



**Integration Guide  
For Publishers  
(4.2.2\_1.8)**

**YuMe, Inc.**

**© 2012 YuMe, Inc. All rights reserved.**

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, mechanical, electronic, photocopying, recording, or otherwise, without prior written permission of YuMe, Inc.. The YuMe logo is a trademark of YuMe, Inc., registered in the U.S. and other countries. Use of the YuMe logo for commercial purposes without the prior written consent of YuMe may constitute trademark infringement and unfair competition in violation of federal and state laws.

**YuMe, Inc.**

1204 Middlefield Road  
Redwood City, CA 94063  
650.591.9400  
www.yume.com

**YuMe, the YuMe logo, are trademarks of YuMe, Inc., registered in the U.S. and other countries.**

Flash is a trademark or registered trademark of Adobe Systems Incorporated in the U.S. and/or other countries.

## CONTENTS

Contents	3
Preface	4
Technical Skills Required	5
Process Overview	6
Integration Instructions	7

## PREFACE

Welcome to YuMe, and thank you for deciding to become a YuMe Publisher. To start making money from ads, you will need to integrate our technology into your video player using the YuMe Brightcove SDK.

This technology integration process is what sets us apart from typical ad networks. We're not just a sales force. Our video advertising technology platform uses sophisticated targeting, high-impact ad units, and real-time campaign optimization to help you maximize revenue for your video inventory.

This guide explains how to integrate the YuMe SDK into your Brightcove player and provides an overview of the steps you will need to take to start receiving ads from YuMe. We want to make this process as fast and easy as possible, so please email us at [yume\\_support@yume.com](mailto:yume_support@yume.com) if you run into any issues.

If you need a complete set of video ad operations tools, you might be interested in ACE for Publishers (AFP), our publisher ad management system. AFP is a cloud-based service that integrated YuMe Publishers can use without additional development work. With AFP, you can coordinate your in-house sales team, YuMe publisher monetization from the Connected Audience Network, and third-party ad networks to make sure that you are always serving the highest-CPM ad for each impression. If you would like to learn more about AFP, please visit <http://www.yume.com/content/ace-publishers> or email [yume\\_publishers@yume.com](mailto:yume_publishers@yume.com).

## TECHNICAL SKILLS REQUIRED

Integration requires an understanding of how to develop web pages and applications using JavaScript, CSS, and HTML. You should be familiar with the fundamentals of XML and have a basic knowledge of online media formats including image and video. You can use your preferred text editing program to access most source files. We recommend the latest version of Firefox, with the FireBug extension installed, as your development browser.

You will also need to have access to and familiarity with Brightcove development tools.

You will need to have access to a web server in order to test your player, and later, during the QA process, we will ask you to host your YuMe-integrated player on a semi-public web page so that we can validate it.

Finally, you will need to start out with a Brightcove-based media player into which you will integrate the YuMe Ad serving technology. This plug-in is meant for publishers with their own Brightcove player, but we also support integration with many other online video platforms (including thePlatform and Ooyala) and publisher-editable video players (including Flowplayer, JW-Player, Open Source Media Framework and Silverlight), and we can even provide a pre-built, production quality video player of our own. If you are not sure which option is right for you, please contact us at [yume\\_publishers@yume.com](mailto:yume_publishers@yume.com).

## PROCESS OVERVIEW

### ***Step 1: Complete the YuMe Plug-in integration***

Detailed integration instructions are included later in this document. Experienced developers can usually complete the integration process in just a few days.

### ***Step 2: Integration QA***

In order for YuMe to certify your player, we need to run a few tests on it. This includes making sure that each of our ad units work in your player. If you have a public web server, we recommend that you place a copy of your player on this server in a hidden folder where we can access it. Then email us at [yume\\_support@yume.com](mailto:yume_support@yume.com) and be sure to include the URL to your test page.

### ***Step 3: Content standards review***

During the testing process, we will also review the video content on our site to ensure that it does not violate our network standards. We primarily work with major brand advertisers used to buying ads on TV, which means that we will not be able to help you monetize your content if it is user-generated, pornographic, or contains excessive foul language or vulgarity.

### ***Step 4: Start monetizing your content***

As a YuMe Publisher, you will be able to receive pre-rolls and other high-impact ad units from YuMe.

Once your integration has passed QA and we have reviewed your content and verified that it does not violate our standards, we can give you a login for our web-based campaign reporting console and start sending you ads. We will also work with you to determine which content and audience channels your videos should be assigned to and how much inventory you will want YuMe to monetize on a daily basis. Once your content has been assigned to YuMe channels, we will be able to start sending you ads from YuMe's Connected Audience Network.

## INTEGRATION INSTRUCTIONS

### Step 1: Host YuMe files on your web server

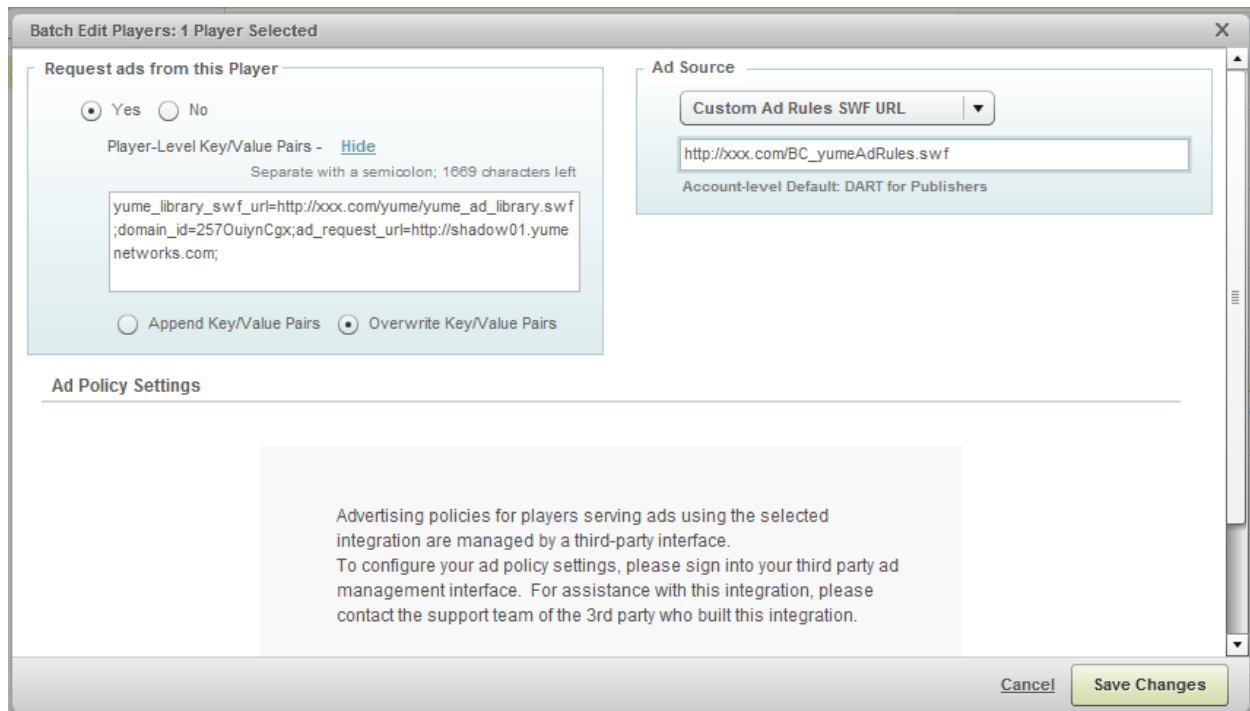
The files: `BC_yumeAdRules.swf`, `yume_ad_library.swf`, `sponsorMess.swf`, and `yume_static_cb.js`, and `empty.html`.

### Step 2: Enable ActionScript/Javascript in the Publishing section of the Brightcove console

On the Publishing tab, double-click the intended player to open its publishing settings. On the **Global > Web Settings**, select **Enable ActionScript/Javascript APIs**.

### Step 3: Enable YuMe ad requests

On the **Advertising** tab, double-click your intended player to open its advertising settings. Select **Yes** under **Request ads from this player**.



The screenshot shows a dialog box titled "Batch Edit Players: 1 Player Selected". It is divided into several sections:

- Request ads from this Player:**
  - Radio buttons for "Yes" (selected) and "No".
  - Link: "Player-Level Key/Value Pairs - [Hide](#)"
  - Text: "Separate with a semicolon; 1669 characters left"
  - Text area containing: `yume_library_swf_url=http://xxx.com/yume/yume_ad_library.swf;domain_id=257OuiynCgx;ad_request_url=http://shadow01.yume.networks.com;`
  - Radio buttons for "Append Key/Value Pairs" and "Overwrite Key/Value Pairs" (selected).
- Ad Source:**
  - Dropdown menu: "Custom Ad Rules SWF URL"
  - Text input field: `http://xxx.com/BC_yumeAdRules.swf`
  - Text: "Account-level Default: DART for Publishers"
- Ad Policy Settings:**
  - Text: "Advertising policies for players serving ads using the selected integration are managed by a third-party interface. To configure your ad policy settings, please sign into your third party ad management interface. For assistance with this integration, please contact the support team of the 3rd party who built this integration."

Buttons at the bottom right: "Cancel" and "Save Changes".

### Step 4: Enter the Key/Value Pairs string

In the same player's advertising settings, expand the **Player level Key/Value Pairs** section by clicking **edit**. In the dialog box, enter the following string, replacing bracketed paths with your hosted paths:

```
yume_library_swf_url=http://xxx.com/yume_ad_library.swf;domain_id=257OuiynCgx;
ad_request_url=http://shadow01.yumenetworks.com;
```

**Note:** The key/value pairs should be delimited with semicolon.

**Required Parameters:**

Parameter	Description
yume_library_swf_url	Location of the YuMe library Example: http://xxx.com/yume_ad_library.swf;
domain_id	Publisher domain Example: 257OuiynCgx;
ad_request_url	URL of the ad server Example: http://shadow01.yumenetworks.com;

**Optional Parameters:**

Parameter	Description
medrect	X,Y co-ordinates of the medrect companion banner Example: medrect=500,300;
fullbanner	X,Y co-ordinates of the full banner companion banner Example: fullbanner=0,0;
leaderboard	X,Y co-ordinates of the leaderboard companion banner Example: leaderboard=100,500;
qs_params	String to specify as an additional parameter while making an FMIL request. The string should contain %3D where ever an "=" is required.  If adkeys have to added along with the playlist request then \$plugin_content_adkeys should be given as is, this would be replaced with the adkeys value for respective media.  Example: <code>qs_params=pubChannel%3D\$plugin_content_tags&amp;xyz%3D\$plugin_content_reference_id&amp;videoId%3d\$plugin_content_id&amp;\$plugin_content_adkeys</code>  The same would apply for the other strings. \$plugin_content_id will be replaced with the content Id if present.

	<p>\$plugin_content_reference_id will be replaced with the reference Id if present.</p> <p>\$plugin_content_tags will be replaced with the content tags if present.</p>
requestPostroll	<p>It is a boolean value indicating whether a postroll ad should be requested or not. By default it is set to true, hence would make a postroll request.</p> <p>Example: <code>requestPostroll=false;</code></p>
preAsMid	<p>It is a boolean value indicating how a midroll ad should be requested.</p> <ol style="list-style-type: none"> <li>1. If this variable is skipped then a midroll will not be requested at all.</li> <li>2. If set to <code>true</code> then a preroll playlist will be requested for all the cuepoints</li> <li>3. If set to <code>false</code> then a midroll playlist will be requested for all the cuepoints</li> </ol> <p>Example: <code>preAsMid=false;</code></p>
support_leader	<p>It is a boolean value indicating whether a leader ad should be requested before a preroll.</p> <p>Example: <code>support leader = false;</code></p>
disable_video_list	<p>It is a string indicating the id of the tileList which has to be disabled during ad.</p> <p>Only if this string is present then the thumbnails will be disabled.</p> <p>For default players the value should be <code>videoList</code></p> <p>Example: <code>disable video_list=videoList;</code></p>

### Required Parameters – Video Display players:

Parameter	Description
spons_mess_url	<p>It is the path to load the swf for a videoDisplay component. If this is not provided then a text with "Sponsor Message" will be displayed. This parameter is used only for a videoDisplay players.</p> <p>Example:</p> <pre>http://xxx/sponsorMess.swf?height%3D20&amp;Width%3D380&amp;padding%3D5;</pre>
controlsHeight	<p>It is height of the media control bar specified for a BEML template.</p> <p>Example: <code>controlsHeight= 30;</code></p>
videoDisplayComp	<p>It is a boolean value defaulted to <code>false</code>. If you are planning for BEML integration with VideoDisplayComponent tag in BEML, then set this variable to <code>true</code>.</p> <p>If <code>VideoDisplayComp</code> is set to <code>true</code>, height of the media control bar</p>

<p>should be passed          Example: VideoDisplayComp=true;</p>
--

### Step 5: Point to BC\_yumeAdRules.swf

Under **Ad Source**, select **Custom Ad Rules SWF URL** and enter your server path for BC\_yumeAdRules.swf.

### Step 6: Configure an In-Page companion banner.

First, you will need to include a reference to yume\_static\_cb.js in the HTML of the page that will host your player.

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

Then, you will need to include code to link the companion banner element to the video player:

```
<script>
function yumeAdPlayer(){
var yumeCbAdObject1 = new yumeStaticCbAd();
yumeCbAdObject1.setPlayerMediumRectangleCompanionBannerDivSource("cb medrect1 div");
yumeCbAdObject1.setPlayerAdDomainUrl("http://shadow01.yumenetworks.com/");
yumeCbAdObject1.setPlayerIframeBannerPlaylist("dynamic banner iframe.html?domain=257OuiynCgx");
//temporary domain ID
}
</script>
```

Note that you need to make a call to yumeAdPlayer() in the "onload" function of your HTML body.

Finally, you will need to place a div element containing the companion banner iFrame into your page as well as host the empty.html also included in this zip:

```
<div id="cb medrect1 div" style="width: 300px; height: 250px; float:right; overflow:hidden">
<iframe name="cb medrect1 frame" id="cb medrect1 frame" src="empty.html" frameborder="0"
scrolling="no" marginheight="0" marginwidth="0" topmargin="0" leftmargin="0"
allowtransparency="true" width="300" height="250">
</iframe>
</div>
```

When we have a companion banner, the integration will make a call back to the functions in the JS to place the banner in the "cb\_medrect1\_div" spot on the page. When YuMe doesn't have a companion banner, we can serve your existing 300x250 tag or a static iFrame, just let us know the URL and we'll send it back to your page.

### A note about WMODE and Z-INDEX

In order for the YuMe SDK to properly render advanced ad units, we ask that you set the wmode property of your video player swf to either opaque or transparent. Not setting the wmode property or setting it to windowed will result in improperly served ad units.

Additionally, it's important not to set your player container elements' z-index property too high. If the z-index is set at its maximum allowable level, no advanced ad unit will be able to display over it. This is required for our higher-CPM, advanced units. We recommend a z-index value for player containers < 999.

***Step 7: Set up a test page with your player and call an ad***

When you've created a test page with an integrated player that shows a sample playing ad, send an email, including the test page URL, to [yume\\_support@yume.com](mailto:yume_support@yume.com) and we will begin the QA process.

Once QA approves the player, we will provide you with your production information.

We look forward to working with you!