



Content Creation Best Practices Guide for Cisco Digital Media Designer 5.3.x

Revised: January 24, 2014



Tip


This information is updated as needed. Its newest and best revision is on Cisco.com

- [Fonts and Typography, page 1](#)
- [SWF Animations, page 3](#)
- [Audio, page 9](#)
- [Video, page 9](#)
- [HTML and JavaScript, page 10](#)
- [URLs, page 12](#)



Audience

We prepared this material with specific expectations of you.

-  *You already understand the concepts and methods that apply to content creation for personal computers. However, you want to understand the best ways to prepare and composite digital media assets for rendering by embedded, solid-state systems.*
-



Note

This guide does not describe any application programming interface (API) calls that registered Cisco developers and other advanced users can invoke to handcraft playlists outside of Cisco Digital Media Designer (DMD). To learn about these APIs, see <http://developer.cisco.com/web/dms>.

Fonts and Typography

- [Embedded in SWF Animations, page 2](#)
- [Specified in Third-party Website HTML or CSS, page 2](#)
- [Video Titles, Captions, and Other Uses of Text, page 2](#)



Cisco Systems, Inc.
www.cisco.com

Embedded in SWF Animations

When a font is embedded within a Shockwave Flash (SWF) file that your Cisco Digital Signs play, the corresponding font will render correctly even if it is not installed on your DMP.

Specified in Third-party Website HTML or CSS

DMPs use factory-installed software and fonts to render webpage text for Cisco Digital Signs. DMPs use font substitution methods when they render a webpage that is designed to use unavailable fonts.

Video Titles, Captions, and Other Uses of Text

Sometimes, digital video includes typographic representations. These might be the opening titles from a TV newscast, for example, or the translated subtitles from a foreign film. We render such characters correctly, no matter which fonts are installed on your DMPs.

Bitmapped (Raster) Images

When you use a bitmapped image in a playlist, be sure that the playlist object uses the same height and width in pixels that the bitmapped file uses. **When the playlist object is too small, DMD crops the bitmapped image.** DMD does not automatically resize or scale the bitmap.

BMP

DMP 4305G	<ul style="list-style-type: none"> Not supported.
DMP 4310G	
DMP 4400G	

Animated GIF Images

DMP 4305G	<ul style="list-style-type: none"> Supported
DMP 4310G	<ul style="list-style-type: none"> Not supported
DMP 4400G	<ul style="list-style-type: none"> Supported

GIF Images

DMP 4305G	<ul style="list-style-type: none"> Supported.
DMP 4310G	
DMP 4400G	

Progressive JPEG Images

DMP 4305G	<ul style="list-style-type: none"> Not supported.
DMP 4310G	
DMP 4400G	

Nonprogressive JPEG Images

DMP 4305G	<ul style="list-style-type: none"> Supported.
DMP 4310G	
DMP 4400G	

PNG Images

DMP 4305G	<ul style="list-style-type: none"> Supported.¹
DMP 4310G	
DMP 4400G	

1. PNG rendering is slow on DMP 4305G and 4400G endpoints, so we recommend that you prefer another image format.

SWF Animations

These guidelines cite values that are approximate. Do not deploy any design until you have tested its playback performance on at least one DMP.

**Note**

- **When errors prevent a DMP 4310G from rendering a SWF asset, failover does not begin immediately. Instead, it starts after a delay of 3 minutes and 10 seconds. The root cause is a defect in the Flash runtime from Adobe Systems. There is no workaround.**
- **A DMP 4305G might show only a white screen during failover when its failover URL points to a SWF asset. The root cause is a defect in the Flash runtime from Adobe Systems. There is no workaround. (CSCtn12943)**

**Note**

Any SWF that includes `Stage.scaleMode = "noScale"` in its ActionScript code will prevent older DMP models from scaling it correctly.

DMP 4310G Considerations

- We do not support any use of ActionScript 3 on a DMP 4310G. We support ActionScript 2.
- A DMP 4310G might stop rendering a playlist immediately before it should load and render a SWF asset. Or, it might fail to render a SWF asset in some other context. In either case, examine the FLA source for the SWF and verify that it does not use ActionScript 3.

**Tip**

When you do not have the FLA source:

1. **Open DMPDM on the DMP 4310G.**
2. **Choose Flash Playback from the Display Actions list.**
3. **Direct DMPDM to play the SWF.**

Memory Management

Rather than crashing when they run low on memory, DMPs are designed to restart automatically. This behavior clears their memory and causes downtime of much less than 1 minute, as opposed to the lengthy downtime that a hard crash would cause. In the rare cases when DMPs do run out of memory and restart automatically, SWF files are almost always responsible. The known scenarios when this can occur are as follows.

- The file size is greater than 500KB for your SWF file. Larger SWF files do work correctly in most cases, but we recommend as a best practice that you should always strive to use the smallest possible SWF files. Smaller files are far less likely to be burdensome to your DMPs.
- Your SWF file uses bitmapped image files outside itself that have a very large file size, either individually or collectively. Any bitmapped image files that you use in the production of a SWF file should be small files. If a bitmapped file has a large file size, it is important for you to understand that merely reducing the height and width of its placeholder on your canvas in Adobe Flash (or any similar authoring tool that you might use to develop a SWF file) will not reduce the actual file size.
- The web page that you are showing uses too many embedded SWF files.

Additionally, we recommend that you use the following guidelines when you create SWF files.

- The resolution of the SWF can be up to 1920 x 1080 when animations within the SWF are small and are restricted to a 640x480 region.
- Avoid redraw of the whole screen in your Flash animation.
- Multiple movements on different parts of the screen cause more slowdown than localized movement.
- The FLV recommended resolution should be 320x240.

Related Topics

- [Understand How to Render Shockwave Flash \(SWF\) Assets on a DMP 4310G, page 5](#)
- [Quality Control Considerations for SWF Animations, page 6](#)
- [Transparency Considerations for SWF Animations, page 6](#)
- [Performance Considerations for SWF Animations, page 7](#)
- [Text Considerations for SWF Animations, page 7](#)
- [Bitmap Considerations for SWF Animations, page 8](#)

Understand How to Render Shockwave Flash (SWF) Assets on a DMP 4310G

Among all other possible options for what any DMP model might render, you can direct DMPs specifically to play:

- One supported SWF asset at a time, absent any supporting context. You might use one file this way as the DMP's startup URL, failover URL, or GoToURL, for example.

OR

- Multiple assets of various supported types, all together—in tandem or in sequence—that take part in a presentation or a playlist.

Many SWFs are self-contained in the sense that they do not have any dependencies outside themselves. Nonetheless, the SWF format specification does not impose or enforce any requirement that they all be self-contained. SWF behaviors can be defined externally. Values from an XML data file that you store at “Location A,” for example, might populate a SWF that you store at “Location B.”

This content development approach supports versatility, speed, and reuse. However, a DMP 4310G cannot take advantage of this approach unless you use a special variable that only a DMP 4310G understands. Moreover, a DMP 4310G cannot begin to render a SWF asset of this kind until you tell it explicitly where to obtain the external values that it is programmed to present.



Note

Other DMP models do not require or recognize these special instructions. So, if you mistakenly instruct other DMP models to play URLs that include the variable, they merely disregard every character after the SWF filename extension in the URL. Earlier DMP models—specifically the 4305G and 4400G—are engineered to check for Flash variables in an HTML wrapper file. We use a different method for the DMP 4310G because many of its components and features differ from earlier DMP models.

Thus, the URL syntax requirements for a DMP 4310G are sharply different to render a self-contained SWF, versus one whose values are external.

- Suppose that you want your DMP 4310G to play a SWF. When the SWF is self-contained, you can use typical URL syntax to load it on the DMP. This is just what you might do with any other supported asset type or any other DMP model. If the SWF location were `http://mediahost_FQDN/path/PRESENTATION.SWF`, for example, then this would be exactly the correct URL to use, without modification.
- But now, suppose instead that exactly this same SWF at exactly this same URL is not self-contained. Instead, it relies on a range of values inside an XML data file that you store at `http://datahost_FQDN/path/VALUES.XML`. In this case, you must use special syntax to serve the full combination of all these essential goals:
 - Identify both URLs (SWF and XML) for your DMP 4310G.
 - Use a query parameter to invoke a MIB function that we call “fl_flashvars”—which exists exclusively on the DMP 4310G.
 - Pass a MIB variable that we call “Data_Url”—which indicates that the two URLs share one, unified purpose. (Third-party Flash applications would use other names for their variables.)
 - Apply percent encoding (HTTP URL encoding) to reserved characters in:
 - The MIB variable, including its “=” character (which becomes %3D).
 - The variable value, including its “:/” character string (which becomes %3A%2F%2F).

So, the final string that you would enter for this scenario in place of a conventional URL would be consistent with this example.

```
http://mediahost_FQDN/path/PRESENTATION.SWF?fl_flashvars=Data_Url%3Dhttp%3A%2F%2Fdata_
host_fqdn%2Fpath%2FVALUES.XML
```



Tip

Remember that there is no need to take these measures unless you use a DMP 4310G and your SWF depends on external values.

Quality Control Considerations for SWF Animations

- Verify per project that output quality and playback performance are both acceptable for all assets—no matter how you developed or acquired them—on all DMP models that you use.
- Never delete or overwrite your Flash (FLA) source file. You can modify its settings to produce suitable output for various DMP models.
- Consider carefully how bandwidth limitations anywhere in your network might affect playback.

Transparency Considerations for SWF Animations

- We do not support transparent SWF files.
- Avoid using transparent shapes or objects when you use animation effects or text rotation for SWF files used within playlists.
- Alpha transparency is most effective for objects whose dimensions are less than 200 x 150 pixels.

Performance Considerations for SWF Animations

We consider an animation to be nonstandard when it does not abide by our best practices to limit complexity and use DMPs effectively. And any animation is complex—for our purposes—when its playback uses so much of the CPU capacity on a DMP that performance suffers. These guidelines suggest how to avoid and reduce complexity in ways that improve performance.

Dimensions

- Ensure that each animation is smaller than 640 x 480 pixels.
- Ensure that none of your animations contain any embedded or referenced object whose dimensions are greater than 640 x 480 pixels.

Timeline Segments

- Ensure that you have not used more than one animation effect per timeline segment. For example, do not fade an object in a segment where you also resize an object.
- When you use an animation effect to resize an object, avoid changing its dimensions by more than 10 percent per timeline segment.

Other Factors

- Split the animation apart by function, into smaller files. For example, instead of combining a text ticker and a slideshow in one file, isolate them inside two separate animation files.
- Avoid combining the animation with multiple or complex JavaScripts. If you cannot simplify a complex animation, simplify your JavaScripts instead, and use fewer of them.
- Avoid shape-tweening.
- We recommend that your animation frame rate be 12 frames per second (fps) or less.

Text Considerations for SWF Animations

- DHTML's "marquee" method is an effective way to produce a text ticker.
- When your design includes a text ticker:
 - Ensure that the ticker is not any wider than 1,366px.
 - Ensure that the text height is not any greater than 28pt.

Bitmap Considerations for SWF Animations

Imported (embedded)

DMP 4305G	• Do not import low-quality or low-resolution images into your FLA source file.
DMP 4310G	• Do not import images with resolution higher than 1920x1080 in presentations.
DMP 4400G	

External (referenced)

DMP 4305G	<ul style="list-style-type: none"> • Do not refer to any external JPG file unless it is <i>nonprogressive</i>. • The maximum file size <i>per image</i> must be less than 450KB. Similarly, the <i>combined</i> file size of all images must be less than 1.45MB. Otherwise, your DMP might run out of memory and restart automatically. • Do not refer to images with resolution higher than 1920x1080 in presentations.
DMP 4310G	<ul style="list-style-type: none"> • Do not refer to any external JPG file unless it is <i>nonprogressive</i>. • The maximum file size <i>per image</i> must be less than 2MB. Similarly, the <i>combined</i> file size of all images must be less than 5MB. Otherwise, your DMP might ignore the large images and skip past them when it plays your presentations. • Do not refer to images with resolution higher than 1920x1080 in presentations.
DMP 4400G	<ul style="list-style-type: none"> • Do not refer to any external JPG file unless it is <i>nonprogressive</i>. • The maximum file size <i>per image</i> must be less than 1MB. Similarly, the <i>combined</i> file size of all images must be less than 1.5MB. Otherwise, your DMP might ignore the large images and skip past them when it plays your presentations. • Do not refer to images with resolution higher than 1920x1080 in presentations.

Exported

DMP 4305G	• Set the publishing options in Flash to 100 percent high quality JPG.
DMP 4310G	
DMP 4400G	

The Installed Flash Version on a DMP

DMP 4305G endpoints in this release run Flash 7. They support ActionScript 2.

DMP 4310G endpoints in this release run FlashLite 3.1. They support ActionScript 2.

DMP 4400G endpoints in this release run Flash 10. They support ActionScript 2 and 3

Audio

- We do not support the Dolby Digital Plus (E-AC3) audio codec.
- We do not support audio in SWF files.



Tip

You can use video editing software to add an audio track to an MPEG file.

Video

- [Progressive, Not Interlaced, page 9](#)
- [DMP Model Implications for Video, page 9](#)
- [Bitrate Implications for Video, page 9](#)
- [Aspect Ratio Implications for Video, page 9](#)
- [Limited Support for VC-1, page 10](#)

Progressive, Not Interlaced

Interlaced video is inferior to progressive-scan video. Modern equipment must expend some of its capacity to deinterlace what is interlaced. By avoiding interlaced video, you can conserve this capacity and expend it on more important functions.

DMP Model Implications for Video

- DMP 4305G endpoints do not support MPEG-2 TS/H.264.
- We recommend that you use MPEG-2 instead of FLV.

Bitrate Implications for Video

- 16 Mbps maximum for MPEG-2 SDTV (HDTV)
- 12 Mbps maximum for MPEG-4 Part 10 (H.264) SDTV (HDTV)
- 10 Mbps maximum for WMV9 SDTV (HDTV)

Aspect Ratio Implications for Video

When the width and height of a rectangle are exactly the same—making it a perfect square—we say that its *aspect ratio* is “1:1.” This notation compares width to height in relative terms. However, there are actually **two** aspect ratios to consider when you think about digital video.

- The *frame aspect ratio* describes the entire visible space, which contains every pixel.
- The *pixel aspect ratio* describes the shape that contains merely one pixel.



Note

Use a 1:1 pixel aspect ratio when you encode video for playback at a resolution of 1366 x 768 pixels.

Limited Support for VC-1



Note

Despite the extreme and misleading similarity of their names, the VC-1 codec for digital video and the .VC1 container format (“wrapper”) for digital video are not the same thing—nor are they equivalent, interchangeable, or mutually compatible.

- Some DMP models support *part* of the VC-1 codec, when used with the .ASF container format.
- No DMP models support any part of the .VC1 container format.

The Microsoft standard for its Windows Media VC-1 video codec includes three profiles: *Simple*, *Main*, and *Advanced*. Although DMP 4400G and 4310G endpoints support the *Simple* profile and the *Main* profile, they do not support the *Advanced* profile.

Video assets with .VC1 as their filename extension use the SMPTE 421M codec, which we do not support. (Such files are used on Blu-ray discs and HD DVD discs.) .VC1 files are also likely to be protected with DRM technologies that prevent unauthorized playback.

HTML and JavaScript



Note

DMP 4310G endpoints do not support HTML or JavaScript.

- Never use JavaScript code-generators, such as Microsoft Word or ASP.NET, when you create or edit anything for DMPs. JavaScript code-generators produce elaborate and wasteful code that depletes memory. When a DMP interprets such code, memory usage grows significantly and DMP performance degrades.
- Do not use more than two JavaScript timers per playlist
- Avoid refreshing any webpage frequently on DMPs. Instead, we recommend that you use AJAX or Shockwave Flash (SWF) to load dynamic content.



Tip

For more advanced JavaScript guidance, see *Best Practices to Handcraft Playlists for Cisco Digital Signs* at <http://developer.cisco.com/web/dms/docs>.

Prevent DMPs from Playing Cached Copies of Dynamic Webpages and DHTML



Note

DMP 4310G endpoints in this release do not include any web browser. Therefore, this topic applies only to DMP 4305G and 4400G endpoints.

Does your Cisco DMS network use a *content distribution system*, such as Cisco ECDS, Cisco ACNS, or Cisco WAAS? If so, special considerations apply to each **nonvideo** playlist that includes any dynamic web assets. There are two types of such assets.

DHTML

Dynamic HTML (DHTML) combines static markup, presentation layer descriptions, client-side scripting, and the Document Object Model to support animations, user interaction, and situational behaviors in webpages. In the end, DHTML's user experience relies equally on three factors:

- The quality and nature of the original page code, as it exists on its production webserver.
- Client-side rendering by browsers, after they download page code from the production webserver.
- User interactions with fully rendered DHTML in a browser.

Dynamic Webpages

Some webservers consider a range of parameters and variables per request before their server-side programming outputs fresh page code from a database. Even when they receive requests for a static URL, these servers transmit fresh page code, which generally will differ somewhat on a case-by-case basis.

The file extension for a dynamic webpage is likely to be CFM, ASP, PHP, JSP, CGI, PL, or SH, depending on the webserver's scripting environment.



Note

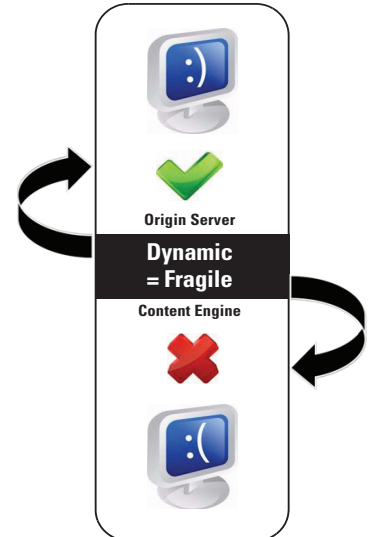
Some dynamic websites combine these methods.

The Problem

Content distribution systems such as Cisco ECDS, Cisco ACNS, and Cisco WAAS make perfect copies of files from an origin server. Then, they seed these duplicate copies—which are frozen in time—to any number of content engines in your network. So, when a content distribution network shows users its cached and unchanging snapshot of a formerly dynamic file, there is no longer anything dynamic about it.

Client systems that request the dynamic page receive only its pale substitute. And yet, dynamic values lose all validity unless they fluctuate, increase, or decrease in ways that are appropriate to their respective contexts. For example, fragile values might include:

- dates or times of day
- weather conditions
- transit schedules
- alert notifications
- seasonal messaging
- status or capacity indicators
- ostensibly current news headlines
- financial information or product and commodity prices



Cached information in these and other categories might be construed as amateurish or silly at best, and as misleading or legally actionable at worst. In any event, digital signs that play misleading information undermine their sponsors' credibility, efforts, and agenda.

Do Not Let This Happen to You

Instead, consider your alternatives. The availability and suitability of such alternatives might sometimes differ slightly between any two DMD releases or any two DMP models.

Alternatives, Tips, Options, and Best Practices

Note There is no web browser on a DMP 4310G. For this method, use a 4305G or a 4400G.

- You can create a wrapper HTML file, which calls the dynamic page. Then, add the wrapper file to your nonvideo playlist instead of adding the dynamic page. Only the wrapper file is cached. For example:

```
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>Today's Date
  </title >
    <style type="text/css">
      iframe.c1 {border:0px}
    </style >
  </head >
  <body>
    <iframe id="frame1" class="c1"
      src="http://www.example.com/includes/today_date.cfm" name="frame1">
    </iframe >
  </body >
</html >
```

Note There is no web browser on a DMP 4310G. For this method, use a 4305G or a 4400G.

- We create a wrapper for you automatically whenever the URL for a nonvideo playlist asset includes at least one query string “get” parameter. Therefore, you can use the actual URL in such cases and nothing else is required of you. For example, if the actual URL were <http://example.com/index.php?dynamic>, we would generate a wrapper for it on your behalf.

Note There is no web browser on a DMP 4310G. For this method, use a 4305G or a 4400G.

- You can append an arbitrary query string “get” parameter to the URL.¹ For example, suppose that the actual URL were <http://example.com/index.html>. You might point instead to <http://example.com/index.html?nocache>. The origin server ignores the specious parameter and delivers the actual URL instead.
- You can create a rule for your CDS that forbids it to cache the URL.²

- Do not append any query string arbitrarily to a webpage whose file extension is CFM, ASP, PHP, JSP, CGI, PL, or SH.
- Some content distribution systems might not support rules of this kind. Please check the product documentation from your system's manufacturer of record.

URLs

- Restrictions apply to your use of any URL as an asset.
 - The URL must not contain any more than 254 characters.
 - The URL must not contain any spaces.

- The URL must use ISO/IEC-8859 (Latin-1) character encoding.
- The file type and filename extension must be ones that we support explicitly. Their remote location does not make such assets any more or less compatible with Cisco Digital Signs.

