

转 Flash 平台音视频直播的实现

2013年10月15日 16:00:23 阅读数：7124

开发环境及工具准备：

Adobe Flex Builder 3

FlashMediaServer 2 及以上（简称：FMS 本人有一个FMS2破解版，需要的话联系我）

技能准备：

ActionScript 3 （简称：AS，本文中代码为AS）

本功能的实现方式基于FMS为主，ActionScript为辅，双方交互的原则，示意图如下：

□

描述：教师端采集本地摄像头视频及MIC音频通过AS发布到FMS，学生端通过AS连接FMS服务器或者教师传输的音视频数据进行播放。至此为止，即可实现教师的音视频数据共享（即直播效果）。

具体实现（教师端）：

STEP1:在Flex Application 的creationComplete事件调用init方法

```
[plain]
1. private var conn:NetConnection = new NetConnection();//用来连接FMS
2. private var stream:NetStream = null;//盛放音视频数据
3. //FMS连接串（例如：rtmp://192.168.1.101/fmsapp）
4. //其中需要在FMS_HOME/applications/新建一个fmsapp的文件夹
5. private var fms:String = "";
6.
7. private function init():void {
8.     //轨道通信，必须要设置
9.     conn.objectEncoding = ObjectEncoding.AMF0 ;
10.    //共享对象轨道通信设置
11.    SharedObject.defaultObjectEncoding = ObjectEncoding.AMF0;
12.    //连接FMS服务器
13.    conn.connect(fms) ;
14.    //连接FMS结果事件处理器
15.    conn.addEventListener(NetStatusEvent.NET_STATUS,connectFMS) ;
16. }
```

STEP2:连接FMS成功后，发布音视频数据到FMS服务端

```
[plain]
1. private function connectFMS(e:NetStatusEvent):void {
2.     var result:String = e.info.code ;//连接FMS的返回代码
3.     switch(result) {
4.         case "NetConnection.Connect.Success"://连接FMS成功
5.             playStream();//开发传送音视频数据
6.             break;
7.         case "NetStream.Play.StreamNotFound":
8.             Alert.show( "视频流没有找到");
9.             break;
10.        default :
11.            Alert.show( "连接服务器失败,检查网络环境");
12.            break ;
13.    }
14. }
15. public var cam1:Camera = null;//摄像头对象
16. private var mir:Microphone = null;//麦克风对象
17. private function playStream():void {
18.     stream = new NetStream(conn);//创建于FMS连接相关的音视频数据流对象
19.     cam1 = Camera.getCamera("0");//获得默认摄像头
20.     cam1.setMode(320,240,15); //经试验：320*240效果较佳而占用网络流量较小 15为keyFrame
21.     cam1.setQuality(100 * 1000,90); //100*1000为上传带宽 90为图像质量
22.     mir = Microphone.getMicrophone();//获得麦克对象
23.     mir.setUseEchoSuppression(true);//抑制回声
24.     stream.attachAudio(mir);//将音频加入到流中
25.     stream.attachCamera(cam1);//将视频加入到流中
26.     vd.attachCamera(cam1);//在Flex Application中放一个VideoDisplay对象，用来本地观看图像
27.     stream.publish("conf","live");//将音视频数据发布到FMS服务器上，流名称为：conf 发布类型为：live（实时）
28. }
```

至此：教师已将音视频数据发布到FMS。

需要注意的点：摄像头对象的分辨率设置，这样会直接影响到图像的清晰度和传送数据量的大小。

接下来将由学生端对此数据进行读取

STEP1:如教师端STEP1进行相关设置

STEP2:成功连接FMS服务器后，进行音视频数据播放（学生端playStream方法的实现）

```
[plain]
1.  stream = new NetStream(conn); //学生端的流对象 conn为NetConnection对象
2.  //学生端的缓冲时间。此参数很重要，很多朋友反应：播放视频不流畅。
3.  //要明确一个观点：没有绝对的实时直播，所谓的直播其实就是在不断的播放缓冲区的数据
4.  stream.bufferTime = 15;
5.  //按照道理来说这里应该是检测缓冲区的代码
6.  //但本程序为展示直播的核心将此代码略掉，有兴趣的朋友可以问我
7.  v = new Video();
8.  v.attachNetStream(stream); //将音视频流附加到video对象中
9.  v.width = vd.width;
10. v.height = vd.height;
11. v.smoothing = true;
12. vd.addChildAt(v,1); //VideoDisplay通过对Video的添加来播放音视频流
13. stream.play("conf"); //播放教师发布的conf流
```

至此，学生将实现音视频流的播放。

需要注意的点：缓冲区的设置，实际应用中应根据实际客户带宽来设置缓冲区大小

大家最好找个局域网环境测试下，最好不要在本机测试，这会隐藏很多问题，有什么疑问大家尽管留言！

原文地址：<http://mhqawjh.iteye.com/blog/726913>

文章标签：[flash](#) [视频](#) [直播](#) [actionsript](#) [摄像头](#)

个人分类：[Flash相关](#)

此PDF由spygg生成,请尊重原作者版权!!!

我的邮箱:liushidc@163.com