中更新是否正确，DS是否更新正确，并查看更新后设备的一些功能是否能够正常实现。
8. 删除设备，并查看数据库和DS是否删除正确

**[执行测试前的必备条件]**

1. 测试程序和CoreEngine安装在同一台服务器(否则需要CoreEngine服务器开放MySQL的远程访问权限)
2. 服务器安装了Python环境
3. 服务器安装了python MySQLdb模块

--检查方法：

运行python,在终端输入import MySQLdb

如果不报错，说明安装了该模块，否则，退出Python终端，执行sudo apt-get install python-mysqldb

1. CoreEngine必须正确安装，并且Arbiter、MySQL和AMTK的DeviceServer正在运行

注：python 采用2.7.3版本

**[如何运行测试程序]**

1. 将python-unittest目录拷贝到服务器上任意目录下
2. 执行 python main.py

详细说明请看: [Core Engine自动化测试使用文档](Core%20Engine自动化测试使用文档.docx)