

DIGITAL ADVERTISING SETUP GUIDE

Questions? Contact your Ad Operations Campaign Manager,
or email campaignmanagers@globe.com



BostonGlobeMedia

TABLE OF CONTENTS

1 - THE BASICS

- DISPLAY REQUIREMENTS OVERVIEW3
- BROWSER TESTING4
- LEAD TIME REQUIREMENTS.....5

2 - DIGITAL AD SPECS

- BOSTON.COM6
- BDCWIRE7
- BETABOSTON8
- BOSTONGLOBE.COM.....9
- CLASSIFIEDS 10

3 - CLICKTAG SET UP

- ADDING A CLICKTAG (AS 3)..... 11
- USING MULTIPLE CLICKTAGS 12

4 - ANIMATION

- ANIMATION TIME LIMITS 14
- FLASH: ADJUSTING FRAME RATE 15
- FLASH: STOPPING ANIMATION..... 16
- FLASH: LOOPING ANIMATION..... 17

5 - RICH MEDIA AD UNITS

- INTRODUCTION 18
- ABOUT THIRD PARTY RICH MEDIA TAGS..... 19
- SLIDING BILLBOARD WITH PENCIL (BOSTON.COM) 20
- SLIDING BILLBOARD WITH PENCIL (BOSTONGLOBE.COM)..... 23
- OVERLAY 26
- INTERSTITIAL 29
- PUSHDOWN.....30
- PUSHOVER 33

1 - THE BASICS

DISPLAY REQUIREMENTS OVERVIEW

Standard Units

Standard units are defined as any advertising confined to preset ad positions on any Boston Globe Media digital property. Common sizes include 300x250, 728x90, 160x600, and others. Consult an Ad Operations Campaign Manager for available sizes on any property.

- All motion/animation must cease at 30 seconds.
- Animated ads may loop but must cease after third display, or at 30 seconds, whichever comes first.
- Ads may animate indefinitely if user initiated via click on a clearly labeled button.
- No sound/audio unless user initiated via click on a clearly labeled button.
- In-banner video may play on load but must be muted; sound can be turned on by user click, and optionally rewind to start of video. A prominent pause button must also be included.
- Flash ads must be published for Flash Player 10.1 or lower.
- Static ads can be accepted as JPG, GIF, or PNG.
- Consult ad specs on page 22 or a Campaign Manager for individual file size limits.

Rich Media Units

Rich Media units are defined as any ad that overlaps with, interrupts, or pushes editorial content. We currently offer five Rich Media products: Sliding Billboard, Overlay, Interstitial, Pushdown, and Pushover. (Limited to Sliding Billboard on BostonGlobe.com.) See Chapter 4: Rich Media Ad Units for specifics on each. For all products:

- Rich Media advertising can display automatically on page load, limited to once per user per day. This is the **“autoplay”** or **“capped”** campaign.
- On subsequent same-day pageviews, Rich Media may appear only on user click from a “companion” ad unit, such as 300x250, 728x90, or 982x30 (pencil). This is the **“replay”** or **“uncapped”** campaign.
- Autoplay must close or collapse at 10 seconds.
- Replay may stay open indefinitely until the user clicks a **“close”** button.
- Replay may optionally display different creative than autoplay.
- Flash ads must be published for Flash Player 10.1 or lower.
- Consult ad specs on page 22 or a Campaign Manager for individual file size limits.

1 - THE BASICS

BROWSER TESTING

Current Browser Support

Users view our sites on a wide variety of web browsers. To ensure the majority of users can properly view your ad, browser testing is recommended for the following browsers and versions:

Google Chrome

Version 16.0 and later

Microsoft Internet Explorer

Version 8.0 and later

Mozilla Firefox

Version 3.6 and later

Safari

Version 5.1 and later

1 - THE BASICS

LEAD TIME REQUIREMENTS

Standard IAB Ad Units

Boston Globe Media	5–7 business days (up to three different creative versions per ad size)
Client supplied	3–5 business days

Standard Rich Media (templated)

Boston Globe Media created	2 weeks prior to go live date, assets and direction due
Client supplied	1 week prior to go live date, all files due

Custom Rich Media

Boston Globe Media created	6 weeks from client kickoff call, 5 weeks from when assets are received
----------------------------	--

Custom Rich Media Timeline

6 weeks prior to launch	Client kickoff call with Boston Globe Media
5 weeks prior to launch	Client provides all assets and direction to Campaign Manager
3 weeks prior to launch	Work in-progress test page is provided to the client for visual approval
2 weeks prior to launch	Test page is sent out to our QA and Traffic teams for full testing
1 week prior to launch	Fully functioning test page is sent to client for final approval
24 hours prior to launch	Final, finished test campaign is sent to traffic to set live

2 - DIGITAL AD SPECS

BOSTON.COM

Type of Ad	Dimensions (with max file size)	Format	Animation	Placement & Notes
Overlay	Desktop: 800x600 (80K) Tablet: 550x550 (60K) Mobile: 480x480 (40K)	gif, jpg, swf	10s	Site-wide. Must be designed to overlay other Flash elements on the page. It must have a visible close button, swf format and cap to 1/user/day. Can use a 160x600, 728x90 or 300x250 banner to replay overlay.
Pencil	982x30 (20K)	gif, jpg, swf	10s	Homepage & Section Fronts. A 982x30 banner that appears below the navigation bar near the top of the page.
Pushdown	Desktop: 1044x600 (100K) Tablet: 550x550 (60K) Mobile: 480x480 (40K)	gif, jpg, swf	10s	Homepage & Section Fronts. A 1044x600 banner ad that pushes all content down.
Pushover	Desktop: 768x600 (70K) Tablet: 550x550 (60K) Mobile: 480x480 (40K)	gif, jpg, swf	10s	Homepage & Section Fronts. A 768x600 banner ad that pushes all content to the left.
Sliding Billboard w/ Pencil	Collapsed– Desktop and Tablet: 982x30 (20K) Mobile: 320x50 (10K) Expanded– Desktop and Tablet: 982x300 (80K) Mobile: 320x480 (40K)	gif, jpg, swf	10s	Homepage & Section Fronts. A 982x30 banner that appears below navigation bar, above content. Expands to 982x300, pushing content down.
Sponsor Logo	88x31 (10K)	gif, jpg	n/a	Homepage & Section Fronts.
Sponsored Section	n/a	n/a	n/a	Sponsored section on Boston.com. See sales rep for more details
Stream Banner	320x50 (10K)	gif or jpg	15s / 2 loops	Homepage well or site-wide streams (spots 1-20).
Stream Box	300x250 (40K)	gif, jpg, swf	30s / 4 loops	Homepage well or site-wide streams (spots 1-20).
Stream Horizon	600x300 (60K)	gif, jpg, swf	30s / 4 loops	Homepage well or site-wide streams (spots 1-20).
Top Right Box	300x250 (40K)	gif, jpg, swf	30s / 4 loops	Site-wide, upper right corner of all pages.
Top Right Halfpage	300x600 (60K)	gif, jpg, swf	30s / 4 loops	Site-wide, upper right corner of all pages.
Top Right Portrait	300x1050 (80K)	gif, jpg, swf	15s	Site-wide, upper right corner of all pages.
Video PreRoll	4:3 or 16:9 (1MBps)	flv, avi, mov, 3gp, mpg4, wmv, h.264, mpeg2, dv	15s	Site-wide (no weather).
Wallpaper	2000x1800 (80K)	gif or jpg	n/a	Wallpaper art must be limited to graphics only. No copy/text unless it is integral to the logo or brand name of the advertiser.

2 - DIGITAL AD SPECS

BDCWIRE

Type of Ad	Dimensions (with max file size)	Format	Animation	Placement & Notes
Skyscraper	160x600 (40K)	gif, jpg, swf	30s / 4 loops	Site-wide. Appears on the right rail in the 5th Feature Stack spot only.
Leaderboard	728x90 (40K)	gif, jpg, swf	30s / 4 loops	Site-wide. Appears at the top of each page.
Mobile Banner	320x50 (10K)	gif or jpg	15s / 2 loops	Site-wide.
Stream - Sponsored Content/Post	Image: 199x136 (10K) Logo: 88x31 (5K)	gif or jpg	static	Site-wide. Appears within stream of Boston.com stories. Looks similar to content but with a distinguishing color. Includes an image, title (50 characters max), tease (130 characters max) and logo.
Stream Banner	320x50 (10K)	gif or jpg	15s / 2 loops	Site-wide. Appears within the infinite scroll stream. Can run in spots 1-20.
Stream Box	300x250 (40K)	gif, jpg, swf	30s / 4 loops	Site-wide. Appears within the infinite scroll stream. Can run in spots 1-20.
Stream Horizon	600x300 (60K)	gif, jpg, swf	30s / 4 loops	Site-wide. Appears within the infinite scroll stream. Can run in spots 1-20.
Top Right Box	300x250 (40K)	gif, jpg, swf	30s / 4 loops	Site-wide. Appears in the upper right corner of all pages.
Top Right Halfpage	300x600 (60K)	gif, jpg, swf	30s / 4 loops	Site-wide. Appears in the upper right corner of all pages.

All ad units support 3rd party tracking.

2 - DIGITAL AD SPECS

BETABOSTON

Type of Ad	Dimensions (with max file size)	Format	Animation	Placement & Notes
Skyscraper	160x600 (40K)	gif, jpg, swf	30s / 4 loops	Site-wide. Appears on the right rail in the 5th Feature Stack spot only
Leaderboard	728x90 (40K)	gif, jpg, swf	30s / 4 loops	Site-wide. Appears at the top of each page.
Mobile Banner	320x50 (10K)	gif or jpg	15s / 2 loops	Site-wide.
Stream - Sponsored Content/Post	Image: 199x136 (10K) Logo: 88x31 (5K)	gif or jpg	static	Site-wide. Appears within stream of Boston.com stories. Looks similar to content but with a distinguishing color. Includes an image, title (50 characters max), tease (130 characters max) and logo.
Stream Banner	320x50 (10K)	gif or jpg	15s / 2 loops	Site-wide. Appears within the infinite scroll stream. Can run in spots 1-20.
Stream Box	300x250 (40K)	gif, jpg, swf	30s / 4 loops	Site-wide. Appears within the infinite scroll stream. Can run in spots 1-20.
Stream Horizon	600x300 (60K)	gif, jpg, swf	30s / 4 loops	Site-wide. Appears within the infinite scroll stream. Can run in spots 1-20.
Top Right Box	300x250 (40K)	gif, jpg, swf	30s / 4 loops	Site-wide. Appears in the upper right corner of all pages.
Top Right Halfpage	300x600 (60K)	gif, jpg, swf	30s / 4 loops	Site-wide. Appears in the upper right corner of all pages.

All ad units support 3rd party tracking.

2 - DIGITAL AD SPECS

BOSTONGLOBE.COM

Type of Ad	Dimensions	Format	Animation	Placement & Notes
Big Ad	300x250 (40K)	gif, jpg, swf	30s / 4 loops	Site-wide. All locally served ads, require a SWF, FLA, GIF/ JPG and click-through URL.
In-Banner Video Big Ad	300x250 (initial load: 60K; video load: 1.2MB)	swf	30s / 4 loops	Site-wide. A pause button and user initiated sound are required. All rich media ads need both an animated and a static creative. Flash ads cannot have site URLs embedded within the creative.
Sticky Ad	180x85 (10K)	jpg and gif	n/a	Site-wide. Runs on article pages only.
Pencil	1232x30 (25K)	gif, jpg, swf	10s	Homepage & Section Fronts. A 1232x30 banner that appears below the navigation bar near the top of the page.
Sliding Billboard	1232x415 (100K)	gif, jpg, swf	10s	Homepage & Section Fronts. A 1232x415 banner that appears over the content and slides down from below the navigation bar.

All ad units support 3rd party tracking.

2 - DIGITAL AD SPECS

CLASSIFIEDS

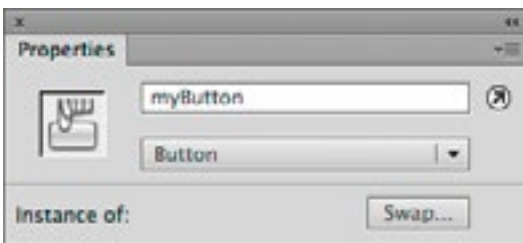
Type of Ad	Dimensions	Max File Size	Format	Animation	Placement & Notes
Marquee	275x105	30K	gif,jpg or swf	30s / 4 loops	Cars homepage. Appears on top left corner of Cars Homepage
Expanding Marquee	275x105 & 327x207	30K & 50K	gif, jpg or swf	30s / 4 loops	Cars homepage. Appears on top right corner of Cars Homepage
Tile	175x342	30K	gif, jpg or swf	30s / 4 loops	Cars homepage. Appears on right side of Cars Homepage
Box Ad	300x142	30K	gif or jpg	static	Cars homepage. Appears on left and right side of Cars Homepage
Mini Box	148x87	30K	gif, jpg	static	Cars homepage. Appears on left and right side of Cars Homepage
Mid-Banner	609x48 & 609x90	30K	gif, jpg or swf	rollover	Cars homepage. Appears in the center of Cars Homepage
Showcase Ad	400x375	80K	gif, jpg or swf	30s / 4 loops	Luxury Living homepage
Showcase Banner	329x75	30K	gif, jpg	static	Luxury Living homepage.
Spotlight Ad	460x150		Template	n/a	Search Results Page. Real Estate Section
Mini-banner	300x130	20K	gif, jpg or swf	30s / 4 loops	Real Estate homepage
T6	120x90		Template	static	Real Estate Sub section front.

3 - CLICKTAG SET UP

ADDING A CLICKTAG (AS 3)

clickTAGs are used in Flash files in place of Click URLs. The clickTAG allows the Ad Operations team to assign the destination URL within the ad server and track clicks.

Once the button is created in Flash, select the button and open the Properties panel. Give the button an **Instance Name**—for example, “myButton.”



Create an **“actions” layer** on the timeline. In Frame 1 of that layer, copy and paste the following code, replacing **xx** with the Instance Name of the button:

```
xx.addEventListener(MouseEvent.CLICK, myClickTag);

function myClickTag(e:MouseEvent) {
    var click_url:String = root.loaderInfo.parameters.clickTAG;
    if (click_url) {
        var userAgent:String = "false";
        try{userAgent = ExternalInterface.call('function(){ if(/MSIE/).
test(navigator.userAgent) || (/Trident.*rv\:11\./).test(navigator.
userAgent))){return "true"}else{return "false"} }');}catch(e){}
        if (userAgent === "true") {
            ExternalInterface.call('window.open', click_url, '_blank');
        } else {
            navigateToURL(new URLRequest(click_url), "_blank");
        }
    }
};
```

Please make sure to inform the Campaign Manager of the destination URL associated with the clickTAG.

NOTE: The *syntax* of “clickTAG” is essential—all one word, **click** all lowercase and **TAG** in caps.

3 - CLICKTAG SET UP

USING MULTIPLE CLICKTAGS

For ads that require multiple destination URLs, multiple clickTAGs should be used. The maximum number of clickTAGs allowed per ad unit is five.

For the first clickTAG, follow the same instructions outlined on Page 7.

For all subsequent clickTAGs, follow the same steps, then **add a number** to the end of the instance name, function name and “clickTAG” portion as shown in red text below. The Actionscript for the second clickTAG would be:

```
xx1.addEventListener(MouseEvent.CLICK, myClickTag1);

function myClickTag1(e:MouseEvent) {
    var click_url1:String = root.loaderInfo.parameters.clickTAG1;
    if (click_url1) {
        var userAgent:String = "false";
        try{userAgent = ExternalInterface.call('function(){ if(((/MSIE/).
test(navigator.userAgent) || (/Trident.*rv\:11\.\/).test(navigator.
userAgent)))}{return "true"}else{return "false"} }');}catch(e){}
        if (userAgent === "true") {
            ExternalInterface.call('window.open', click_url1, '_blank');
        } else {
            navigateToURL(new URLRequest(click_url1),"_blank");
        }
    }
};
```

Note there are EIGHT (8) places where the number must be added.

The Actionscript for a third clickTAG would be:

```
xx2.addEventListener(MouseEvent.CLICK, myClickTag2);

function myClickTag2(e:MouseEvent) {
    var click_url2:String = root.loaderInfo.parameters.clickTAG2;
    if (click_url2) {
        var userAgent:String = "false";
        try{userAgent = ExternalInterface.call('function(){ if( (/MSIE/).
test(navigator.userAgent) || (/Trident.*rv\:11\./).test(navigator.
userAgent))){return "true"}else{return "false"} }');}catch(e){}
        if (userAgent === "true") {
            ExternalInterface.call('window.open', click_url2, '_blank');
        }else {
            navigateToURL(new URLRequest(click_url2), "_blank");
        }
    }
};
```

Be sure to inform the Campaign Manager of the destination URL associated with each clickTAG.

NOTE: The *syntax* of “clickTAG” is essential—all one word, **click** all lowercase and **TAG** in caps.

4 - ANIMATION

ANIMATION TIME LIMITS

Standard display units that run on any Boston Globe Media property must stop all automatic animation after 30 seconds of playback or three loops of animation, whichever comes first. Sound/audio cannot play unless user initiated via click on a clearly labeled button. Animation and sound may continue indefinitely if user initiated via click on clearly labeled buttons.

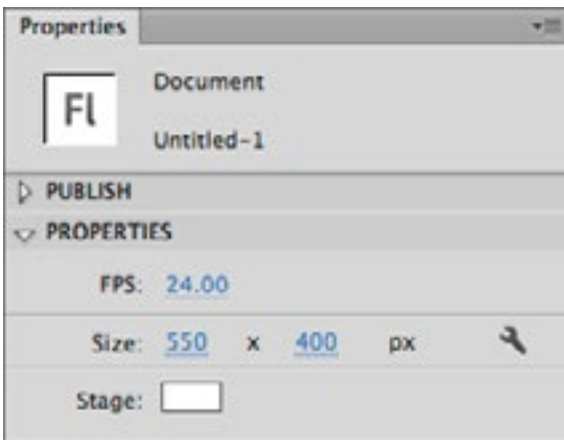
Rich Media ads interrupt, overlap or move editorial site content. These units have an automatic initial load that must stop animation and close after 10 seconds of playback. User initiated replays may remain open indefinitely until the user hits a **“Close”** button.

4 - ANIMATION

FLASH: ADJUSTING FRAME RATE

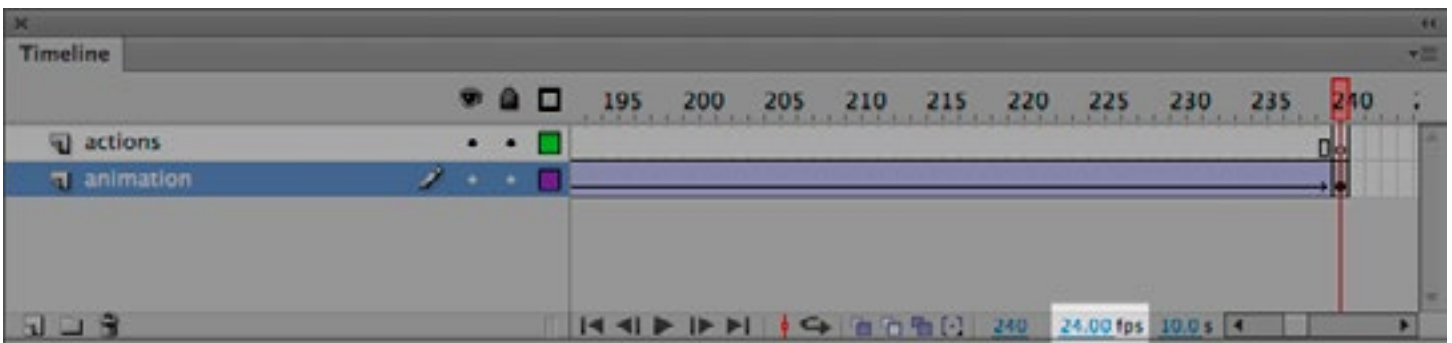
The speed at which the animation plays in Flash is called the Frame Rate and it is measured in Frames Per Second (FPS).

In the Properties panel, select the FPS setting and adjust.



Increasing the number will speed up the animation; decreasing the number will slow down the animation. Our recommended Frame Rate is 24 FPS, as most users' computers will not process a higher FPS.

The frame rate can also be adjusted at the bottom of the Timeline panel.



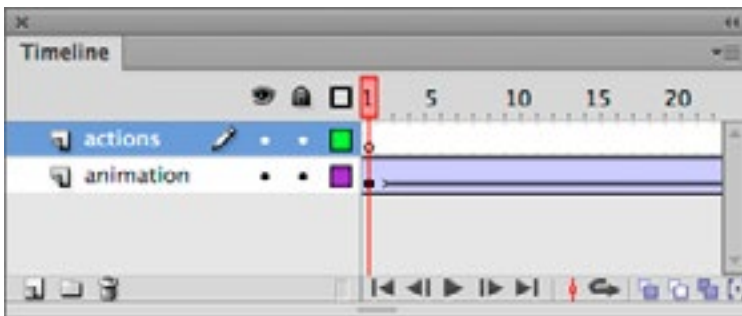
In a Flash file with a frame rate of 24 FPS, the final frame should be Frame 720 (24 FPS x 30 seconds = Frame 720).

4 - ANIMATION

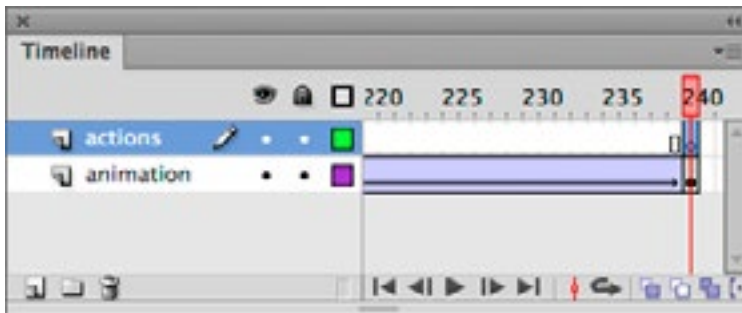
FLASH: STOPPING ANIMATION

Once the animation has completed, it is important to add a **stop command** to prevent the animation from endlessly looping.

Add an **“actions” layer** to the timeline.



In the final frame of the animation, create a **blank key frame** in the “actions” layer.



Select the key frame just created and open the Actions panel and enter the following Actionscript:

```
stop();
```


4 - ANIMATION

FLASH: LOOPING ANIMATION

If the animation in an ad is short, it can be looped while still complying with the time limit guidelines outlined on Page 14.

First, follow the instructions on Page 16 to set up an “actions” layer and final keyframe.

Select the key frame just created and open the Actions panel. Copy and paste the following Actionscript:

```
stop();

var totalLoops:int = X;
var loopCount:int;

if(loopCount != totalLoops) {
    gotoAndPlay(1);
    loopCount++;
}
```

Replace “**X**” with the number of times the animation should loop before stopping. The end of the last loop must be completed within 30 seconds. For example, if the animation from beginning to end is 10 seconds, it can only loop 3 times.

5 - RICH MEDIA AD UNITS

INTRODUCTION

Rich Media ads interrupt, overlap or move editorial site content. These units have an automatic initial load that must stop animation and close after 10 seconds of playback. Sound/audio cannot play unless a user initiates via click on a clearly labeled button.

User initiated replays may remain open indefinitely until the user hits a **“Close”** button.

These ad units automatically display once per user per day. Upon future visits, they appear on user initiation only. Animation, audio and video can reoccur when a user initiates an ad.

All instructions in this chapter detail the required set up in **Adobe Flash** to work with Boston Globe Media rich media templates.

"FLA" refers to the source Flash file for editing, and "SWF" refers to the published file that is trafficked to BGM.

Alternatively, static creative can also be used, which require only the static assets to be delivered.

For optimal integration with our Rich Media templates, it is strongly suggested that only Actionscript 3 be used.

5 - RICH MEDIA AD UNITS

ABOUT THIRD PARTY RICH MEDIA TAGS

Stop! Consider using a BGM Rich Media template...

We have developed templates for all of our Rich Media ad products. They have been optimized to run perfectly on our sites, are rich in functionality and compatibility, and are at the forefront of ad rendering technology.

Other advantages:

- Templates have been built to be responsive, as all BGM sites are now responsively designed. Creative(s) will remain visible and display optimal creative no matter the browser window size.
- Templates have been through rigorous quality assurance testing on all modern browser types, operating systems, and smartphone/tablet devices.
- Flash and static formats are all compatible with templates.
- Our templates allow for alternative creative to display on replay (or any user invoked action), giving you greater flexibility to convey your message.
- The advertiser can optionally provide a 1x1 transparent tracking pixel.

If your campaign still requires a third party rich media tag...

To prevent technical issues, immediately inform your Campaign Manager of your intention to use a third party tag.

Other considerations:

- Ads on BGM sites are initially loaded into an out-of-page 1x1 ad slot. Not all third party tags work as predicted in this environment. As a result, **testing is required** to ensure reliable display.
- In order to test, **the tag must be delivered no later than 2 weeks prior to campaign launch.**
- If the deadline is not met, we will default to a BGM rich media template and the advertiser will be asked to provide assets and tracking pixels.
- All controls must fully comply with Boston Globe Media display requirements (page 3) with regard to duration, close button, expand/replay controls, muted sound, etc. **Ads that violate display requirements will be rejected.**

RICH MEDIA AD UNITS

SLIDING BILLBOARD WITH PENCIL (BOSTON.COM)

General Info

[View Live Example](#)

The Sliding Billboard pushes down editorial content. The unit has an automatic initial load that must stop animation and close after 10 seconds. Sound/audio is allowed, but cannot play unless a user initiates via click on a clearly labeled button. All ad units must include a clearly labeled **“Close”** button in the top right corner of the ad. For optimal integration with our rich media templates, it is strongly suggested that only Actionscript 3 be used.

To serve on all screens the Sliding Billboard with Pencil requires the following files:

PENCIL:

- Desktop (swf, jpg or gif): 982x30 pixels - 20K max file size
- Tablet (jpg or gif): 982x30 pixels - 20K max file size
- Mobile Static (jpg or gif): 320x50 - 10K max file size

BILLBOARD:

- Desktop (swf, jpg or gif): 982x300 pixels - 80K max file size
- Tablet Static (jpg or gif): 982x300 pixels - 80K max file size
- Mobile Static (jpg or gif): 320x480 pixels - 40K max file size

The expanded creative must have a clearly labeled “Close” button in the top right hand corner.

Pencil File Setup

In the Flash file, create an **“actions” layer** on the timeline. In the Actions panel, enter the following **security code** in the first frame of that layer:

```
Security.allowDomain("http://cache.boston.com");  
Security.allowDomain("http://www.boston.com");  
Security.allowDomain("http://graphics.boston.com");  
Security.allowDomain("http://boston.com");
```

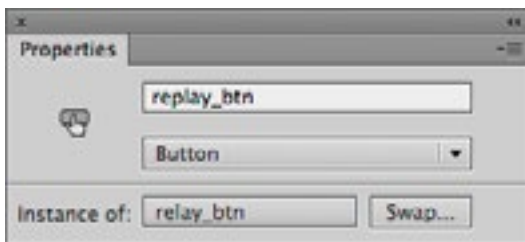
NOTE: All code samples are in Actionscript 3.0

In the same Actions panel, enter the following **import code** underneath the securities code.

```
import flash.external.ExternalInterface;
```

Adding a Replay Button

Create a prominent **“Replay”** button on the right side of the creative. The replay button should be a minimum size of 100x30 pixels. The replay button layer must be above the clickTAG layer; otherwise, the clickTAG will supercede it and the replay button will not work. Select the replay button, and give it an instance name in the properties panel:



In the **“actions” layer** on the timeline. In Frame 1 of that layer, enter the following code, replacing **xx** with the Instance Name of the button:

```
xx.addEventListener(MouseEvent.CLICK, replay_billboard);  
  
function replay_billboard(myEvent:MouseEvent){  
    ExternalInterface.call(root.loaderInfo.parameters.openAd);  
}
```

Billboard File Setup

In the desktop Flash file, create an **“actions” layer** on the timeline. In the Actions panel, enter the following **securities code** in the first frame of that layer:

Continued on the next page...

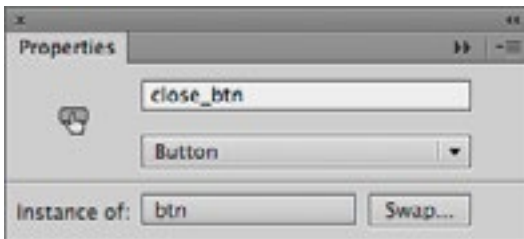
```
Security.allowDomain("http://cache.boston.com");
Security.allowDomain("http://www.boston.com");
Security.allowDomain("http://graphics.boston.com");
Security.allowDomain("http://boston.com");
```

In the same Actions panel, enter the following **import code** underneath the securities code.

```
import flash.external.ExternalInterface;
```

Adding a Close Button

Create a prominent **“Close”** button at the top right of the creative. The close button should be a minimum size of 100x30 pixels. The close button layer must be above the clickTAG layer; otherwise, the clickTAG will supercede it and the close button will not work. Select the close button, and give it an instance name in the properties panel:



In the “actions” layer on the timeline. In Frame 1 of that layer, enter the following code, replacing **XX** with the Instance Name of the button:

```
xx.addEventListener(MouseEvent.CLICK, close_billboard);

function close_billboard(myEvent:MouseEvent){
    ExternalInterface.call(root.loaderInfo.parameters.closeAd);
}
```

RICH MEDIA AD UNITS

SLIDING BILLBOARD WITH PENCIL (BOSTONGLOBE.COM)

General Info

[View Live Example](#)

The Sliding Billboard pushes down editorial content. The unit has an automatic initial load that must stop animation and close after 10 seconds. Sound/audio is allowed, but cannot play unless a user initiates via click on a clearly labeled button. All ad units must include a clearly labeled **“Close”** button in the top right corner of the ad. For optimal integration with our rich media templates, it is strongly suggested that only Actionscript 3 be used.

To serve on all screens the Sliding Billboard with Pencil requires the following files:

PENCIL:

- Desktop (swf, jpg or gif): 1232x30 pixels - 25K max file size
- Tablet (jpg or gif): 982x30 pixels - 20K max file size
- Mobile Static (jpg or gif): 320x50 - 10K max file size

BILLBOARD:

- Desktop (swf, jpg or gif): 1232x415 pixels - 80K max file size
- Tablet Static (jpg or gif): 982x300 pixels - 80K max file size
- Mobile Static (jpg or gif): 320x480 pixels - 40K max file size

The expanded creative must have a clearly labeled “Close” button in the top right hand corner.

Pencil File Setup

In the desktop Flash file, create an **“actions” layer** on the timeline. In the Actions panel, enter the following **securities code** in the first frame of that layer:

```
Security.allowDomain("http://cache.boston.com");  
Security.allowDomain("http://www.boston.com");  
Security.allowDomain("http://graphics.boston.com");  
Security.allowDomain("http://boston.com");
```

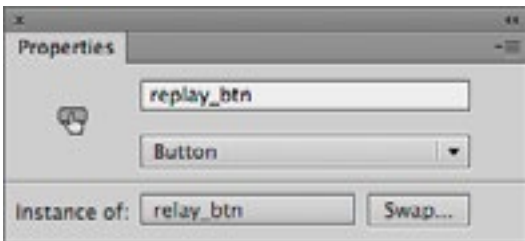
NOTE: All code samples are in Actionscript 3.0

In the same Actions panel, enter the following **import code** underneath the securities code.

```
import flash.external.ExternalInterface;
```

Adding a Replay Button

Create a prominent **“Replay”** button on the right side of the creative. The replay button should be a minimum size of 100x30 pixels. The replay button layer must be above the clickTAG layer; otherwise, the clickTAG will supercede it and the replay button will not work. Select the replay button, and give it an instance name in the properties panel:



In the “actions” layer on the timeline. In Frame 1 of that layer, enter the following code, replacing **xx** with the Instance Name of the button:

```
xx.addEventListener(MouseEvent.CLICK, replay_billboard);  
  
function replay_billboard(myEvent:MouseEvent){  
    ExternalInterface.call(root.loaderInfo.parameters.openAd);  
}
```

Billboard File Setup

In the desktop Flash file, create an **“actions” layer** on the timeline. In the Actions panel, enter the following **securities code** in the first frame of that layer:

Continued on the next page...

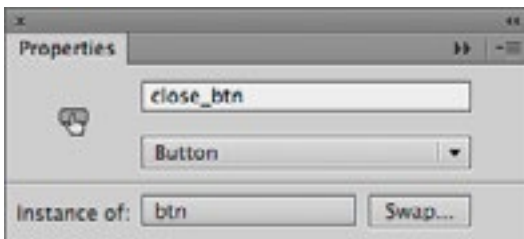

```
Security.allowDomain("http://cache.boston.com");
Security.allowDomain("http://www.boston.com");
Security.allowDomain("http://graphics.boston.com");
Security.allowDomain("http://boston.com");
```

In the same Actions panel, enter the following **import code** underneath the securities code.

```
import flash.external.ExternalInterface;
```

Adding a Close Button

Create a prominent **“Close”** button at the top right of the creative. The close button should be a minimum size of 100x30 pixels. The close button layer must be above the clickTAG layer; otherwise, the clickTAG will supercede it and the close button will not work. Select the close button, and give it an instance name in the properties panel:



In the “actions” layer on the timeline. In Frame 1 of that layer, enter the following code, replacing **xx** with the Instance Name of the button:

```
xx.addEventListener(MouseEvent.CLICK, close_billboard);

function close_billboard(myEvent:MouseEvent){
    ExternalInterface.call(root.loaderInfo.parameters.closeAd);
}
```

RICH MEDIA AD UNITS

OVERLAY

General Info

[View Live Example](#)

The Overlay covers editorial content on Boston.com. These units have an automatic initial load that must stop animation and close after 10 seconds. Sound/audio is allowed, but cannot play unless a user initiates via click on a clearly labeled button. All ad units must include a clearly labeled **“Close”** button in the top right corner of the ad. For optimal integration with our rich media templates, it is strongly suggested that only Actionscript 3 be used.

To serve on all screens the Overlay requires the following files:

- Desktop (swf, jpg or gif): 800x600 pixels - 80K max file size
- Tablet Static (jpg or gif): 550x550 pixels - 60K max file size
- Mobile Static (jpg or gif): 480x480 - 40K max file size

Overlays must have a clearly labeled “Close” button in the top right hand corner.

File Setup

In the desktop Flash file, create an **“actions” layer** on the timeline. In the Actions panel, enter the following **securities code** in the first frame of that layer:

```
Security.allowDomain("http://cache.boston.com");  
Security.allowDomain("http://www.boston.com");  
Security.allowDomain("http://graphics.boston.com");  
Security.allowDomain("http://boston.com");
```

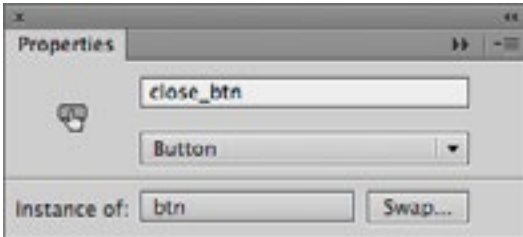
In the same Actions panel, enter the following **import code** underneath the securities code.

```
import flash.external.ExternalInterface;
```

NOTE: All code samples are in Actionscript 3.0

Adding a Close Button

Create a prominent **“Close”** button at the top right of the creative. The close button should be a minimum size of 100x30 pixels. The close button layer must be above the clickTAG layer; otherwise, the clickTAG will supercede it and the close button will not work. Select the close button, and give it an instance name in the properties panel:



In the **“actions” layer** on the timeline. In Frame 1 of that layer, enter the following code, replacing **xx** with the Instance Name of the button:

```
xx.addEventListener(MouseEvent.CLICK, close_overlay);

function close_overlay(myEvent:MouseEvent){
    ExternalInterface.call(root.loaderInfo.parameters.closeAd);
}
```

The initial load can only run for 10 seconds. The replay can be a separate file that may remain open indefinitely until the user hits the close button. A close button must be included in both files.

Replaying the Overlay

For an Overlay to replay, it must be accompanied by a **“companion”** ad unit—a big ad (300x250 pixels) or leaderboard (728x90 pixels)—with a replay button.

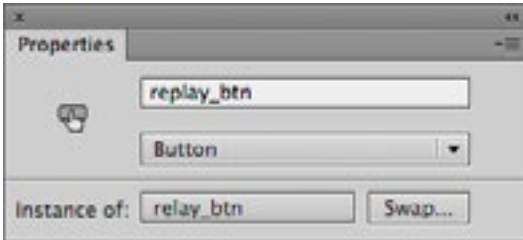
For the unit with the replay button, add the **security** code to the **“actions” layer**:

```
Security.allowDomain("http://cache.boston.com");
Security.allowDomain("http://www.boston.com");
Security.allowDomain("http://graphics.boston.com");
Security.allowDomain("http://boston.com");
```

In the same Actions panel, enter the following **import code** underneath the securities code.

```
import flash.external.ExternalInterface;
```

Create a **“Replay”** button in the ad. Select the button and give it an **Instance Name** in the Property Panel.



Add the following code to your actions layer on frame 1:

```
xx.addEventListener(MouseEvent.CLICK, replayAd);

function replayAd(myEvent:MouseEvent){
    ExternalInterface.call(root.loaderInfo.parameters.openAd);
}
```

NOTE: Be sure to changed the **"xx"** to match your instance name.

RICH MEDIA AD UNITS

INTERSTITIAL

General Info

[View Live Example](#)

The Interstitial covers editorial content on Boston.com and has an automatic initial load that must stop animation and close after 10 seconds. Sound/audio is allowed, but cannot play unless a user initiates via click on a clearly labeled button. These ads do not a close button, the template will add one automatically. For optimal integration with our rich media templates, it is strongly suggested that only Actionscript 3 be used.

To serve on all screens the Interstitial requires the following files:

- Desktop (swf, jpg or gif): 1044x800 pixels - 100K max file size
- Tablet Static (jpg or gif): 550x550 pixels - 60K max file size
- Mobile Static (jpg or gif): 480x480 - 40K max file size

Interstitials do not need a close button, the template will add a close button automatically

File Setup

In the desktop Flash file, create an **“actions” layer** on the timeline. In the Actions panel, enter the following **security code** in the first frame of that layer:

```
Security.allowDomain("http://cache.boston.com");  
Security.allowDomain("http://www.boston.com");  
Security.allowDomain("http://graphics.boston.com");  
Security.allowDomain("http://boston.com");
```

In the same Actions panel, enter the following **import code** underneath the security code.

```
import flash.external.ExternalInterface;
```

NOTE: All code samples are in Actionscript 3.0

RICH MEDIA AD UNITS

PUSHDOWN

General Info

[View Live Example](#)

The Pushdown pushes editorial content down on Boston.com and has an automatic initial load that must stop animation and close after 10 seconds. Sound/audio is allowed, but cannot play unless a user initiates via click on a clearly labeled button. All ad units need to include a clearly labeled **“Close”** button in the top right corner. For optimal integration with our rich media templates, it is strongly suggested that only Actionscript 3 be used.

To serve on all screens the Pushdown requires the following files:

- Desktop (swf): 1044x600 pixels - 90K max file size
- Desktop Static (jpg or gif): 1044x600 pixels - 90K max file size
- Tablet Static (jpg or gif): 550x550 pixels - 60K max file size
- Mobile Static (jpg or gif): 480x480 - 40K max file size

Pushdown creative must have a clearly labeled “Close” button in the top right hand corner.

File Setup

In the desktop Flash file, create an **“actions” layer** on the timeline. In the Actions panel, enter the following **securities code** in the first frame of that layer:

```
Security.allowDomain("http://cache.boston.com");  
Security.allowDomain("http://www.boston.com");  
Security.allowDomain("http://graphics.boston.com");  
Security.allowDomain("http://boston.com");
```

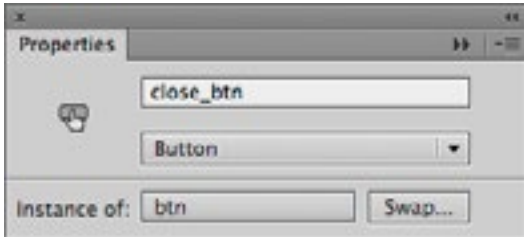
In the same Actions panel, enter the following **import code** underneath the securities code.

```
import flash.external.ExternalInterface;
```

NOTE: All code samples are in Actionscript 3.0

Adding a Close Button

Create a prominent **“Close”** button at the top right of the creative. The close button should be a minimum size of 100x30 pixels. The close button layer must be above the clickTAG layer; otherwise, the clickTAG will supercede it and the close button will not work. Select the close button, and give it an instance name in the properties panel:



In the **“actions” layer** on the timeline. In Frame 1 of that layer, enter the following code, replacing **xx** with the Instance Name of the button:

```
xx.addEventListener(MouseEvent.CLICK, close_pushdown);  
  
function close_pushdown(myEvent:MouseEvent){  
    ExternalInterface.call(root.loaderInfo.parameters.closeAd);  
}
```

The initial load can only run for 10 seconds. The replay can be a separate file that may remain open indefinitely until the user hits the close button. A close button must be included in both files.

Replaying the Pushdown

For a Pushdown to replay, it must be accompanied by a **“companion”** ad unit—a big ad (300x250 pixels) or leaderboard (728x90 pixels)—with a replay button.

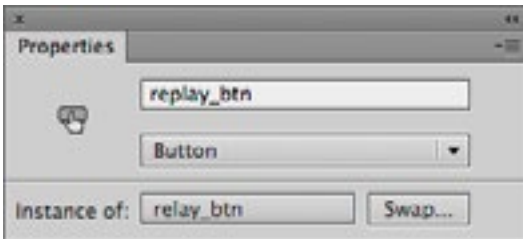
For the unit with the replay button, add the **securities** code to the **“actions” layer**:

```
Security.allowDomain("http://cache.boston.com");  
Security.allowDomain("http://www.boston.com");  
Security.allowDomain("http://graphics.boston.com");  
Security.allowDomain("http://boston.com");
```

In the same Actions panel, enter the following **import code** underneath the securities code.

```
import flash.external.ExternalInterface;
```

Create a **“Replay”** button in the ad. Select the button and give it an **Instance Name** in the Property Panel.



Add the following code to your actions layer on frame 1:

```
xx.addEventListener(MouseEvent.CLICK, replayAd);

function replayAd(myEvent:MouseEvent){
    ExternalInterface.call(root.loaderInfo.parameters.openAd);
}
```

NOTE: Be sure to changed the **"xx"** to match your instance name.

5 - RICH MEDIA AD UNITS

PUSHOVER

General Info

[View Live Example](#)

The Pushover pushes the editorial content on the Boston.com site over and has an automatic initial load that must stop animation and close after 10 seconds. Sound/audio is allowed, but cannot play unless a user initiates via click on a clearly labeled button. All ad units need to include a clearly labeled **“Close”** button in the top right corner. For optimal integration with our rich media templates, it is strongly suggested that only Actionscript 3 be used.

To serve on all screens the Pushover requires the following files:

- Desktop (swf, jpg or gif): 760x600 pixels - 70K max file size
- Tablet Static (jpg or gif): 550x550 pixels - 60K max file size
- Mobile Static (jpg or gif): 480x480 - 40K max file size

Pushovers must have a clearly labeled “Close” button in the top right hand corner.

File Setup

In the desktop Flash file, create an **“actions” layer** on the timeline. In the Actions panel, enter the following **securities code** in the first frame of that layer:

```
Security.allowDomain("http://cache.boston.com");  
Security.allowDomain("http://www.boston.com");  
Security.allowDomain("http://graphics.boston.com");  
Security.allowDomain("http://boston.com");
```

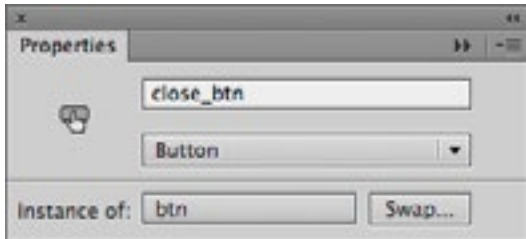
In the same Actions panel, enter the following **import code** underneath the securities code.

```
import flash.external.ExternalInterface;
```

NOTE: All code samples are in Actionscript 3.0

Adding a Close Button

Create a prominent **“Close”** button at the top right of the creative. The close button should be a minimum size of 100x30 pixels. The close button layer must be above the clickTAG layer; otherwise, the clickTAG will supercede it and the close button will not work. Select the close button, and give it an instance name in the properties panel:



In the **“actions” layer** on the timeline. In Frame 1 of that layer, enter the following code, replacing **xx** with the Instance Name of the button:

```
xx.addEventListener(MouseEvent.CLICK, close_pushover);  
  
function close_pushover(myEvent:MouseEvent){  
    ExternalInterface.call(root.loaderInfo.parameters.closeAd);  
}
```

The initial load can only run for 10 seconds. The replay can be a separate file that may remain open indefinitely until the user hits the close button. A close button must be included in both files.

Replaying the Pushover

For a Pushover to replay, it must be accompanied by a **“companion”** ad unit—a big ad (300x250 pixels) or leaderboard (728x90 pixels)—with a replay button.

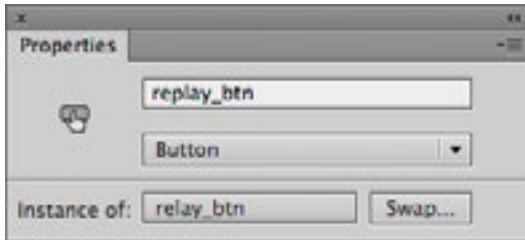
For the unit with the replay button, add the **security** code to the **“actions” layer**:

```
Security.allowDomain("http://cache.boston.com");  
Security.allowDomain("http://www.boston.com");  
Security.allowDomain("http://graphics.boston.com");  
Security.allowDomain("http://boston.com");
```

In the same Actions panel, enter the following **import code** underneath the securities code.

```
import flash.external.ExternalInterface;
```

Create a **“Replay”** button in the ad. Select the button and give it an **Instance Name** in the Property Panel.



Add the following code to your actions layer on frame 1:

```
xx.addEventListener(MouseEvent.CLICK, replayAd);  
  
function replayAd(myEvent:MouseEvent){  
    ExternalInterface.call(root.loaderInfo.parameters.openAd);  
}
```

NOTE: Be sure to changed the "xx" to match your instance name.