



PLYmedia Inc.  
AS3 Flash Player Integration Document



## Purpose

The goal of this document is to outline and explain the process for implementing Flash Player AS3 integration with PLYmedia's PLYplayer, as well as provide step-by-step instructions for this procedure.

## Target Audience

IT personnel, webmasters and Flash developers within companies that utilize Flash player technology and who wish to enable PLYmedia's services in streaming video to the player on their websites.

## Table of Contents

|   |   |
|---|---|
| Purpose .....   | 2 |
| Target Audience.....  | 2 |
| Table of Contents.....  | 2 |
| Overview.....   | 3 |
| Step-by-Step instructions: .....  | 3 |
| 1. Define 2 LocalConnection instances: .....                                      | 3 |
| 2. Load PLYmedia's transparent player and instantiate the 2 LocalConnections:.... | 4 |
| 3. Define a function name GotConn: .....  | 4 |
| 4. Define the utility functions used above: .....                                 | 5 |
| Sending video time .....  | 6 |
| Important reminder:.....  | 6 |
| Additional steps.....   | 6 |
| Sample Application .....  | 7 |



## Overview

PLYmedia's player is a transparent Flash Player created and loaded for the purpose of displaying layers (PLYs) on top of a video. The integration process creates a link between the Flash Player on the hosting site and PLYmedia's Flash Player, and requires the following set of actions:

- Defining 2 Local Connection instances, one for input another for output, and a connection id.
- Loading PLYmedia's transparent player (plyPlayer.swf) on top of your player
- Instantiate the 2 Local Connections
- Defining a function named GotConn() . This function will be called when PLYmedia's player is loaded and ready to communicate.
- In GonConn, first setting up the parameters, then calling resize and passing PLYmedia your flv url.
- Making sure to call the updatePLYtime function at least 4 times a second and passing your current video's time.
- Defining utility functions for communicating with PLYmedia's player, for updating time, resizing, setting parameters and passing the flv path.

PLYmedia's current player is written in AS2, thus, to allow the two instances to communicate we will be using LocalConnection.

## Step-by-Step instructions:

### 1. Define 2 LocalConnection instances:

One is for receiving method calls from PLYmedia's player, another is for sending them. Define the variables in the class handling the whole PLYmedia loading.

```
private var inLC:LocalConnection;  
private var outLC:LocalConnection;  
private var connectionNum:String = String(Math.floor(Math.random() *  
10000000000)); // This is just a suggestion, it can be anything you want.
```



## 2. Load PLYmedia's transparent player and instantiate the 2 LocalConnections:

In this segment we will load PLYmedia's player from the shown url.

Note that we are passing the cn variable via query string. We will be using this variable to give our local connection a unique name. This is especially important if you have more than one player on one page. Please do not change the local connection's name.

```
private function loadPly():void{
    var plyURL:String =
"http://www.bubbleply.com/plyPlayer.swf?cn="+connectionNum;
    plyLoader = new Loader();
    plyLoader.load(new URLRequest(plyURL));

    outLC = new LocalConnection();
    outLC.allowDomain("");
    outLC.addEventListener(StatusEvent.STATUS, onPlyStatus);
    inLC = new LocalConnection();
    inLC.allowDomain("");
    inLC.client = this;
    inLC.connect("_Incoming" + connectionNum);
}
```

## 3. Define a function name GotConn:

It is very important that you do not leave out this part. This is the function PLYmedia's player calls, once it has been loaded. It indicates that the LocalConnection was established successfully.

In this function you will execute the following:

- use `plyPassVars()`, to pass the parameter `pType='mode4'` to PLYmedia. You can use this method to pass any other parameters to PLYmedia.
- use `plyResize()`, to adjust PLYmedia's player's size to suit your video screen. You must call this method each time you go to and from FullScreen as well. Please make sure you are passing the video's current size and not the whole player's.
- use `plySetVideoPath()`, to pass your flv url to PLYmedia's player.



```
public function GotConn() {
    trace('[as3] GotConn!!!!');
    plyPassVars();
    plyResize();
    plySetVideoPath('http://www.domain.com/video/youcurrentflvpath.flv');
}
```

#### 4. Define the utility functions used above:

```
private function updatePLYtime(time:Number) {
    var str:String = "_ConnectAS2" + connectionNum;
    outLC.send(str, "updateTime", time);
}

private function plySetVideoPath(path:String):void {
    trace("Plyresize");
    var str:String = "_ConnectAS2" + connectionNum;
    outLC.send(str, "sitePath", path);
}

private function plyResize():void{
    trace("Plyresize");
    var str:String = "_ConnectAS2" + connectionNum;
    outLC.send(str, "reSize", videoWidth, videoHeight);
}

private function plyPassVars():void{
    trace("plyPassVars ");
    var str:String = "_ConnectAS2" + connectionNum;
    var varsObj:Object = {pType:'mode4'}
    outLC.send(str, "setPlayerRules",varsObj);
}
```



## Sending video time

- In order to synchronize your video player with plyPlayer, you must update plyPlayer's time at least four times per second (interval rate of 250).
- The time should be in seconds and it is the streamNS.time.
- In the previous paragraph you defined a function updatePLYtime(time); utilize it.
- It is recommended you call this function within the function that handles your player's progress bar.

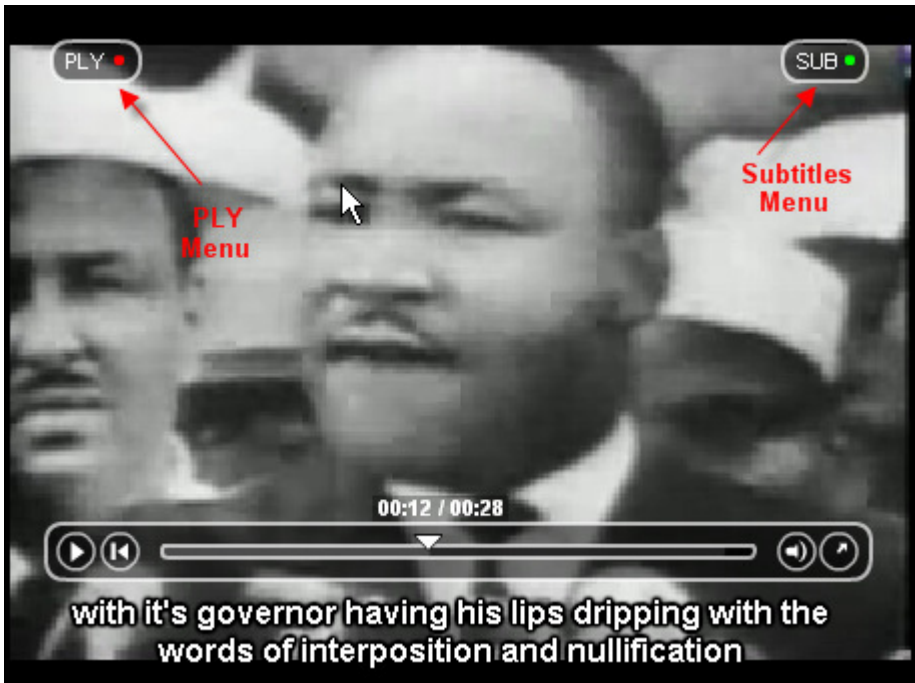
## Important reminder:

On any change of your video size (e.g. full-screen mode), it is important to ensure PLYmedia's player resizes with your player. Also make sure to update its x and y coordinates if your video holder has changed location.

## Additional steps

- Best Practice: It's best to set your video player's frame rate at 24 Frames per Second. Any FPS under 12 may distort time accuracy and animation effects.
- Your video player must have the following line of code embedded:  
"Security.allowDomain("http://www.bubbleply.com/") in order to allow uninterrupted communication between both players.
- After completing all these steps in your test environment, please notify your contact person within PLYmedia to have your services enabled on the server side.

Once you have successfully completed the above steps, and after the relevant services have been opened for you by the PLYmedia team, you should be able to view the video with PLYmedia's menus.



Examples of PLYmedia menus you might have available

## Sample Application

The zip file under the following link contains sample applications, displaying the best way for BubblePLY integration: [http://www.bubbleply.com/sample\\_as3\\_caller.zip](http://www.bubbleply.com/sample_as3_caller.zip)

In order to add the ability to create, save & modify PLYs on your site, and with any integration related questions you might have, please contact our technical department:

PLYmedia Inc.

[Support@PLYmedia.com](mailto:Support@PLYmedia.com)