#

Combined Beacon Media Player Sample Integration

HTML5 Standalone Video Player & JW Player 5

HTML5 VIDEO PLAYER INTEGRATION

Introduction

This document describes steps for integrating HTML5 Video Player with Nielsen's Combined Beacon library.

For "**Standalone Video Player**" approach/example, *please refer to page 2 of this document.*

For "**JW Video Player 5 with HTML5**" approach/example, *please refer to page 10 of this document.*

ggjwxxx.js (where xxx is the version number) is our Example Library for HTML5 Video Player – simply including that typically automates data capture for core events like LoadVideo, Play and Pause.

Prerequisites and Assumptions

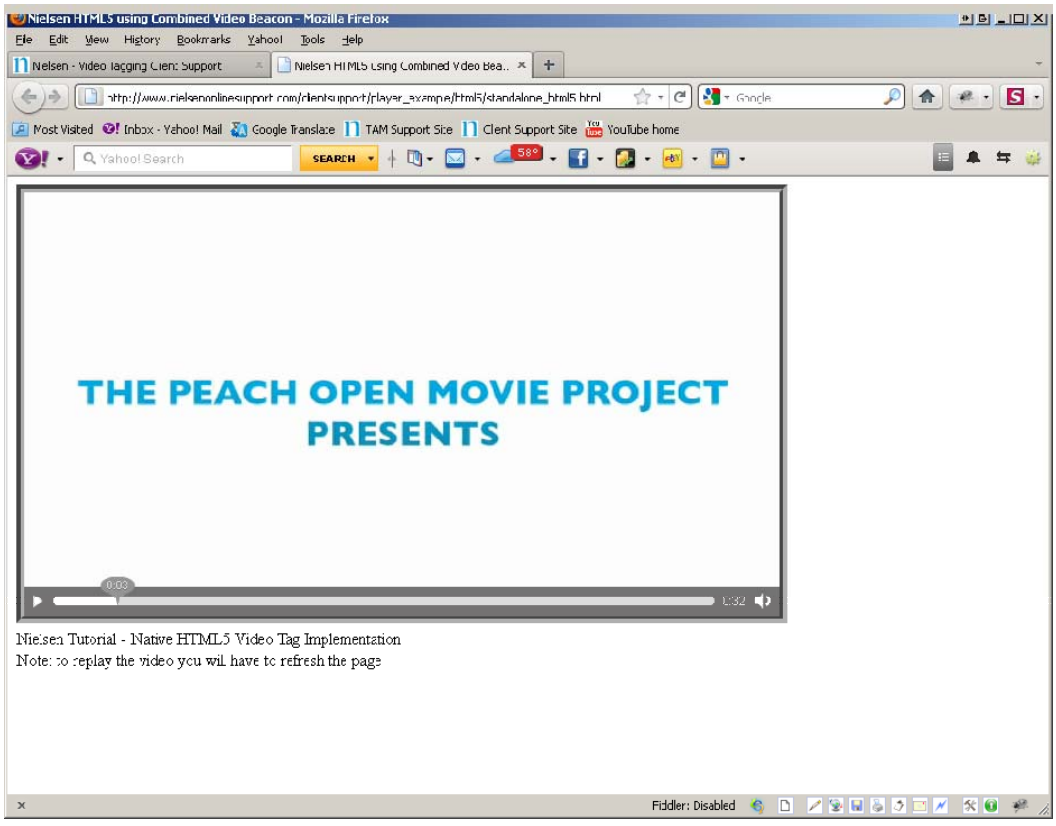
Knowledge of JavaScript and HTML5 Video Player.

The document "Combined Beacon JavaScript API Integration" is prerequisite reading.

Sample Link

The following is an example link of HTML5 Implementation:

http://www.nielsenonlinesupport.com/clientsupport/player_example/html5/standalone_html5.html



Implementation

***NOTE* "nielsenonlinesupport.com" is not a production environment. Please COPY the example code and CHANGE the src setting in step one below to your local/staging/production environment.**

The following steps are required:

1. Include the HTML5 Video Player Example Library like this:

```
<script type="text/javascript"
    src="http://www.nielsenonlinesupport.com/clientsupport/jwplayer/ggCom_embed.js">
</script>
```

2. Initialize clientid and other parameters like this (these values will be supplied to you by Nielsen):

```
<script type="text/javascript">
        var _nolggGlobalParams = {
        clientid: " my Nielsen assigned client id ",
        vcid: " my Nielsen assigned vc id ",
        cisuffix: "my Nielsen beacon type ",
        sfcode: "Nielsen assigned data node ",
        prod: " Nielsen assigned product code "
};
</script>
```

3. Include and initialize Combined Beacon Javascript library for HTML5:

```
<script type="text/javascript"
    src=" http://secure-XX.imrworldwide.com/novms/js/2/ggcmbXXX.js">
</script>
```

Note 1: "XX" is the Nielsen supplied country code for your region. For example, it is "us" for customers in USA.
Note 2: "XXX" is the latest version number of the general JavaScript beacon.

4. Complete the initialization by including the following lines:

```
<script>
   var gg1 = new gg();

   var uid = 0; //provided by Nielsen

   var oldFlashDetect = false; // no longer used (dummy placeholder for legecy implementations)

   var detectBrowser = true; //optional -- used to disable window object call for non-browser use

   gg1.ggInitialize(_nolggGlobalParams, uid, oldFlashDetect, detectBrowser);

</script>
```

5. Once you have filled in all the required information, it will look similar to the example below:

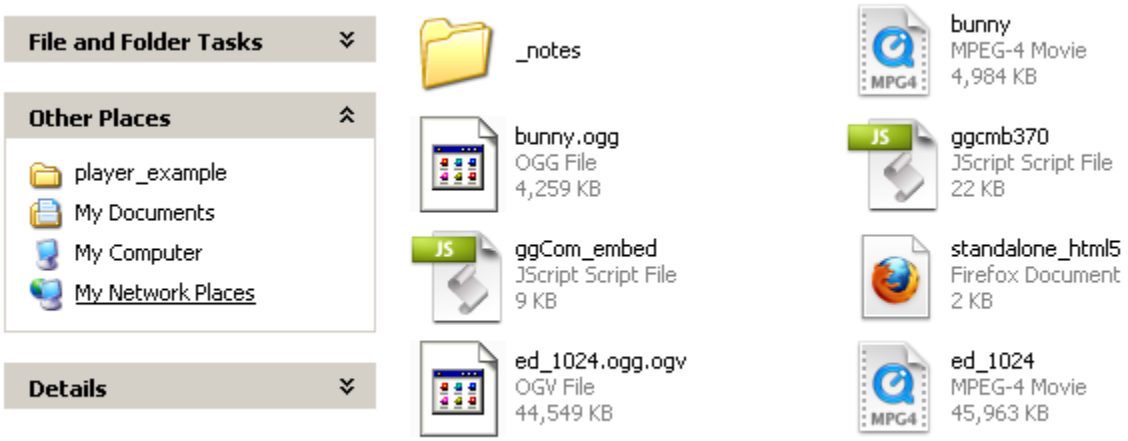
```
<script type="text/javascript" src="ggcmb370.js"></script>

<script type="text/javascript" src="ggCom_embed.js"></script>
<script type="text/javascript">
        var _nolggGlobalParams = {

                clientid:"us-502202", // provided by Nielsen
                vcid : "c09",
                sfcode : "us",
                cisuffix : "gg",
                prod : "sc,iag"
        };

         var gg1 = new gg();
         var uid = 0; //provided by Nielsen
         var oldFlashDetect = false;
         var detectBrowser = true;
         gg1.ggInitialize(_nolggGlobalParams, uid, oldFlashDetect, detectBrowser);
</script>
```

Tutorial & Example (Standalone Video Player)

In this section you will find a basic example of the implementation section detailed previously, plus how to pass video meta-data to the API. Please amend references to http://secure-us.imrworldwide.com to your local Nielsen collection node.

***NOTE* "nielsenonlinesupport.com" is not a production environment. Please COPY the example code and CHANGE the location settings to your local/staging/production environment.**

TUTORIAL FILES TO DOWNLOAD→
http://www.nielsenonlinesupport.com/clientsupport/player_example/html5/html5.zip



Please extract the zip file. Files you will be working with are:

standalone_html5.html – your location URL

bunny.mp4, bunny.ogg, ed_1024.mp4, ed_1024.ogg – Video Footage

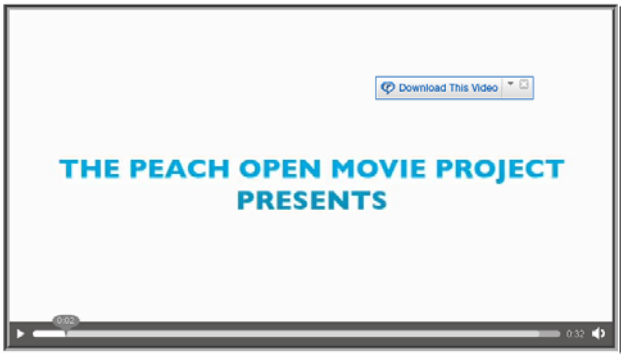
ggcmb370.js – Nielsen JavaScript / Beacon

ggCom_embed – Bridge / Beacon Handler

***Note* You may find the latest "Beacon" & "Bridge" files under:**

http://www.nielsenonlinesupport.com/clientsupport/pages/products/va/download.html

1. Open your Firefox browser and Go to →
http://www.nielsenonlinesupport.com/clientsupport/player_example/html5/standalone_html5.html



Nielsen Tutorial - Native HTML5 Video Tag Implementation
Note: to replay the video you will have to refresh the page

Nielsen Tutorial - Native HTML5 Video Tag Implementation
Note: to replay the video you will have to refresh the page

2. Look under View → Page Source (Ctrl-U) → Take a look at the <body>

```
<body onLoad="myAddListener()">

    <script type="text/javascript" src="ggcmb370.js"></script>
        <script type="text/javascript" src="ggCom_embed.js"></script>
        <script type="text/javascript">
                var _nolggGlobalParams = {

                                clientid:"us-502202", // provided by Nielsen
                                vcid : "c09",
                                sfcode : "us",
                                cisuffix : "gg",
                                prod : "sc,iag"
                };

                var gg1 = new gg();
                var uid = 0; //provided by Nielsen
                var oldFlashDetect = false; // no longer used (dummy placeholder for legecy implementations)
                var detectBrowser = true; //optional -- used to disable window object call for non-browser use
                gg1.ggInitialize(_nolggGlobalParams, uid, oldFlashDetect, detectBrowser);
    </script>


    <video controls="true" style="border:0.2cm groove black">
        <source src="bunny.mp4" />
        <source src="bunny.ogg" />
    </video>

<br>Nielsen Tutorial - Native HTML5 Video Tag Implementation
<br>Note: to replay the video you will have to refresh the page

</body>
```

Nielsen JS Beacon

Bridge / Listener

Unique identifer of Client given by Nielsen

Change this to your video content

In addition, take a look at <head> →

```
<!DOCTYPE html>
<html>
<head>
    <title>Nielsen HTML5 using Combined Video Beacon</title>

    <script type="text/javascript">
        // listener function changes src
        function myNewSrc()
        {
            var myVideo = document.getElementsByTagName('video')[0];
            if(navigator.userAgent.indexOf("MSIE") != -1)
                myVideo.getElementsByTagName('source')[0].src = "ed_1024.mp4";
            else if(myVideo.canPlayType("video/ogg"))
                myVideo.src = "ed_1024.ogg";
            else
                myVideo.src = "ed_1024.mp4";
            myVideo.load();
            myVideo.play();
        }

        // add listener function to ended event
        function myAddListener()
        {
            var myVideo = document.getElementsByTagName('video')[0];
            bindTagEvents(ggl,myVideo);
            myVideo.addEventListener('ended',myNewSrc,false);
        }
    </script>

</head>
```

Searching to see if there's a Video Source.

Sending the player object to bind events.

3. Open → http://www.nielsenonlinesupport.com/clientsupport/player_example/html5/ggCom_embed.js

You will notice two major parts of code to the implementation of this file:

A) Video Census Legacy Beacon / **B) Video Analytics New Combined Beacon**

*Starting with Video Census Legacy Beacon (VC) *Optional*—*

```
4  //  **Legacy Video Census Beacon Code Begin**
5  //  Instantiate image to receive pixel for old Legacy Video Census Beacon
6  var davImg = new Image();
7  function _scdav(_sctitle,_scgroup)
8  {
9  //          _sctitle = Title of the Stream
10 //          _scgroup = Client Defined Variable
11
12     var ScImgSrc;
13
14     var ScRandom = Math.ceil(Math.random()*1000000000);
15             ScImgSrc = 'http://secure-us.imrworldwide.com/cgi-bin/m?ci=us-502202';
16             ScImgSrc += '&cg=' + escape(_scgroup);          //Program/Section Name
17             ScImgSrc += '&tl=dav0-' + escape(_sctitle);       //Title of the Stream
18             //ScImgSrc += '&c3=st,a' + escape('StreamType');  //Use only if the video stream is an advertisement
19             ScImgSrc += '&c6=vc,c09' + escape('');            //VideoCensus ID - specified by Nielsen - varies per entity
20             ScImgSrc += '&cc=1';//Cookie Check (Always on)
21             ScImgSrc += '&rnd=' + ScRandom;
22
23             davImg = "";
24             davImg = new Image();
25             davImg.src = ScImgSrc;
26 }
27 // **Legacy Video Census beacon code end**
28
```

Video Census Legacy Beacon Implementation

If you would like to find out more about VC Implementation, you may also refer to:

http://www.nielsenonlinesupport.com/clientsupport/forms/VideoCensus_Tagging_QuickStartGuide.pdf

Moving to the second part of implementation, Video Analytics New Combined Beacon (VA) —

```
29 // **New Combined Beacon Code begin**
30
31 var ggComl;
32 function bindTagEvents(beacon, sender) {
33     ggComl = new ggCom(beacon, sender);
34     var arg;
35     sender.addEventListener('loadedmetadata', function(args) {ggComl.onMediaMeta(args)}, false);
36     sender.addEventListener('play', function(args) {ggComl.onPlayEvt(args)}, false);
37     sender.addEventListener('pause', function(args) {ggComl.onPauseEvt(args)}, false);
38     sender.addEventListener('ended', function(args) {ggComl.onEndEvt(args)}, false);
39     sender.addEventListener('timeupdate', function(args) {ggComl.updatePos(args)}, false);
40     sender.addEventListener('seeked', function(args) {ggComl.onSeeked(args)}, false);
41     sender.addEventListener('seeking', function(args) {ggComl.onSeeking(args)}, false);
42     sender.addEventListener('volumechange', function(args) {ggComl.onVolume(args)}, false);
43 }
```

Event functions of different types

```
71
72  ggCom.prototype.onPlayEvt = function (arg)
73  {
74      this.movieStatus.played = true;
75      this.movieStatus.stopped = false;
76      if(!this.movieStatus.loaded)
77      {
78          this.onMediaMeta(arg);
79          return;
80      }
81      this.gg.ggPM(5,arg.target.currentTime.toFixed(1));
82      if(this.debug) {
83              this.logger("this.gg.ggPM(5,"+arg.target.currentTime.toFixed(1)+")");
84      }
85  }
86
87  ggCom.prototype.onPauseEvt = function (arg)
88  {
89      if(this.debug) {
90          this.logger("this.gg.ggPM(6,"+arg.target.currentTime.toFixed(1)+")");
91      }
92      this.gg.ggPM(6,arg.target.currentTime.toFixed(1));
93  }
94
95  ggCom.prototype.onEndEvt = function (arg)
96  {
97      if(!this.movieStatus.stopped && this.movieStatus.loaded) {
98          this.movieStatus.stopped = true;
99          this.movieStatus.loaded = false;
100         this.gg.ggPM(7,arg.target.currentTime.toFixed(1));
101         this.cur_position = 0;
102         if(this.debug) {
```

'Play' Event

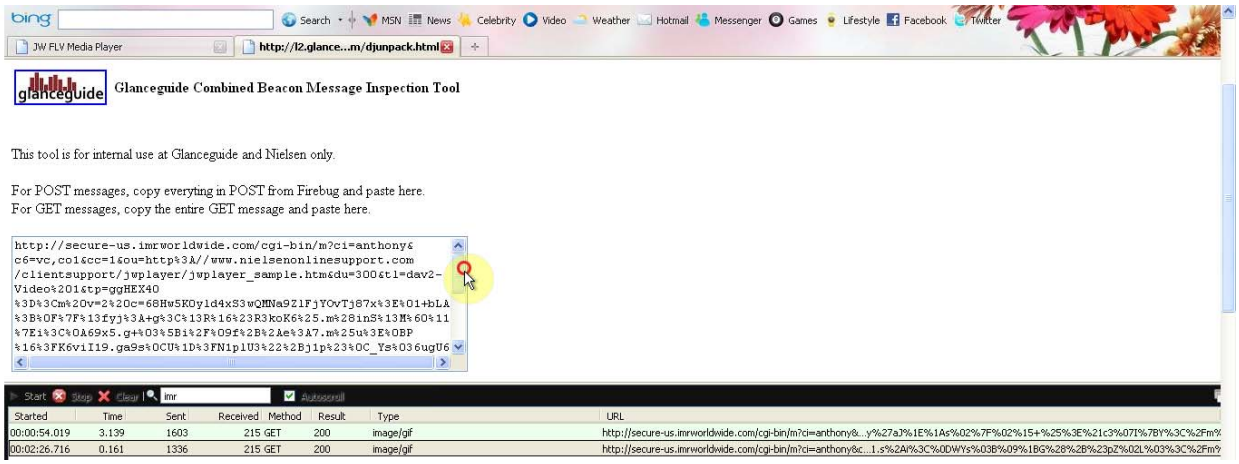
'Pause' Event

To view the full list of all the 'Player Events', you may refer to:

http://www.nielsenonlinesupport.com/clientsupport/forms/Combined_Beacon_Javascript.pdf

5. Upload your own video footage. In this example, bunny.mp4 & bunny.ogg are the video footages, which we will be testing with. Also, be sure to upload ed1024.mp4, ed1024.ogg as part completing the playlist source of this particular example.

6. Upload and test your player. After you've completed all the steps, you may now test it on your FireFox browser with tools such as HTTPFOX. Click the following link to see how testing can be done:



http://www.nielsenonlinesupport.com/clientsupport/jwplayer/jwplayer_test_screenrecord.html

For full tutorial on how to use the test tool, you may refer this to:

http://www.nielsenonlinesupport.com/clientsupport/forms/Combined%20Beacon%20Media%20Player%20QA%20Process%20Checklist.pdf

Given HTML5 Video Player also supports any major mobile devices with Webkit/HTML5 enabled, you may also test your beacon in your mobile platform. For instructions on how to test it with your PC Browser behaving like your mobile device. You can go to the following for further instructions:

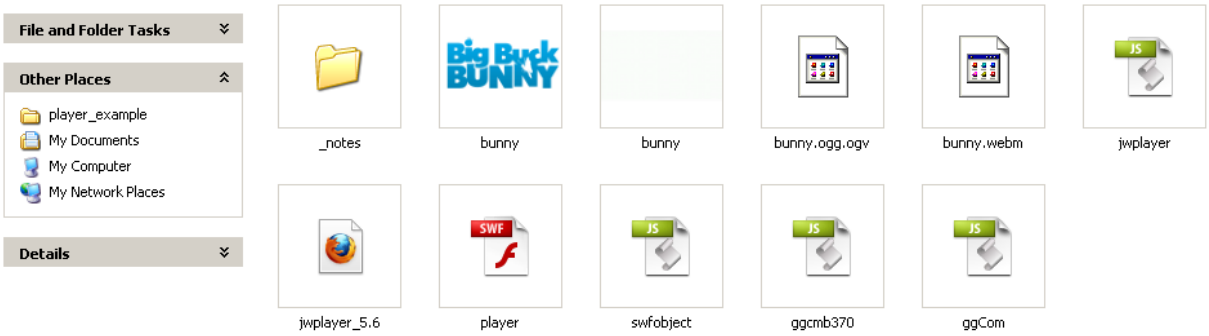
http://www.nielsenonlinesupport.com/clientsupport/forms/Combined%20Beacon%20User%20Agent%20Switcher.pdf

Tutorial & Example (JW Player 5 with HTML5)

In this section you will find a basic example of the implementation using JW Player 5 with HTML5. Please amend references to http://secure-us.imrworldwide.com to your local Nielsen collection node.

***NOTE* "nielsenonlinesupport.com" is not a production environment. Please COPY the example code and CHANGE the location settings to your local/staging/production environment.**

TUTORIAL FILES TO DOWNLOAD→
http://www.nielsenonlinesupport.com/clientsupport/player_example/jwplayer_5.6/jwplayer_5.6.zip



Please extract the zip file. Files you will be working with are:

jwplayer_5.6.html – your location URL

bunny.mp4, bunny.ogg, bunny.webm – Video Footage

ggcmb370.js – Nielsen JavaScript / Beacon

ggCom_embed – Bridge / Beacon Handler

player.swf, jwplayer.js, swfobject – Long Tail Video Player file

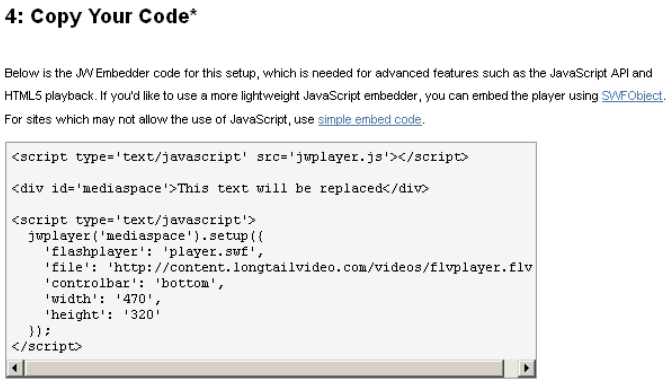
***Note* You may find the latest "Beacon" & "Bridge" files under:**

http://www.nielsenonlinesupport.com/clientsupport/pages/products/va/download.html

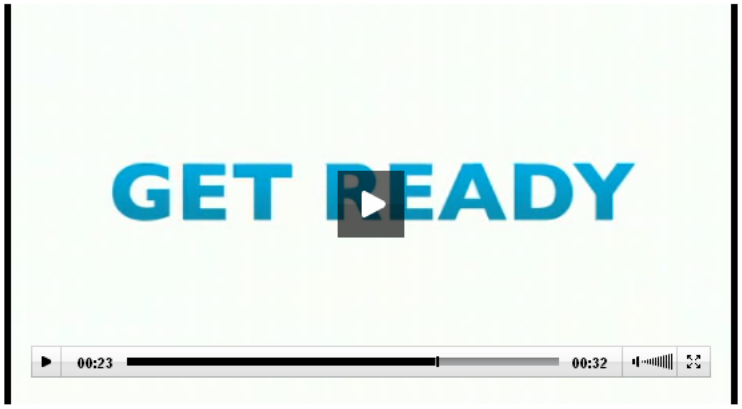
If you like to find out more about JW Player 5 Implementation, go to the following and use the **JW Embedder code setup**:

JW Player 5 Wizard

http://www.longtailvideo.com/support/jw-player-setup-wizard



1. Open your Firefox browser and Go to → http://www.nielsenonlinesupport.com/clientsupport/player_example/jwplayer_5.6/jwplayer_5.6.html



2. Look under View → Page Source (Ctrl-U) → Take a look at the <head>

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>JW Player 5.6 with Nielsen VA Beacon</title>

<script type="text/javascript" src="ggcmb370.js"></script>
<script type="text/javascript" src="ggCom.js"></script>

<script type="text/JavaScript">
        var _nolggGlobalParams = {
            clientid : "us-502202",
            vcid : "c01",
            sfcode : "us",
            cisuffix : "gg",
            prod : "sc"
            };
</script>

<script type="text/javascript">
        var canUseSWF = false;
        var uid = 0;
        var ggl = new gg();
        ggl.ggInitialize(_nolggGlobalParams,uid,canUseSWF);
</script>

</head>
```

Nielsen JS Beacon

Bridge

Unique Identifer of Client given by Nielsen

3. Take a look at the <head> →

```
40      <script type="text/javascript">
41          function setup(id) {
42              $jw(id).setup({
43                  levels: [
44                  {file: 'bunny.mp4', image:'bunny.jpg'},
45                  {file: 'bunny.ogg', image:'bunny.jpg'},
46                  {file: 'bunny.webm', image:'bunny.jpg'}
47                  ],
48                  players:[
49                      {type:'html5'},
50                      {type:'flash', src:'player.swf'}
51                  ],
52                  components: {
53                      playlist: {
54                          position: 'right',
55                          size: '200'
56                      },
57                      controlbar: {
58                          position: 'over'
59                      }
60                  },
61                  width:'550',
62                  height:'300',
63                  skin: '',
64                  events: {
65                      onReady: function() {
66                          bindEvents (ggl, this);
67                      }
```

Different video file types supported

Sending the player object to bind events.

4. Open → http://www.nielsenonlinesupport.com/clientsupport/player_example/jwplayer5.6/ggCom.js

```
1
2   var ggComl;
3   function bindEvents(beacon, sender) {
4       ggComl = new ggCom(beacon, sender);
5       var arg;
6       sender.onMeta(function(data, player) {ggComl.onMediaMeta(data, player)});
7       sender.onPlay(function(args) {ggComl.onCurrentStateChanged(args)});
8       sender.onPause(function(args) {ggComl.onCurrentStateChanged(args)});
9       sender.onComplete(function(args) {ggComl.onCurrentStateChanged(args)});
10      sender.onIdle(function(args) {ggComl.onCurrentStateChanged(args)});
11      sender.onTime(function(args) {ggComl.updatePos(args)});
12      sender.onFullscreen(function(args) {ggComl.onFullscreen(args)});
13      sender.onMute(function(args) {ggComl.onMute(args)});
14      sender.onVolume(function(args) {ggComl.onVolume(args)});
15  }
16
```

Functions that track all the player events

5. Upload your video and test your player. After you've completed all the steps, you may now test it on your FireFox browser with tools such as HTTPFOX.