



PLYmedia Inc.
Windows Media Player
Integration Document



Purpose

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

Target Audience

IT personnel, webmasters and Video Player developers, within companies utilizing Windows Media technology, wishing to enable PLYmedia services for the streaming video player on their websites.

Table of Contents

Purpose	1
Target Audience	2
Table of Contents	2
Overview	3
Parameters	4
Samples:	4
1. Sample call for both players:	4
2. Controls.js file, configuring connectivity for Media Player functions	5



Overview

The purpose of this document is to establish an integration between PLYmedia's Flash based PLY Player and Partner's Windows Media based Player.

PLYmedia's Flash based PLY Player allows adding information layers over streaming videos:

- Subtitles
- User generated bubbles
- Ads
- Info type bubbles

PLYmedia's PLY player enables two basic types of interaction for the end users:

1. Viewing
2. Creation

There is a third option which enables professional content (like subtitles) to be added using additional creation tools supplied by PLYmedia.

The following paragraphs detail instructions and required parameters for a successful integration.

Theory of Operation

Once content site operators decide to add PLYmedia functionality to its videos, they will create a transparent div calling the PLY player. This player will be hosted either on the Partner or PLYmedia's Servers, appearing on top of the Partner's Media Player.

When Calling the PLY Player, you will pass the parameters below + a unique movie URL.

During the viewing of the movie you will pass, to the PLY Player (using Jscript as detailed later), the current timing of the movie at a rate of at least 4 times per second.

The PLY Player will then analyze the passed parameters and will display the applicable PLY (layer).

When an end user goes into Edit Mode (**only applicable for the user generated BubblePLY service**) the PLY Player will expose it's own movie controls and pass movie commands (such as Play, Pause, Seek; using Jscript as detailed below) to your Media Player; in this case your Player controls should be disabled.



Parameters

You should pass the following parameters to the PLY Player:

pType	wmv
movieURL	A unique movie identifier (to be agreed on)

The following "Samples" section demonstrates the required syntax and context.

Samples:

1. Sample call for both players:

```
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML//EN">
<html>
<head>
<meta http-equiv="Content-Type"
content="text/html; charset=iso-8859-1">
<title>PLYmedia WMV</title>
<script src=" http://www.bubbleply.com/Script/controls.js"
type="text/javascript"></script>
<script src=" http://www.bubbleply.com/Script/swfobject.js"
type="text/javascript"></script>
</head>
<body>
<div id="pwmv" style="z-index:0; width:450px; height:420px;position:
absolute;">
<object id="Player" height="455" width="450"
classid="CLSID:6BF52A52-394A-11d3-B153-00C04F79FAA6">
<param name="windowlessVideo" value="true" />
<param name="URL" value="mms://YourSiteDomain/YourVideoId.wmv"/>
<param name="SendPlayStateChangeEvents" value="true" />
<param name="AutoStart" value="true" />
<param name="autoSize" value="false" />
<param name="displaySize" value="4" />
<param name="stretchToFit" value="true" />
<param name="playCount" value="1" />
<param name="AutoRewind" value="true" />
<param name="uiMode" value="full" />
```



```

    </object>
  </div>
<div id="pDiv" style=" z-index:100; width:450px; height:392px;
border:solid 1px red; position: relative;">
<div id="flashcontent"></div>

<script type="text/javascript">
var so = new
SWFObject("http://www.bubbleply.com/plyPlayer.swf?pType=wmv&moviePath=mms://
YourSiteDomain/YourVideoId.wmv", "plyPlayer", "450", "392", "8");
so.addParam("quality", "high");
so.addParam("wmode", "transparent");
so.addParam("allowScriptAccess", "always");
so.write("flashcontent")
</script>
  <script type="text/javascript">
    var inter = setInterval("timer()" , 50);
  </script>
</div>
</body>
</html>

```

2. Controls.js file, configuring connectivity for Media Player functions

```

// JavaScript Document

function checkDuration(str){
    dur = window.Player.currentMedia.duration;
    return dur;
}
function timer(){
    try
    {
        getMovieName("plyPlayer").updateFlashTime(document["Player"].control
s.currentPosition);
    }
    catch(err)
    {
        //alert(err.description);
    }
}
function getMovieName(movieName) {
    if (navigator.appName.indexOf("Microsoft") != -1) {

```



```

        return window[movieName]
    }
    else {
        return document[movieName]
    }
}

var WMState = new Array();
WMState[0] = "Undefined";
WMState[1] = "Stopped";
WMState[2] = "Paused";
WMState[3] = "Playing";
WMState[4] = "Scan Forward";
WMState[5] = "Scan Reverse";
WMState[6] = "Buffering";
WMState[7] = "Waiting";
WMState[8] = "Media Ended";
WMState[9] = "Transitioning";
WMState[10] = "Ready";
WMState[11] = "Reconnecting";
function checkStatus(){
    x = window.aspnetForm.Player.playState;
    return WMState[x];
}
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
function Play() {
    //x = document["Player"].playState;
    //if (x == 2) {
        document["Player"].controls.play();
    //} else {
        //    document["Player"].URL =
"http://www.nyhoff.net/swimcity/Crawl-SwimcityMediaCentre-
AlexPopovTraining.mpg";
    //}
}
function Pause() {
    //alert("Player.playState : "+document["Player"].Time);
    document["Player"].controls.pause();
}
function Stop() {
    document["Player"].controls.stop();
}
function MuteMe() {
    x = document["Player"].settings.mute;
    if (x == 0) {
        document["Player"].settings.mute = "1";
    } else {
        document["Player"].settings.mute = "0";
    }
}
function UnMuteMe() {
    document["Player"].settings.mute = "0";
}
function VolumeUp() {
    X = document["Player"].settings.volume;
    document["Player"].settings.volume = X+10;
}
}

```



```
function VolumeDown() {
    X = document["Player"].settings.volume;
    document["Player"].settings.volume = X-10;
}
function setMovieVolume(val) {
    document["Player"].settings.volume = val;
}
function Seek(t) {
    document["Player"].controls.currentPosition=t;
}
```

Note:

The reference in the javascript is depended on your pages structure.

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