



Zoomify HTML5 Express™

Zoomify HTML5 Pro™

Zoomify HTML5 Enterprise™

**Version 2
User's Guide**

Table of Contents

Chapter I. Welcome to Zoomify.....	4
Zoomify unleashes unlimited resolution.....	4
How Zoomify works.....	6
Zoomify and compression.....	7
Zoomify delivers 'one-step image prep'.....	7
Using this Guide.....	8
Contacting Zoomify Support.....	8
Chapter II. Using Zoomify HTML5 – All Editions.....	9
Using the Zoomify Image Viewer.....	9
Converting your images into Zoomify Images.....	9
Adding the Zoomify Image Viewer to your web page.....	10
Customizing the Viewer with parameters.....	10
Advanced customization.....	11
Publishing your content - copying it to your web server.....	11
Migrating from Flash.....	12
Chapter III. Elements of the Zoomify HTML5 Products.....	14
Main product folder.....	14
1. Documentation folder.....	14
2. Viewer Features folder.....	15
3. Converter folder.....	16
4. Developer Resources folder (Pro and Enterprise editions only).....	18
Chapter IV. Zoomify Image Viewer Features.....	19
Features as demonstrated by example web pages.....	19
Viewing, creating, editing, and saving annotations.....	47
Chapter V. Advanced Topics.....	50
Customizing the Zoomify Image Viewer JavaScript.....	50
Thumbnail support.....	52
Chapter VI. Troubleshooting.....	53
Troubleshooting image conversion.....	53
Troubleshooting image viewing.....	54
Troubleshooting annotations.....	55
Chapter VII. Appendices.....	56
Tips and Tricks.....	56
Custom tile support.....	57
Complementary Zoomify products.....	60
Uninstalling Zoomify HTML5.....	61
Glossary.....	62
System requirements and additional resources.....	63

Chapter I. Welcome to Zoomify

Zoomify unleashes unlimited resolution

Zoomify makes high-quality images useful on the Web. The Zoomify family of products quickly converts images of any size or quality to *stream* for fast initial display and on-demand viewing of fine details!

Zoomify enables publishing of multi-megabyte or even multi-*gigabyte* photos that can be viewed without any large download. Visitors to your website can interactively zoom-in and explore huge images – of truly high quality – in real-time.

Zoomify's solutions are built to meet the needs of your website.

- Zoomify HTML5 Express enables your site visitors to interactively pan and zoom to explore images in detail with only a web browser. Zoomify HTML5 supports web page design flexibility with simple HTML parameters.
- Zoomify HTML5 Pro adds extensive additional features including rotation, tours, slideshows, hotspots, layout control, copyright alerting, watermarks, and more – all driven by HTML parameters, along with helpful examples, as well as complete editable JavaScript source for the Zoomify Image Viewer for complete customization control. With no Flash or other proprietary technology involved, Zoomify content works everywhere – even on iOS, Android, and other mobile devices.
- Zoomify HTML5 *Enterprise* adds sophisticated image annotation capabilities including Points of Interest, Notes, and Labels – freehand drawings, text captions, icons, rectangles, polygons, and measurements. Apply image filters including brightness, contrast, sharpness, edges, and more. Use the same intuitive viewing interface for web-based annotation editing – Markup Mode enables rapid Adding and Deleting of image annotations while Edit Mode provides formal workflow with full explicit Add/Edit and Cancel/Save capabilities.

Zoomify's solutions never involve any license fees, website traffic limits, or arcane contracts. In short, there are no strings on your images. Simply convert your images to zoom and host them on your web server wherever it may be.

A note on security: images that can be seen can be captured - if only by pointing a digital camera at a screen. Zoomify does make images more difficult to capture because it displays an entire image only at low resolution and high-resolution views only for part of an image. In addition, features like Save Image are disabled and the Zoomify Image Viewer provides non-destructive watermarking and interstitial copyright features. For a detailed discussion please see the FAQ 'How secure are my images with Zoomify?' on the Zoomify website at www.zoomify.com/support.htm.

How Zoomify works

How does Zoomify work? The Zoomify Converter intelligently converts any image for *incremental* on-demand access. The Zoomify Image Viewer is then able to display any view of the image – zoomed-in or zoomed-out, and panned left, right, up, or down.

The Zoomify Converter begins by copying an image several times at different resolution levels – from the original source resolution down to a thumbnail. The number of levels or 'tiers' depends on the size of the original source image as each tier is half the width and height of the tier below it. Zoomify cuts each of these tiers into many small tiles. The completed collect of tiles is *pyramidal* – that is, like a pyramid, stacked from a thumbnail down to the highest resolution, level upon level. These tiles are saved as separate JPEG images and organized into subfolders for fast access.

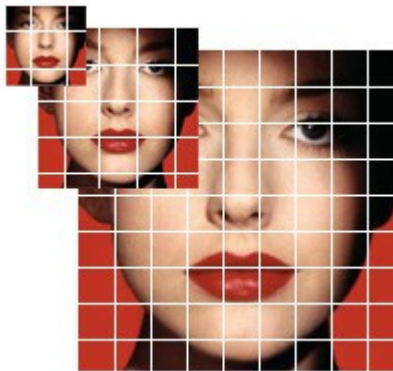


Figure 1: Pyramidal Tiled Multi-Resolution Image

When the new Zoomify Image is viewed in the Zoomify Image Viewer in a web browser, the Viewer requests only the tiles necessary to display the current view from the web server. Each pan or zoom causes a request for only a small additional number of tiles – those for the part of the image panned to, at the level of zoom desired. No tiles are ever delivered unless required for the current display, or for a display that is anticipated to immediately follow (intelligent *pre-fetching*).

These requests for image data are all made via standard HTTP Internet protocol just like all the other requests browsers make when displaying any web page. The entire process is exactly what every browser and web server were designed for – the image data has simply been better organized for faster access, and the viewing interface has been enhanced for more intuitive interaction.

Zoomify and compression

Zoomify's image streaming approach is complementary to compression because compression reduces the size of data delivered while Zoomify ensures that no unnecessary data is delivered. In fact, Zoomify leverages the complementary benefits of compression by applying JPEG compression to the image tiles within a converted Zoomify Image.

Compression alone is inadequate for publishing high-quality images. A multi-megapixel image from a digital camera or a multi-megabyte image from a stock photo house – even if compressed with wavelet technology – will still be too large for fast viewing over even high-bandwidth Internet connections. And attempting to publish an image hundreds of megabytes or even gigabytes in size – as can be created by combining images or as are commonplace in image archiving, geographic, or microscopy applications – is entirely impractical without Zoomify image streaming. Indeed, due to memory requirements and navigation limitations, such files are difficult to view without Zoomify even if the file is on your local machine.

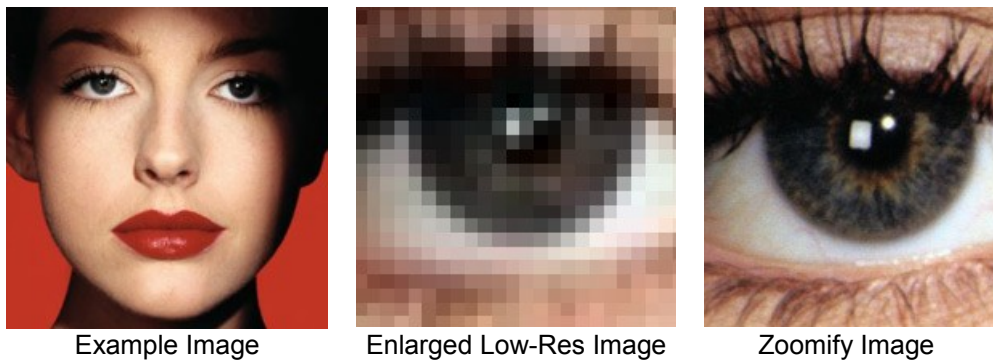


Figure 2: Image Streaming Illustrations

Zoomify delivers 'one-step image prep'

Zoomify HTML5 enables creative professionals to convert images to stream directly with the Zoomify Converter. Zoomify conversion presents a single, end-of-workflow step to prepare content for publishing on the Web. Simply double-click the Zoomify Converter icon to display the Converter dialog. Then use the File | Open menu to select an image to convert – or simply drop your chosen image onto the clearly labeled 'drop zone' of the Converter dialog - and the Zoomify does the rest. To convert multiple images at once (batch conversion) simply drag multiple images onto the dialog drop zone.

Please note: the image folder created by the Converter *is* your new Zoomify Image. There is no need to worry about its contents. The files and folders within are used by the Zoomify Image Viewer. You need only provide the Viewer the Zoomify Image folder name in your web page's HTML parameters and it will do the rest!

Using this Guide

This User's Guide describes the features of all of the Zoomify HTML5 products. Users unfamiliar with web page editing, high-quality imaging, or Zoomify, will most likely prefer to follow the chapters in the order presented. The chapter entitled “Using Zoomify HTML5” can be read now or referred to once you become more familiar with the Zoomify and its capabilities.

Contacting Zoomify Support

For additional help, Zoomify’s outstanding Support Team is standing by. Simply email support@zoomify.com - every effort is made to respond promptly. Standard business hours are 8 a.m. to 5 p.m. Pacific Time. Please be sure to review relevant portions of this Guide prior to contacting us and search the extensive Support FAQ on the Zoomify website at <http://www.zoomify.com/support.htm>. The Zoomify website is also your best source for the latest updates and other product news.

If reporting a problem, please provide the following information to the best of your ability:

Description:	Area of product affected. Application, browser, and operating system behavior.
Errors:	Exact wording of any error message displayed.
Steps to reproduce:	Exact order of actions leading up to the problem including every menu and button choice.
Reproducibility:	Is the problem 100% repeatable or intermittent?
Configuration:	Platform and model, operating system and version, browser and version, and Zoomify product and version.

We are committed to your successful use of Zoomify HTML5 Express, Zoomify HTML5 Pro, and Zoomify HTML5 Enterprise. Thank you for being our customer.

The Zoomify Team

Chapter II. Using Zoomify HTML5 – All Editions

If you're already familiar with Zoomify's products or you just want to jump right in, this section is for you. This step-by-step outline will explain how to use the Zoomify Image Viewer, how to create your own Zoomify Images, and even customize the Zoomify Image Viewer – all in a few minutes. And don't miss the Tips and Tricks section in the Appendix!

Using the Zoomify Image Viewer

The simplest, fastest introduction to Zoomify requires only a brief tour of the many features of the Zoomify Image Viewer using the example web pages in the main folder of the product download. To use any example (".htm" file) simply open it in a browser by double-clicking it. You'll see a Zoomify Image and be able to interact with it. NOTE: recent versions of most browsers prevent local viewing. For local testing, use Mozilla Firefox or change the settings of other browsers. For more information see the file Setting Up Local Host.txt in the folder Developer Resources | Testing Tools | Local Host or visit Zoomify website's Support page.

Converting your images into Zoomify Images

The next step in a simple introduction to Zoomify is converting your own image to zoom. You'll find the Zoomify Unlimited Converter application in the Converter subfolder of the product download. Simply double click the Converter to open it, then use the File | Open menu to select any TIF, PNG, BMP, or JPG image file to convert. During conversion you'll see a display of the conversion in progress.

When the conversion completes, you can locate the new folder the Converter has created – your new Zoomify Image – directly next to the original source image (that is, in the same folder as the source image). To most easily see your new Zoomify Image in action, you can copy it into the product folder and temporarily change its name to the name "ZoomifyImageExample". (You'll need to change the name of the standard example image first to avoid a naming conflict.) The example web pages will then show your image instead of the standard example image.

You can also 'batch convert' images – convert many at once - by simply dragging multiple image files onto the drop zone of the Zoomify Converter dialog. The Converter will then convert multiple image files sequentially. Please ensure that you have enough disk space for both the original images and the newly created Zoomify Image folders.

For the best results when converting images, we recommend that you use an image saved in the bitmap (.bmp) or TIFF (.tif or .tiff) formats as image files stored in these formats are typically uncompressed or losslessly compressed. The Zoomify Converter applies JPEG compression. In order to obtain the best possible quality you'll want to begin with a source image that has not been previously compressed with a 'lossy' image format.

Adding the Zoomify Image Viewer to your web page

The Zoomify Image Viewer is incredibly simple to use. Adding it to any web page simply requires adding three lines to the page using a text or HTML editor. The first line tells the web page to include the Viewer script file: "ZoomifyImageViewer.js". The second line creates the container element for the Viewer to exist in: for example, "myContainer". And the third line creates the Viewer, tells it the container name, and instructs it to display an image, for example, "ZoomifyImageExample".

Together, the lines look like this:

```
<head>
  <script type="text/javascript" src="ZoomifyImageViewer.js"></script>
  <style type="text/css"> #myContainer { width: 900px; height: 550px } </style>
  <script type="text/javascript"> Z.showImage("myContainer", "ZoomifyImageExample"); </script>
</head>
```

With those lines in place, the Zoomify Image Viewer can display the desired Zoomify Image anywhere in your web page where you tell it to:

```
<body>
  <div id="myContainer"></div>
</body>
```

More detailed information on this simple process is provided in the Parameters List document which is included in this product's Documentation folder.

Customizing the Viewer with parameters

The Zoomify Image Viewer included in Zoomify HTML5 Express supports many HTML parameters. These provide a simple means to customize the Zoomify Image Viewer. Zoomify HTML5 Pro's Viewer supports numerous additional HTML parameters to allow for more powerful customizations.

Applying these HTML parameters is simple. As noted above, just use any text or HTML editor to edit your web page and add additional parameters in a **special additional string** of characters. For example, the third line seen above:

```
<script type="text/javascript"> Z.showImage("myContainer", "ZoomifyImageExample"); </script>
```

can become

```
<script type="text/javascript"> Z.showImage("myContainer", "ZoomifyImageExample",
  "zInitialZoom=100"); </script>
```


This simple change will tell the Zoomify Image Viewer to initially display the Zoomify Image zoomed-in 100%.

Or, add several optional parameters joined together with the '&' character. **Please note:** the additional optional parameters are included **together** in a **third** string **separate** from the container and image path parameters.

```
<script type="text/javascript"> Z.showImage("myContainer", "ZoomifyImageExample",
    "zInitialX=1125&zInitialY=1385&zInitialZoom=100"); </script>
```

This example displays an initial view fully zoomed-in *and* panned to the specified position stated in X and Y pixel coordinates.

All supported HTML parameters are listed and described in the Zoomify Parameters List document in the Documentation folder. Additionally, each product can be seen in action in the many example web pages included in the main folder of each product download.

Advanced customization

Zoomify HTML5 Pro includes complete JavaScript source code for the Zoomify Image Viewer. You can easily edit this in the same text or HTML editor you use for editing your web pages. Detailed information about editing the Zoomify Image Viewer to implement more advanced customizations than the built-in HTML parameters enable is provided in the Advanced Topics chapter below, under the heading “Customizing the Viewer with parameters”.

Publishing your content - copying it to your web server

To publish your Zoomify Image to a website, just copy four things to your webserver:

1. The Viewer file “ZoomifyImageViewer.js”.
2. Your Zoomify Image folder (the entire folder, not just its contents).
3. The Assets folder from the product download (and its subfolders).
4. Your web page '.htm' file.

That's it!

Zoomify's HTML5-based technology is implemented using only HTML, JPEGs, and one small JavaScript file – no Flash files or other proprietary technologies are involved. There is therefore no need for any special setup. Publishing really is as simple as copying files to your website – and viewing really is as simple as pointing a web browser at the right web address.

If you haven't already, visit the Zoomify website to see the very same example web pages included in the Zoomify HTML5 products in action: <http://www.zoomify.com>.

The Assets Folder

The Assets folder contains many subfolders. Among these are the Skins folder that holds the small graphics files that are used to create the toolbar of the Zoomify Image Viewer. The Assets folder also contains other subfolders that support certain features of the Viewer such as Watermarks, or Hotspots (Pro and Enterprise products only). For many sites it is simplest to upload the entire Assets folder to your web server. For basic uses, however, only the folder "Assets/Skins/Default" will be need to be uploaded.

Important note: the Viewer expects to find a folder named "Assets" containing a folder named "Skins containing a folder named "Default" in the same location as the ZoomifyImageViewer.js file. The files can actually be placed anywhere - if the above folder structure is not used, simply use the zSkinPath parameter to specify the location of the files. For example, the 'ToolbarSkinned' example web page (Pro and Enterprise products only) uses the zSkinPath parameter to tell the Viewer to use the skin graphics in the Assets/Skins/Dark folder to create a different toolbar look.

Migrating from Flash

Migrating from Flash-based viewing to HTML5-based viewing is simple – no need to reconvert any images and only slight changes needed to web pages.

Adding the Viewer to a web page: The lines needed to display a Zoomify Image in HTML5 are fewer and simpler that were required in Flash. In addition, the Zoomify parameter names have been shortened and made more clear. The key elements, however, remain the same: the path to the image folder must be specified and the location of the Viewer file must be as well (a JavaScript '.js' files rather than a Flash '.swf' file).

Toolbar graphics: Flash was able to contain the graphics for the Zoomify Viewer's Toolbar inside the Viewer file. In HTML5 the graphics are in the Assets folder which must be uploaded to the web server. Because these graphics files are now easily accessible, 'skinning' the Toolbar is far simpler.

Controlling the Viewer from scripts in the web page: It is also far easier to control the HTML5 Viewer from JavaScript in the web page – as the Viewer is in the page and open to access (Flash Player required special and problematic techniques for 'talking into' the Viewer).

Annotations: see the section 'Updating Annotation XML Creating In Flash' in the READ ME FIRST.txt file in the main product folder for information about automatically updating annotations.xml files created in Flash.

Multiple Viewer instances in a web page: Flash Player mad it simple to have multiple instances of the Viewer in the same web page. This functionality will be introduced for the HTML5 Viewer soon.

Chapter III. Elements of the Zoomify HTML5 Products

This chapter provides a brief list of the correct contents of each product download file as viewed after it is unzipped.

Main product folder

Unzipping the product download reveals a single “_READ ME FIRST.txt” file and four folders: 1 Documentation, 2 Viewer Features, 3 Converter, and 4 Developer Resources (Prod and Enterprise editions only).

The contents of these folders is as follows:

1. Documentation folder

The Zoomify HTML5 products include multiple information resources to assist web designers and developers in most effectively leveraging the features and functionality of the product:

Zoomify License Agreement

Terms of use for Zoomify products. Developers, please read this document for appropriate use of Zoomify source code.

Zoomify HTML5 User's Guide

The User's Guide, this document, provides information on working with Zoomify HTML5 – all editions.¹

Zoomify Parameter List

The Parameter List includes helpful descriptions of every HTML parameter supported by the Zoomify Image Viewer as included in the Zoomify HTML5 products.

Zoomify Function List

The Function List includes descriptions of every JavaScript function included in the Zoomify Image Viewer codebase – the file “ZoomifyImageViewer.js”. These descriptions are grouped by function categories including Initialization, Interaction, Core, Object, Event, Get and Set, and Utility.

¹ Published in Adobe's Acrobat format – the reader is free from <http://www.adobe.com/>.

2. Viewer Features folder

The Viewer Features folder contains example web pages for all the major features of the relevant Zoomify HTML5 product.

Template web pages

Numerous example web pages are included, each of which can serve as a template for rapid creation of your own web site. See highlights of these examples in Chapter IV below 'Zoomify Image Viewer Features'. See the example themselves for included detailed descriptions.

ZoomifyImageViewer.js file

This is the Viewer. It is to be placed on the web server. It is a text file that can be opened in any text or HTML editor.

Note: this copy of the Viewer has been both 'obfuscated' and 'minified' – that is, the function and variable names have been replaced with short versions and the white space has been removed. This makes the file significantly smaller and faster to download (it will also be automatically compressed by most web servers and then compressed by the browser for additional savings in size). The fully readable version of the Viewer JS file for customization editing can be found in the folder Developer Resources | Viewer Source (Pro and Enterprise editions).

Assets subfolder

This folder contains the example copyright text file, example custom logo graphic file, skin graphic files, example watermark graphic file, and other support files that allow the many example web pages in the product to work. Pro and Enterprise editions also include support files for Hotspot and Annotation features, and more.

If the Asset Folder or its contents are moved to a different location relative to the example web page files, the example web page files will not work properly. The references they contain would have to be edited for the examples to work again.

Note: the folder Assets/Skins/Default is essential to the Viewer because it contains the graphic files for the Toolbar. It must be copied to the webserver along with the Viewer JS file.

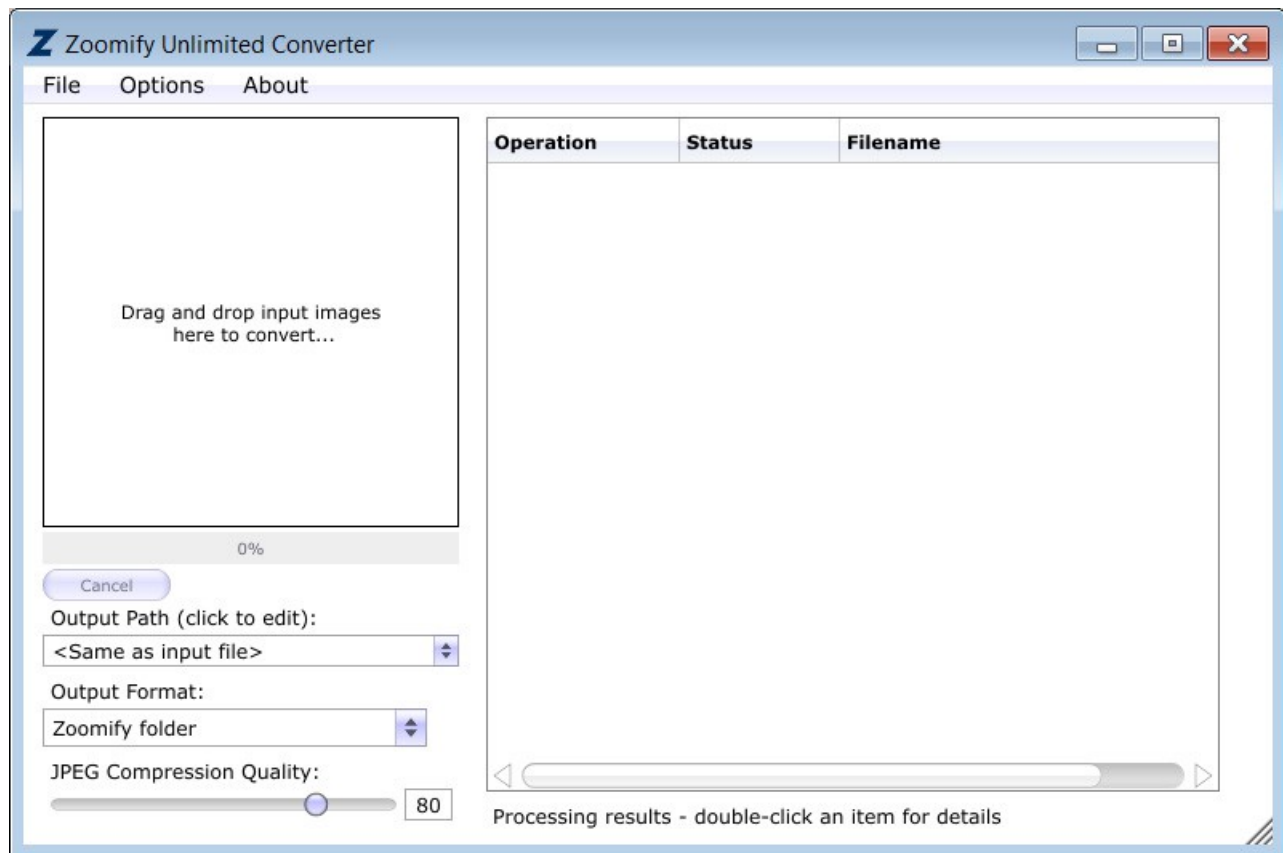
Zoomify Image Example subfolders and files

Example Zoomify Image files are the used by the many example web pages of the product. Renaming or moving an example image folder will cause one or more example web pages not to properly display that image. In that case, the names or locations will have to be corrected, or the example web pages will have to be edited to correct the name and/or add a path.

3. Converter folder

The Converter folder includes the Zoomify Unlimited Converter application, along with an example source image to test conveniently test the conversion process, and a READ ME FIRST text file describing how to use the Converter (for users who prefer to skip reading User's Guides).

To use the Converter, simply double-click the Converter icon and drop any TIF, PNG, BMP, or JPG image file onto the drop zone that is presented. Progress text is presented in the pane at right – double-click any row for detailed information. When conversion completes, the Zoomify Image Folder, ZIF file or PFF file will be found directly next to the original source image (that is, in the same folder as the source image) unless the Output Path has been changed.



Note that if Zoomify Folder output is selected, the new folder will have the same name as the source image - without any three letter extension ('.jpg', '.tif', '.bmp'). The new folder IS your new Zoomify Image. It is NOT necessary to view or modify the contents of the new folder. The XML file and subfolder containing numerous small JPEG image files are for the use of the Zoomify Viewer, not a web designer or website manager.

Note also that a JPEG compression setting of 80 is the default. This applies to the tiles created by the Converter. A lower setting may create visible 'artifacts' (jagged lines, chunky color gradations) and a higher setting will only produce any quality benefit in extremely rare cases.

No File size limits

The Zoomify Unlimited Converter included in all Zoomify products will convert source images of any size – whether tens of megabytes or hundreds of gigabytes.

Customers working with images many gigabytes in size may encounter limits in the file formats themselves – the TIF, PNG, BMP, and JPG formats all have limits (4 gigabytes for TIF and BMP, 2,147,483,647 pixels in either dimension for PNG, and 1-4 gigapixels for JPEG depending on aspect ratio). Two alternatives exist: the Zoomify Tiled-Input Converter (use multiple source images to create one Zoomify Image, separate purchase), and the Photoshop 'Export Zoomify' feature (use files in RAW or PSB format).

Zoomify Converter upgrades

Three additional Converter editions are available for separate purchase – all with support of source images of unlimited size. For detailed information see the Converters page of the Zoomify website. Key benefits of each follow:

The *Zoomify Quad-Thread Converter With Tiled Input* improves performance by increasing support from 2 processing threads to 4. This Converter also adds support for tile input source images including tiled TIF images as well as multiple input source files using the ACI format. This enables the creation of massive Zoomify Images from image sets - exceptionally useful given the size limits of many capture devices and image formats.

The *Zoomify Command-Line Converter* enables scripted conversion! Use PHP, Python, JSP, and other scripting languages to instruct the Converter to convert an image. This powerful functionality enables the integration of conversion within server-side systems and full automation of the conversion process.

The *Objective Pathology Universal Converter* delivers support for the most important microscopy formats with enhancements for maximum performance including 64-bit processing and unlimited threads. Supported formats include: tiled ACI, Aperio SVS, Bacus CWS, Mirax & 3D Histech MRXS, Motic CS, BioImagene & Ventana TIF, Hamamatsu NDPI, JPEG 2000, Leica SCN, Objective Imaging & MikroScan SWS, and Zeiss CZI.

Please note: If you are using an operating system that pre-dates Windows XP or OS X v10.5 you may need an earlier version of the Zoomify Converter. Simple contact us at support@zoomify.com and we'll provide one to you.

4. Developer Resources folder (Pro and Enterprise editions only)

Viewer Source folder

This folder contains the ZoomifyImageViewer.js source file in full editable form. This file is intended for development use. It can be edited using any text or HTML editor. Further details on this process can be found in the Advanced Topics chapter below, under the heading “Customizing the Zoomify Image Viewer JavaScript”.

Please note that this version of the Viewer is not for use on a production web site. The version of the file that is in the Viewer Features folder has been approved for use on production websites as that version of the file has been both 'obfuscated' and 'minified'.

Obfuscation replaces function and variable names with shorter versions. Minification removes white space, comments, unused functions (if any). These elements can be helpful when editing the JavaScript but are unnecessary when using it on a website. These steps reduce the Viewer file size by more than half, thereby speeding downloads and reducing bandwidth requirements.

After editing the JavaScript source file to customize the Viewer, the source file must be minified before being used on a website. This is necessary to shrink the file for optimal performance. It is also necessary to comply with the terms of the Zoomify License Agreement. More information on this process can be found in the READ ME FIRST text file in the Viewer Source folder.

Also included in the Developer Resources folder are the following subfolders - detailed information for each is provided in the Read Me First text file in each folder:

Zoomify Function List: JavaScript API information for the Zoomify Image Viewer.

One File Storage Options: Descriptions and resources for using optional ZIF or PFF storage.

Annotation Posting Options: Alternative scripts for posting (saving) edited annotations.xml data on a web server. Included are ASP, JSP, PERL, and PHP example script files (Enterprise only).

Testing Tools: Utility applications and web pages helpful when working with Zoomify Technical Support to isolate the cause of a viewing or other technical problem, or when developing large or complicated web sites incorporating Zoomify Images.

Chapter IV. Zoomify Image Viewer Features

Features as demonstrated by example web pages

The Zoomify HTML5 products each include numerous example web pages demonstrating the features of the include Zoomify Image Viewer. This section highlights some key features and describes the HTML parameters used to implement it. For complete information, view the many examples in the Viewer Features folder.

Default values

The Zoomify Image Viewer includes default values for almost every available HTML parameter – every parameter is therefore optional with the exception two: the image to display, and the place in the web page to display it. With just those two parameters you can easily use the Zoomify Image Viewer in any web page with a fully functional Toolbar and Navigator (bird's-eye-view thumbnail) and helpful zoom-to-fit initial zoom setting.



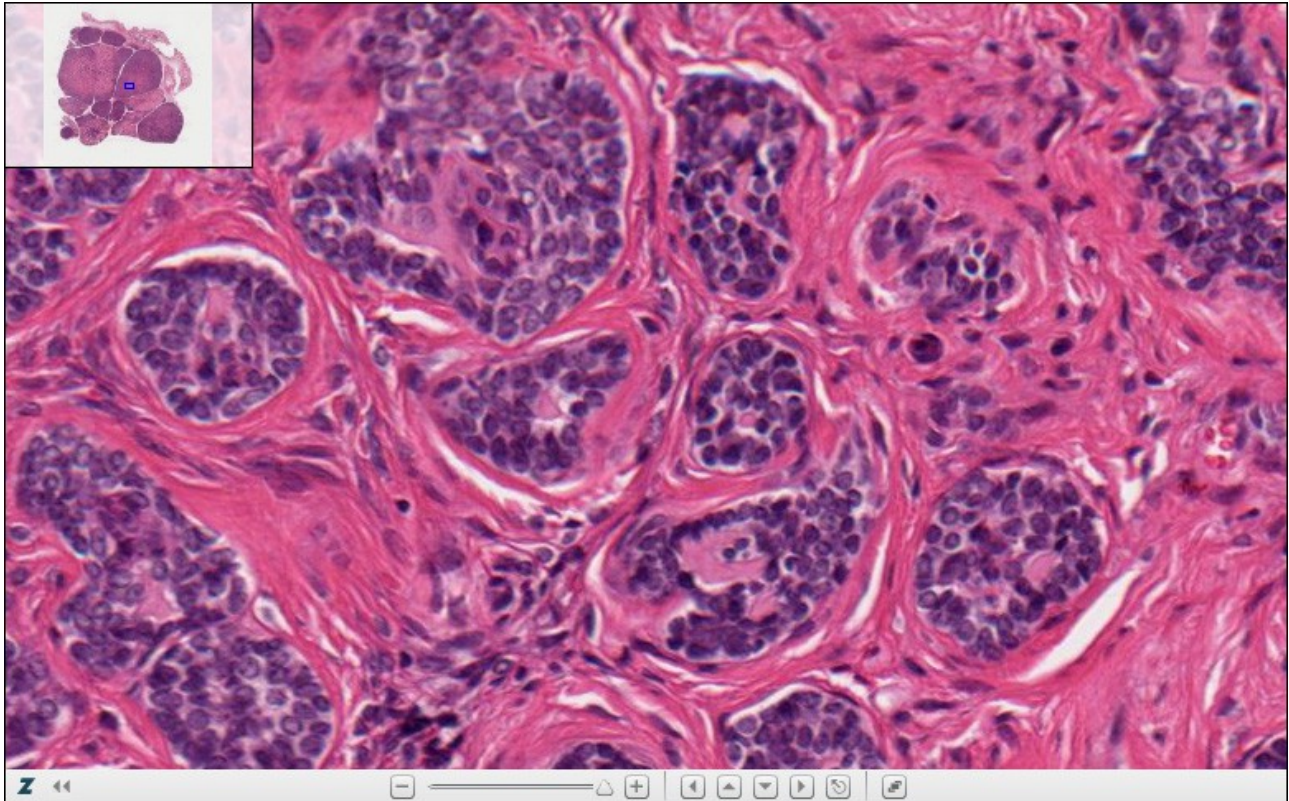
The simplest call to display the Zoomify Image Viewer looks like this:

```
Z.showImage("myContainer", "ZoomifyImageExample");
```

See this in action in the example web page "1-Defaults.htm".

Setting the initial view

A great place to begin using the Zoomify Image Viewer's optional HTML parameters is setting the X, Y, and Zoom coordinates of the initial view – the first display users will see.



The above display can be created with only the smallest addition to the default method of displaying the Zoomify Image Viewer. Instead of this:

```
Z.showImage("myContainer", "ZoomifyImageExample");
```

simply use something like this:

```
Z.showImage("myContainer", "ZoomifyImageExample", "zInitialX=1125&zInitialY=1385&zInitialZoom=100");
```

These options can be seen in action in the example web page “2-ViewPreset.htm”.

Using full screen view

The Zoomify Image Viewer has built-in support for full screen and full page views – the automatic expansion of the Viewer to overlay all other content in the screen or all other content in the web page. Full screen mode is supported in recent versions of Chrome, Firefox, and Safari. Support in Internet Explorer 11 is unreliable. Full Page view is available as an automatic fall-back. In both modes, the Toolbar moves to the bottom of the browser window and the Navigator moves to the top left. Upon leaving these modes the Viewer shrinks to its original size and position (as defined in the web page HTML) and the Navigator and Toolbar return to their original places as well.



In the above display the Viewer is seen filling the entire window. The Escape key can be used at any time to exit full page view. The button in the Toolbar that is furthest to the right can be used to enter or leave full page view. This button can be hidden using the HTML parameter `zFullViewVisible=0`, as follows:

```
Z.showImage("myContainer", "ZoomifyImageExample", "zFullViewVisible=0");
```

In addition, the HTML parameter `zInitialFullPage=1` can be used to cause a web page to load with the Viewer in full page view initially. This is not usually advisable as web site visitors may not appreciate having their view preset (note that the example web page uses JavaScript to present an alert informing the user that they can use the Escape key to exit).

These options can be seen in action in the example web page “4-View-FullPage.htm” and “5-View-FullScreen.htm”.

Skinning the Toolbar

By default the Zoomify Image Viewer loads a set of simple skin graphic files to create a Toolbar with a simple, standard appearance. It is very easy to use one of two alternative skin file sets included in the product ('Dark' and 'Light') or to create your own and point instruct the Viewer to use that. The example below uses the 'Dark' file set for a different look.



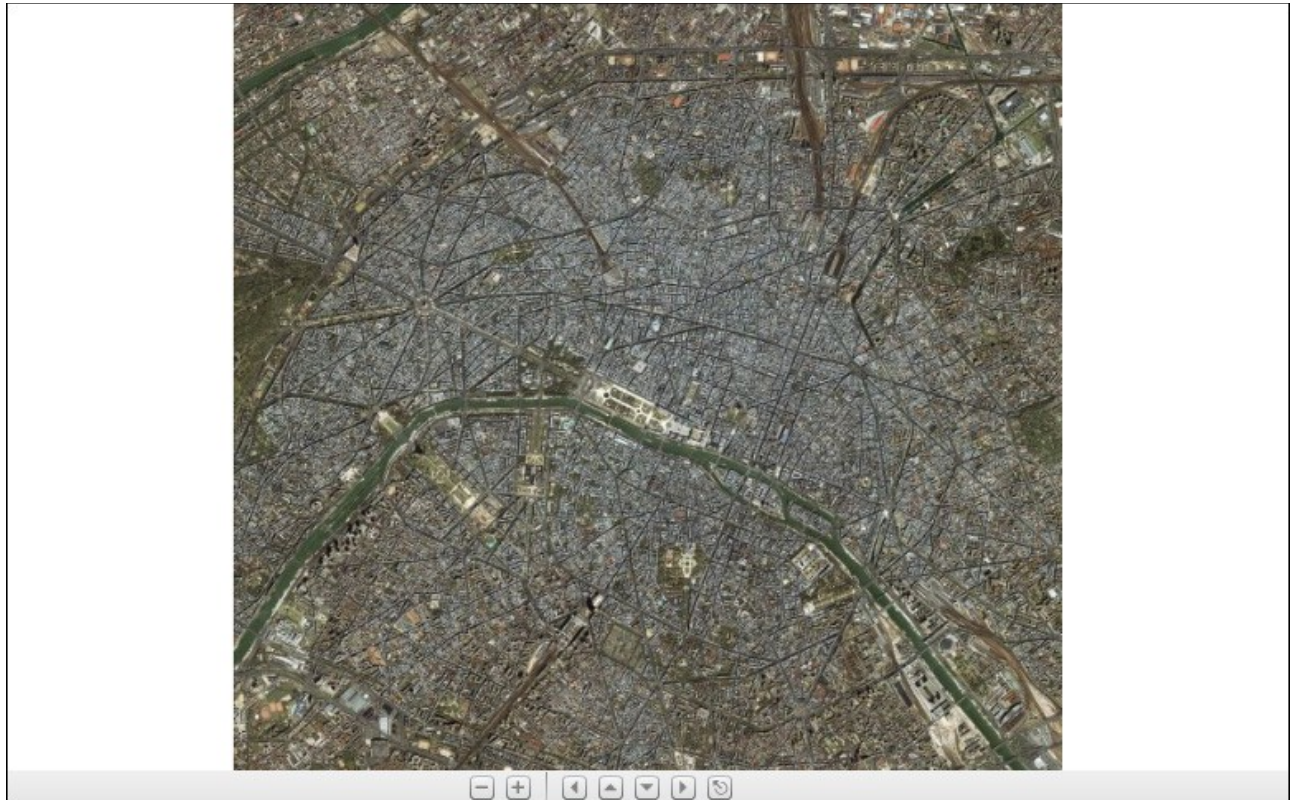
The Toolbar above is implemented as follows:

```
Z.showImage("myContainer", "ZoomifyImageExample", "zSkinPath=Assets/Skins/Dark");
```

This option can be seen in action in the example web page “7-ToolbarSkinned.htm”.

Simplifying the Viewer layout

It is also possible to hide or show many elements of the Zoomify Image Viewer Toolbar. The Zoomify logo, Minimize and Maximize buttons, the Slider, the Full Page button, and the Progress indicator can all be hidden, as in the display below.



These options are implemented using this approach:

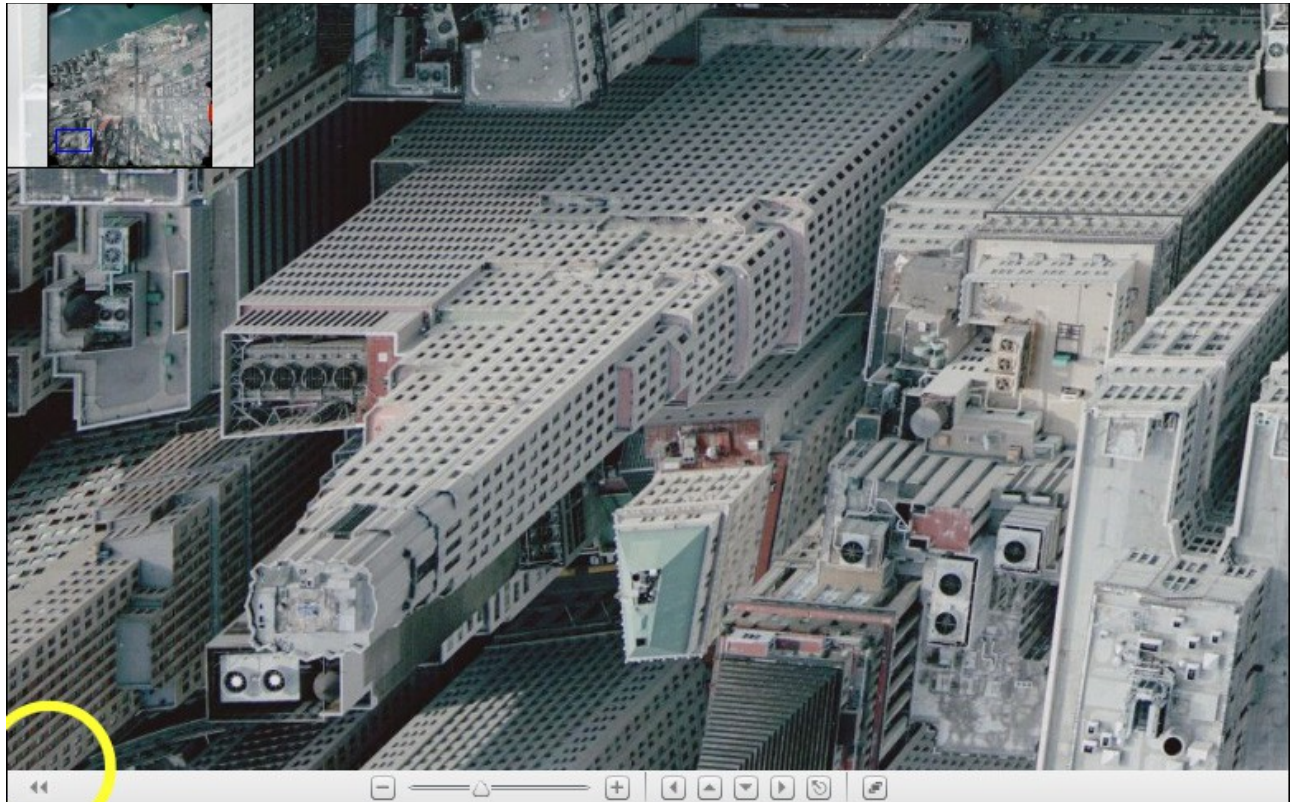
```
Z.showImage("myContainer", "ZoomifyImageExample",
  "zNavigatorVisible=0&zToolbarVisible=1&zLogoVisible=0&zSliderVisible=0
  &zFullPageVisible=0&zProgressVisible=0&zTooltipsVisible=0");
```

(Note that the above string will typically be seen as one line in your text or HTML editor. It is wrapped here due to the limits of this document.)

These options can be seen in action in the example web page “8-LayoutSimple.htm”.

Hiding the Zoomify logo

As noted above, the Zoomify logo can be hidden quite easily. In the display below the Toolbar is standard with the exception of the removal of the logo. This approach may be desired in some business or institutional contexts.



Hiding the logo only requires one small addition to the default display approach:

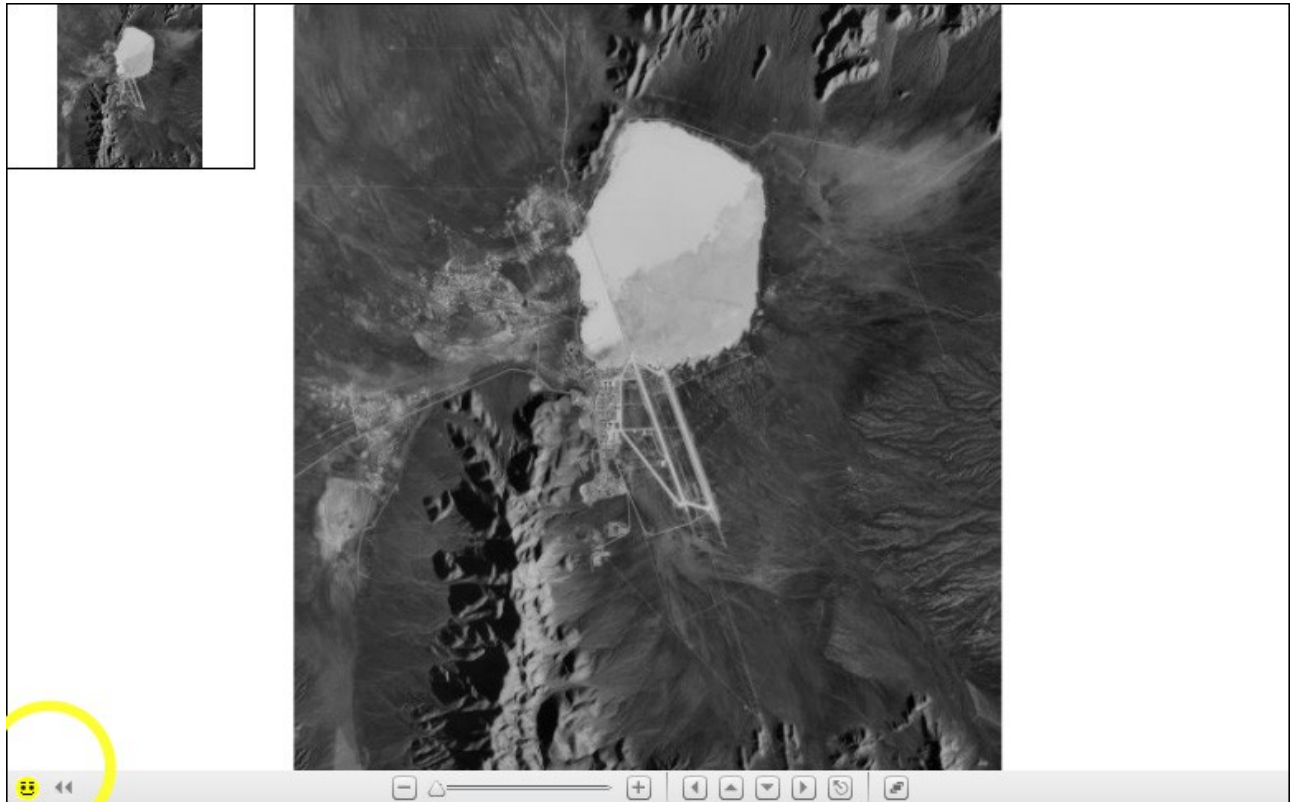
```
Z.showImage("myContainer", "ZoomifyImageExample", "zLogoVisible=0");
```

This option can be seen in action in the example web page “10-LogoOff.htm”.

(Note that the Pro and Enterprise products allow the logo to be removed – or replaced. An example follows below.)

Using a custom logo (Pro and Enterprise editions)

Some businesses or organizations will prefer to display their own logo graphic rather than the Zoomify logo. This is permitted for customers purchasing the Zoomify HTML5 Pro product and is quite simple. Just place your GIF, JPEG, or PNG graphic file on your web server and use a single parameter to tell the Zoomify Image Viewer where to find the file.



The logo replacement demonstrated above is implemented as follows:

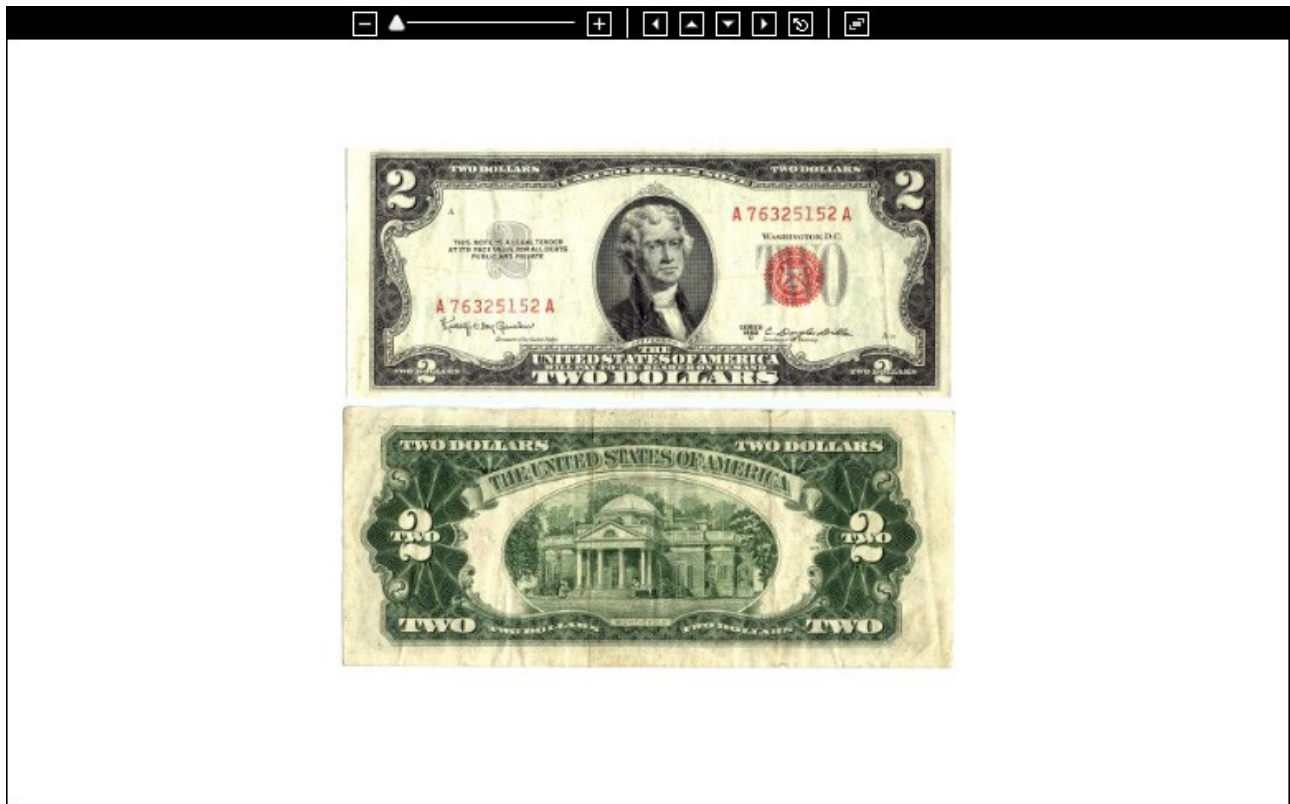
```
Z.showImage("myContainer", "ZoomifyImageExample",
            "zLogoCustomPath=Assets/LogoCustom/toolbarLogoCustom.png");
```

(Note that the above string will typically be seen as one line in your text or HTML editor. It is wrapped here due to the limits of this document.)

This useful effect can be seen in action in the example web page “11-LogoCustom.htm” (Pro and Enterprise products).

Customizing the Viewer layout (Pro and Enterprise editions)

More extensive changes to the Zoomify Image Viewer are also supported – and are also quite simple. For example, the display below places the Toolbar at the top of the Viewer, hides the Zoomify log, and sets an initial zoom value that zooms-out the Zoomify Image. It is important to note that the minimum zoom allowed is also changed – so as not to conflict with the initial zoom. Lastly, the default pan constraint of the Viewer is turned off so that the image can be freely dragged rather than being automatically forced to the center of the view. (Some users prefer to be able to pan an image freely, for example, to place a corner of the image at the center of the view area and then zoom in and out on that point.)



The above display is achieved as follows:

```
Z.showImage("myContainer", "ZoomifyImageExample",
  "zSkinPath=Assets/Skins/Dark&zInitialX=970&zInitialY=1630&zInitialZoom=50
  &zNavigatorVisible=1&zNavigatorWidth=30&zNavigatorHeight=16
  &zNavigatorLeft=860&zNavigatorTop=3&zNavigatorFit=0&zToolbarVisible=1
  &zToolbarPosition=0&zLogoVisible=0");
```

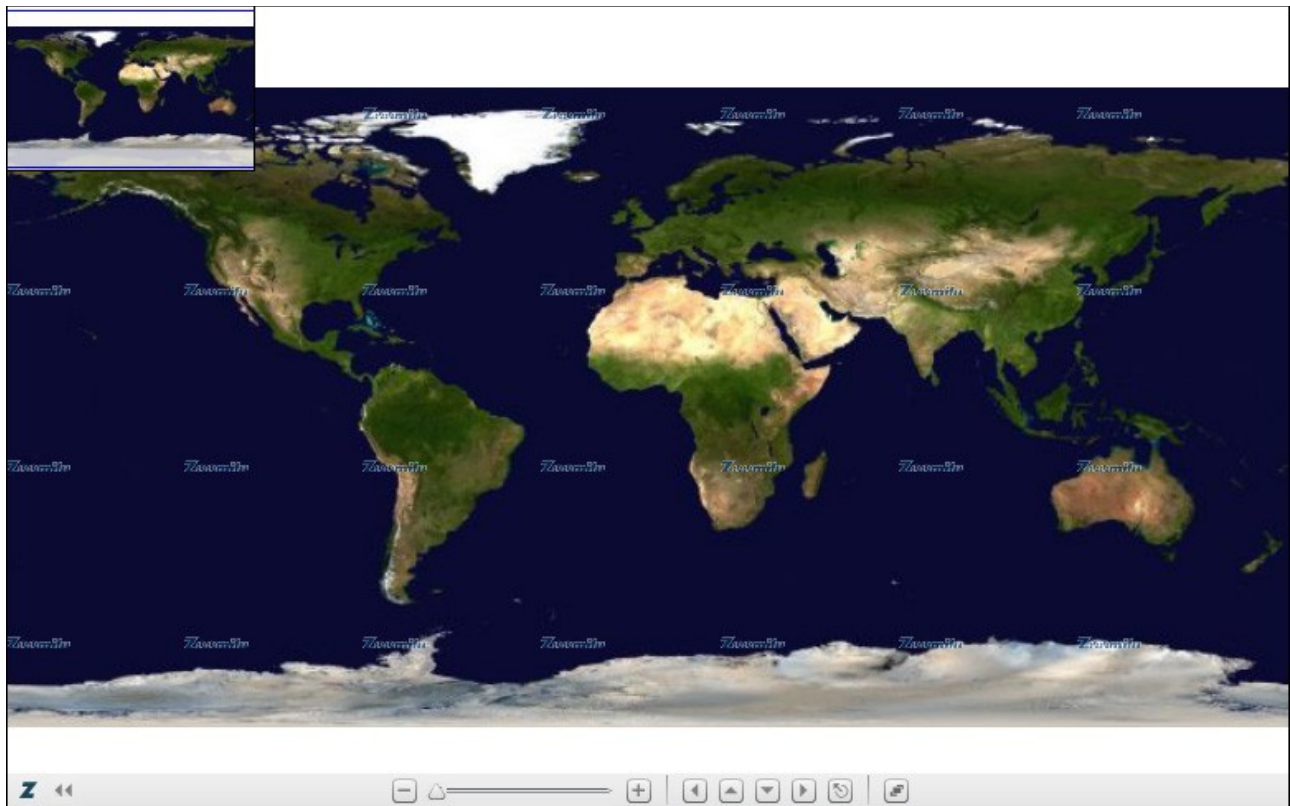
(Note that the above string will typically be seen as one line in your text or HTML editor. It is wrapped here due to the limits of this document.)

These options can be seen in action in the web page example “12-LayoutCustom.htm” (Pro and Enterprise products).

Turning on watermarks (Pro and Enterprise editions)

Protecting valuable images can be greatly assisted by watermarking their display in the Zoomify Image Viewer. Please remember, however, that there can be no guarantee that images on the Web cannot be captured and misused – the Web was designed to deliver content, not protect it. Unscrupulous website visitors can use many approaches to acquire web content – from replacing screen drivers with custom code to searching the browser cache to right-clicking the display and selecting 'Save As' – or even simply pointing a digital camera at the screen.

The Zoomify Image Viewer uses JavaScript to disable the right-click Save As menu and adds the Watermarking feature to help prevent screen captures. While this can significantly discourage site visitors from attempting to capture and misuse Zoomify Images, the only true protection for your images is to carefully select those that you absolutely don't want misused and only present parts of them, or only present them at partial resolution.



Parameter to implement watermarking:

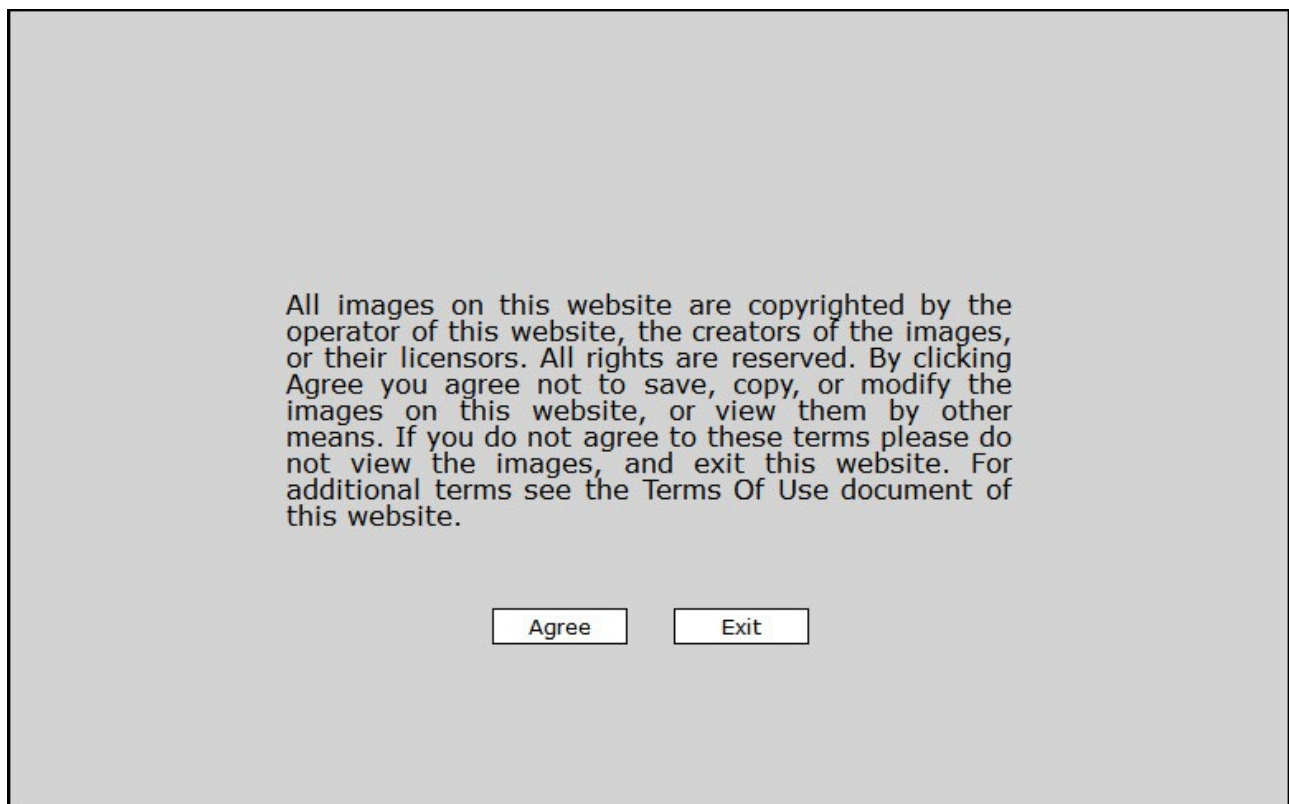
```
Z.showImage("myContainer", "ZoomifyImageExample",
    "zWatermarkPath=Assets/Watermarks/watermark.png");
```

Example web page: “22-Watermarks.htm” (Pro and Enterprise products).

Turning on the copyright alert (Pro and Enterprise editions)

Another great way to protect your images is to clearly state the allowed uses of the content on your website. Most site visitors will respect such a statement – and if any misuse does occur, being able to demonstrate that there could be no mistake about what was and was not allowed will give site managers and content owners maximum recourse under copyright laws.

The display below presents an example image copyright statement included in the Zoomify HTML5 Pro product. The XML (text) file containing this file can be found in the Assets | Copyrights folder of the product. It can be easily be edited in any text editor and should be reviewed by your organization's counsel to be sure it meets your needs.



The above display is created with the use of a single parameter:

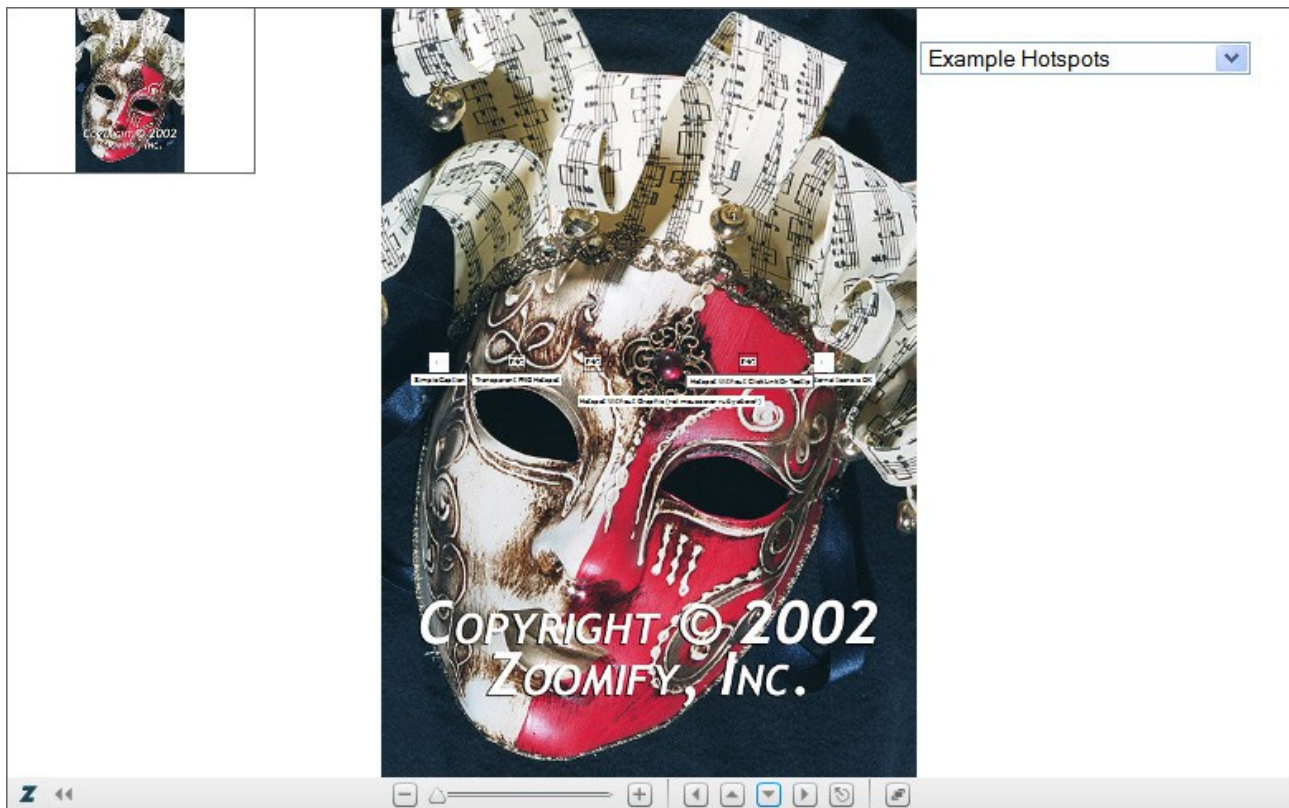
```
Z.showImage("myContainer", "ZoomifyImageExample",  
            "zCopyrightPath=Assets/Copyrights/copyrightText.xml");
```

The above option can be seen in action in the example web page "23-Copyright.htm" (Pro and Enterprise products).

Adding hotspots to an image (Pro and Enterprise editions)

The HTML5-based Zoomify Image Viewer supports *hotspots* - graphical icons that zoom-and-pan with the Zoomify Image. Hotspots can be include captions, tooltips, and click-links to a new web page. Hotspots may also be hidden until mouse-over, and can be listed in a choicelist, the title of which can be set with a simple HTML parameter.

Each of these features is optional and can be easily implemented by editing a small XML-formatted text file named hotspots.xml. This file lists the X and Y coordinates (in pixels) at which to place the hotspot icon, as well as the information needed for other features. Complete details are included in the [Zoomify Parameter List.htm](#) document in the Documentation folder of this product.



The above display is created with the use of these web page parameters:

```
Z.showImage("myContainer", "ZoomifyImageExample",
    "zHotspotPath=Assets/Hotspots&zHotspotListTitle=Example Hotspots");
```

The zHotspotPath must point to the folder in which the hotspots.xml file is located, not to the XML file itself (this differs from the design of the Flash-based Hotspot Viewer.)

The above options can be seen in action in the example web page “24-Hotspots.htm” (Pro and Enterprise products).

Using toolbar buttons in the web page (Pro and Enterprise editions)

Many web designers will prefer to integrate the Zoomify Image Viewer very closely with the look and/or functionality in their website. This is very simple to do with the Zoomify Image Viewer included in the Zoomify HTML5 Pro product because it includes direct page access to its many JavaScript functions. For example, in the display below the Zoomify Toolbar is hidden, and standard HTML buttons are placed in the web page to provide all navigation functionality. These buttons can be modified using all standard HTML options.



The Zoomify Toolbar is hidden using one simple

```
Z.showImage("myContainer", "ZoomifyImageExample", "zToolbarVisible=0");
```

An example web page implementing navigation buttons in the page itself is "31-ToolbarInPage.htm" (Pro and Enterprise products).

Note that standard HTML form instructions are used to create the in-page buttons:

```
<form name="myForm" id="myForm">
  <input type="button" name="zoomOut" id="zoomOut" value="Zoom Out"
    onMouseDown="Z.Viewport.zoom('out')" onMouseUp="Z.Viewport.zoom('stop')"
    onTouchStart="Z.Viewport.zoom('out')" onTouchEnd="Z.Viewport.zoom('stop')"/>
  ....Etc.
```

Setting the view (Pro and Enterprise editions)

Simple JavaScript commands can not only be used to control the zoom-and-pan features of the Zoomify Image Viewer – they can be used to take control of every aspect of the Viewer's functionality. For example, the `setView` command can be used to set the current coordinates of the Viewer display in order to tailor in-page navigation buttons to your image content.



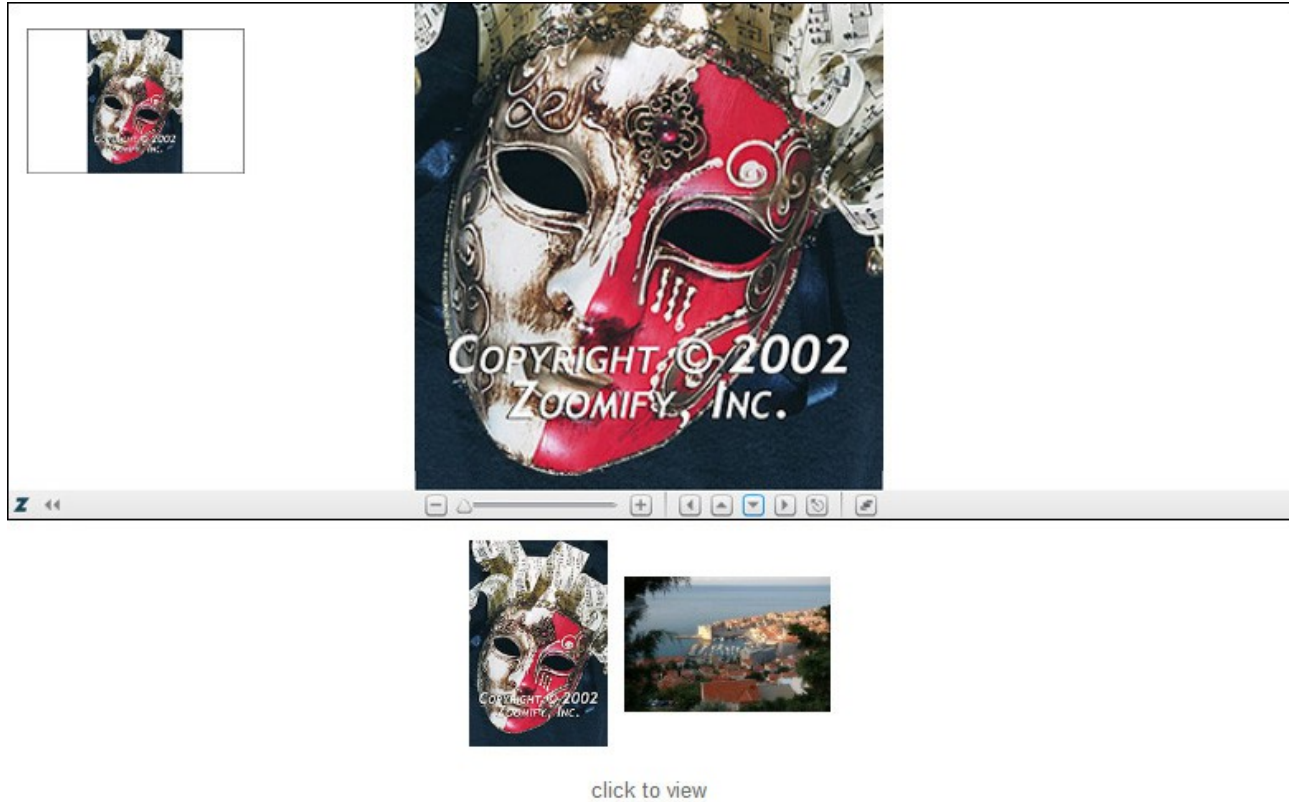
By providing the X, Y, and Zoom coordinates in each button above, a simple and intuitive means to browse points of interest within an image can be implemented quickly and efficiently.

```
<input type="button" name="harbor" id="harbor" value="Harbor"
  onMouseDown="Z.Viewport.zoomAndPanToView(1950, 1100, 100)"
  onTouchStart="Z.Viewport.zoomAndPanToView(1950, 1100, 100)" />
```

The above option can be seen in action in the example web page “33-SetView.htm” (Pro and Enterprise products).

Setting the image (Pro and Enterprise editions)

Just as it is a simple matter to change the view within an image, it is quite simple to change the image itself. By using the `setImagePath` command in a button or on a thumbnail image, a gallery display can be easily created.



The assignment below enables the left thumbnail above to change the image displayed in the Zoomify Image Viewer – note that mouse and touch events are both supported, enabling full functionality on personal computers as well as mobile devices such as Apple's iPad.

```
<a href="#" onMouseDown="Z.Viewer.setImagePath('ZoomifyImageExample')"  
  onTouchStart="Z.Viewer.setImagePath('ZoomifyImageExample')">  
  </a>
```

See example web page: “34-SetImage.htm” (Pro and Enterprise products)

Tip: the thumbnail above is pulled straight from the Zoomify Image folder, which always has a 0-0-0.jpg thumbnail file. This approach saves valuable web designer time – and leverages existing image data for maximum efficiency.

```
img src="ZoomifyImageExample/TileGroup0/0-0-0.jpg"
```

Setting the image and hotspots (Pro and Enterprise editions)

When setting the image to display it is also possible to set the hotspots to display within that image. This simply requires using the `setHotspotPath` command in addition to the `setImagePath` command in a button or on a thumbnail image.



click to view

The assignment below enables the left thumbnail above to change the hotspots displayed within the image – in addition to changing which image is displayed in the Zoomify Image Viewer. Note that mouse and touch events are both supported, enabling full functionality on personal computers as well as mobile devices such as Apple's iPad.

```
<a href="#" onMouseDown="Z.Viewer.setImagePath('ZoomifyImageExample')"  
  onMouseUp="Z.Viewer.setHotspotPath('Assets/Hotspots')"  
  onTouchStart="Z.Viewer.setImagePath('ZoomifyImageExample')"  
  onTouchEnd="Z.Viewer.setHotspotPath('Assets/Hotspots')"></a>
```

See example web page: “36-SetImageAndHotspots.htm” (Pro and Enterprise products)

Viewing in a layer (Pro and Enterprise editions)

Great web page design flexibility is possible using *CSS layers* – and the Zoomify Image Viewer fits excellently with this page design approach. The display below shows the Viewer hovering over a page of text. Note that this is not a feature of the Zoomify Image Viewer, simply a great web design option that the Viewer is completely compatible with.



Note that this approach works best when the Zoomify Toolbar and Navigator are set to appear initially (this is default behavior, but is set explicit here for clarity):

```
Z.showImage("myContainer", "ZoomifyImageExample", "zToolbarVisible=2&zNavigatorVisible=2");
```

See example web page: “39-ViewInLayer.htm” (Pro and Enterprise products).

Note that this example uses additional JavaScript to create, position, and size the Zoomify Image Viewer in a layer. For example, a `showLayer` function is used to make the layer visible and an `onclick` event handler is used to trigger the change of the layer visibility.

Web designers not comfortable with JavaScript can use the example page without any need to fully understand or edit the page, other than to change the string for the folder name “ZoomifyImageExample” to the folder name of the Zoomify Image they wish to display.

Viewing in a link (Pro and Enterprise editions)

Web designers working with large numbers of Zoomify Images can streamline their website development by using a single display page to show any image. Rather than creating a web page for each image, this approach uses just one page, linked to from any other page. The link contains the name and path of the image to display and the display page uses that information to set itself up to show the right image in the Zoomify Image Viewer. Other parameters can also be passed via the link – in fact, any and as many parameters as desired. The display below shows two links that are examples of what is possible – a simple default display of a specific image, or a customized display.

VIEW IN LINK EXAMPLE

The example links below tell a pop-up web page to display a Zoomify Image with specific parameters. This approach enables one page to be used to display any Zoomify Image on a web site, avoiding the need to design multiple display pages.

[CLICK HERE FOR DEFAULT DISPLAY](#)

[CLICK HERE FOR CUSTOM DISPLAY](#)

Note that each link above passes image, skin, and other parameters to the pop-up page which uses the information to determine what image to show as well as what skin to use and what other custom settings to apply. View the page source to see the link contents.

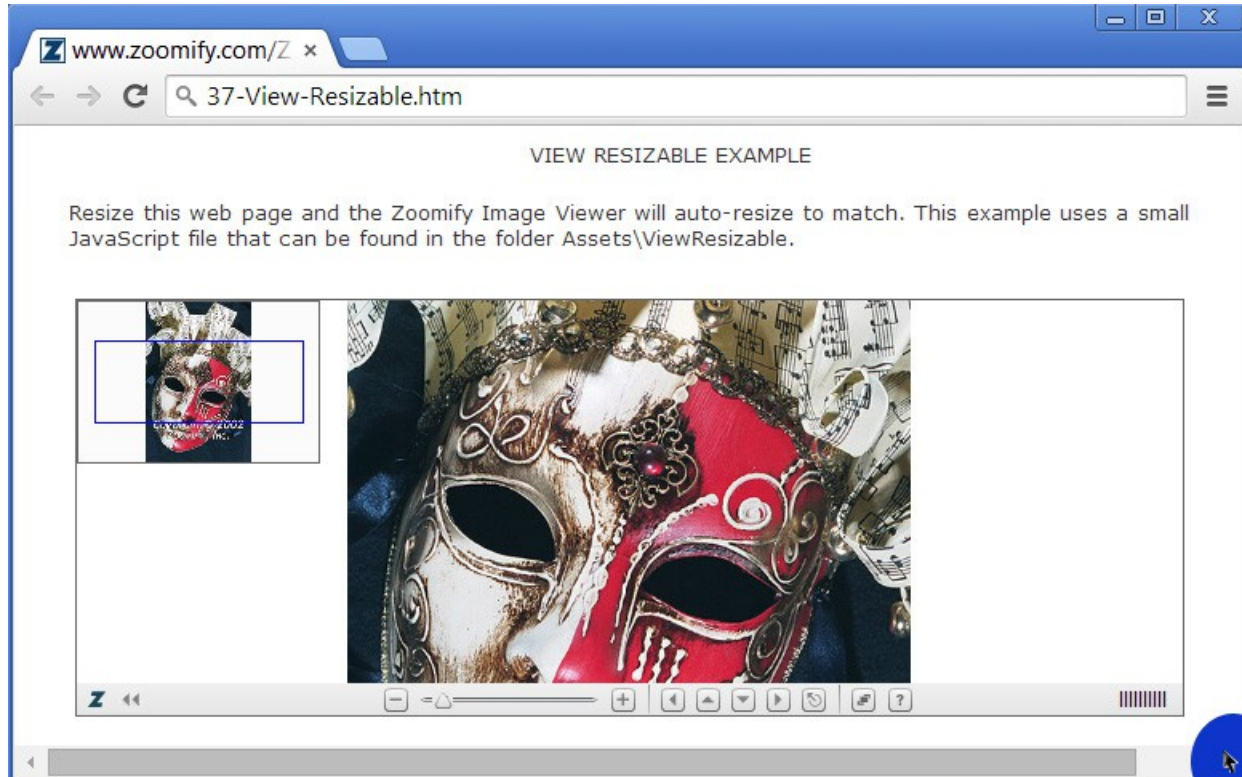
The first link in this example looks like this:

```
<a href="Assets/ViewInLink/PopUpPage.htm?zImagePath=../../ZoomifyImageExample
&zSkinPath=../../Assets/Skins/Default" target="_blank">
CLICK HERE FOR DEFAULT DISPLAY</a>
```

Note that the '?' character ends the name of the web page to open and begins the parameters to pass to that page. Place a link like this in any web and replace the image path above to display any Zoomify Image. See this approach in action in the example web page "40-ViewInLink.htm". The PopUpPage.htm web page can be found in the Assets/ViewInLink folder (Pro and Enterprise products).

Viewing in a resizable web page (Pro and Enterprise editions)

Another helpful behavior web designers may desire is included in the Zoomify Image Viewer: an automatically resizing display. Some website implementations will benefit from the ability to resize a browser window and have the displayed web page automatically resize to fit the new window size. If the Zoomify Image Viewer is placed in such a context it can automatically resize as well. Note that this example is an example of Viewer compatibility with a possible JavaScript implementation, not a demonstration of a Viewer-specific feature. In this (simple) example, the Viewer reinitializes (restarts) and returns to its initial view.



The key functionality in this example is not in the specific insertion of the Zoomify Image Viewer in the example web page but in a separate script file “Assets / ViewResizable / sizeViewerToPage.js” called by the example page. No special Viewer parameters are required.

This approach can be seen in the example web page “41-ViewResizable.htm” (Pro and Enterprise products).

Viewing across domains (Pro and Enterprise editions)

The use of JavaScript to access files is tightly constrained to ensure web site and local system security. Because of this, the Zoomify Image Viewer cannot be used to display a Zoomify Image that is stored on a separate domain from the ZoomifyImageViewer.js file itself. This “same origin” policy is common to many web scripting languages. The term "origin" is defined using the domain name, application layer protocol, and port number of the HTML document running the script. Access is allowed only if all of these values are exactly the same. (These constraints apply whether using relative or absolute paths.)

For example, this Zoomify Image Viewer

`http://www.myDomain.com/folder/ZoomifyImageViewer.js`

can access Zoomify Image folders here:

`http://www.myDomain.com/folder`

`http://www.myDomain.com/folder/subfolder`

`http://www.myDomain.com/`

but not here (different host or protocol):

`http://www.otherDomain.com`

`http://myDomain.com/folder/` (non-exact match)

`http://v2.www.myDomain.com/folder/` (non-exact match)

`http://en.myDomain.com/folder/` (different subdomain)

`http://www.myDomain.com:81/folder/` (different port)

`https://www.myDomain.com/folder/` (different protocol)

The Zoomify Image Viewer does provide a partial workaround: the `zImageProperties` parameter allows the image properties of the Zoomify Image to be included in the initial load of the Viewer. These can be copied directly from the `ImageProperties.xml` file in the Zoomify Image folder (the included double quotes must be changed to single quotes). This approach removes the need to dynamically load this information. The implementation in the example web page “42-ViewCrossDomain.htm” (Pro and Enterprise products) is as follows:

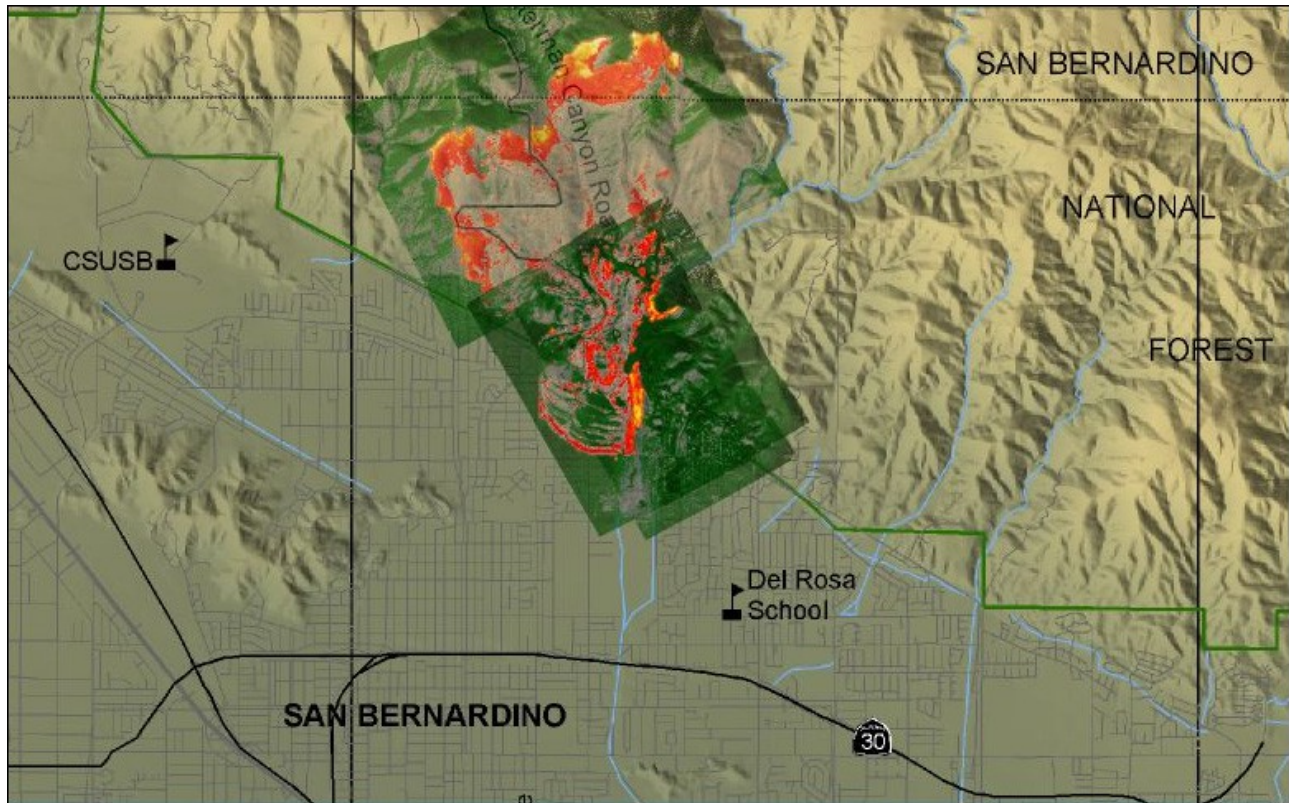
```
<script type="text/javascript"> Z.showImage("myContainer", "ZoomifyImageExample",
"zImageProperties=<IMAGE_PROPERTIES WIDTH='2080' HEIGHT='3120' NUMTILES='169'
NUMIMAGES='1' VERSION='1.8' TILESIZE='256' />"); </script>
```

Note 1: HTML5 provides a method to support extended access - the `postMessage` interface - which is available in recent browsers. JSONP and cross-origin resource sharing can also be used to enable AJAX-like calls across domains. These options are being explored by Zoomify to potentially provide simpler, safe, cross-domain access in future releases.

Note 2: local tests will prevent loading access using the “..” 'up-one-level' term in a relative path when testing in some browsers. Use of this term on a web server will not be blocked.

Turning interactivity off (Pro and Enterprise editions)

It is occasionally useful to display a specific view of an image – zoomed and panned to a particular location in the image. In these circumstances it can be helpful to disable the navigation features of the Viewer. In this case the Zoomify Image Viewer is used to display one or more views of an image – rather than to allow the site visitor to select any view they may prefer. In the display below the Zoomify Toolbar and Navigator have been hidden, and the click-zoom feature and keyboard support have been disabled.



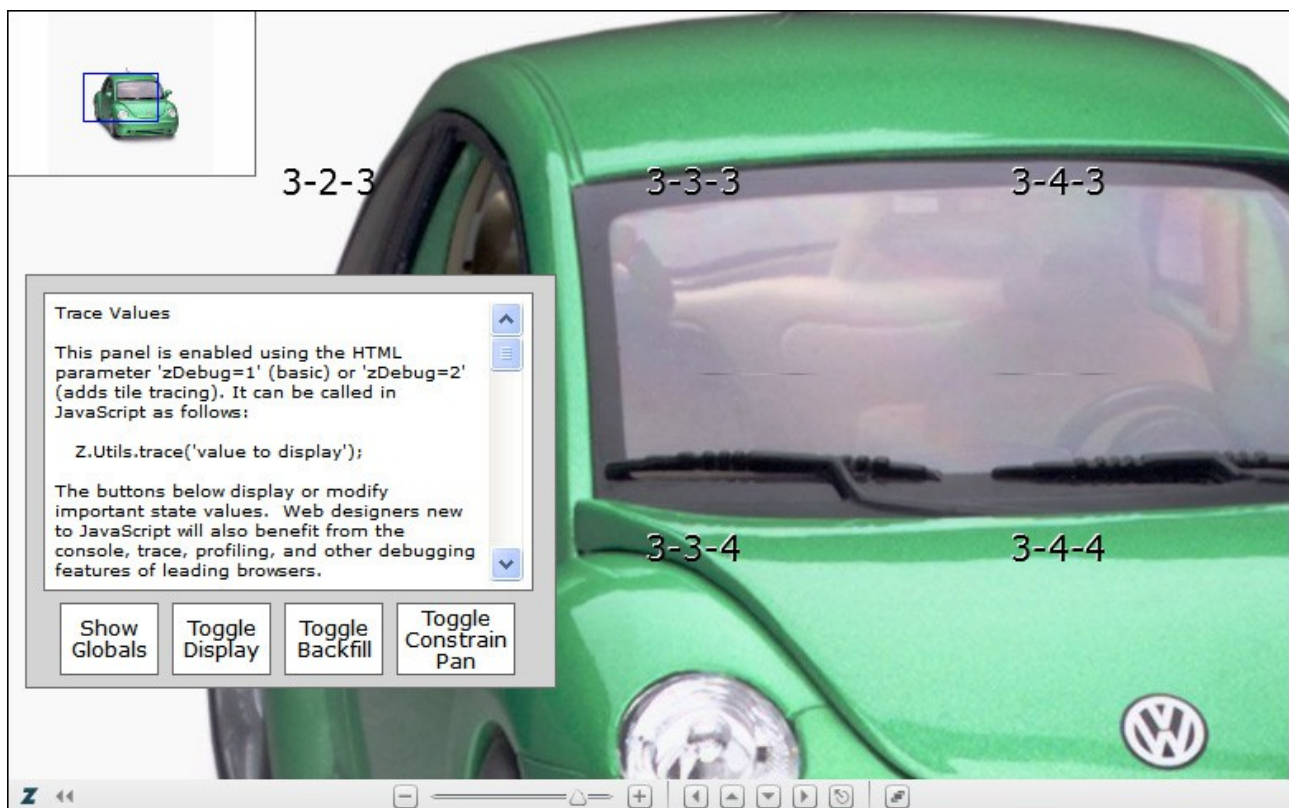
This approach is implemented as follows:

```
Z.showImage("myContainer", "ZoomifyImageExample",
  "zNavigatorVisible=0&zToolbarVisible=0&zClickZoom=0&zClickPan=0
  &zMousePan=0&zKeys=0&zInitialX=970&zInitialY=1630&zInitialZoom=50");
```

This approach can be seen in action in the example web page “43-InteractivityOff.htm” (Pro and Enterprise products).

Turning debugging on (Pro and Enterprise editions)

The Zoomify Image Viewer included in the Zoomify HTML5 Pro product includes support for a debugging dialog that provides helpful functionality. Values can be traced using the simple command “Z.Utils.trace('value to display');” This feature complements the “console.log”, “window.status”, and other debugging features of popular browsers. The debugging display also includes a 'Show Globals' button that will display an alert listing the current values of the key variables of the Zoomify Image Viewer. Additionally, the Viewer debugging dialog presents buttons that enable developers to easily hide or show the main tile display layer and/or the background tiles layer, and to toggle the default pan constraint on and off. These features address needs specific to multi-resolution imaging and help answer questions that arise when customizing the Zoomify Image Viewer.



The debugging features of the Zoomify Image Viewer can be enabled with a single parameter:

```
Z.showImage("myContainer", "ZoomifyImageExample", "zDebug=2");
```

Use `zDebug=1` to enable the debug panel (simple, fast option), use `zDebug=2` to enable the panel with tile loading calculations and tile name displays and tile tracing (slow, detailed option), and use `zDebug=3` for tile loading calculations without tracing the loading of all tiles and without labeling all tiles (moderate option, optimal for watching tile calculations). See this feature in action in the example web page “45-DebugOptions.htm” (Pro and Enterprise products).

List of all HTML Express & Pro parameters (Pro and Enterprise editions)

The Zoomify Image Viewer supports many HTML parameters, as demonstrated in the many examples discussed above. The complete list of these parameters along with their value ranges and default, can be found in the document “Zoomify Parameter List.htm” in the Documentation folder of the Zoomify HTML5, Zoomify HTML5 Pro, and Zoomify HTML5 Enterprise products. The document clearly delineates the parameters supported by the Zoomify Image Viewer included in each product edition.

Some web designers will appreciate seeing the many parameters as they appear when applied in a web page. They can all be seen together, therefore, in one example page: “46-AllParameters-Express&Pro.htm” (Pro and Enterprise products).

Please note that many of these parameters would rarely, if ever, be used together, as their purposes vary greatly. To place them all in a single web page it is therefore simplest to assign many their default values. This avoids creating a web page example with an odd combination of layout or behavior settings.

Additional note: The Zoomify Image Viewer's showImage function supports inclusion of optional parameters - such as zInitialX, zInitialY, and zInitialZoom - in the form of a delimited string as demonstrated in previously referenced examples. The showImage function also supports inclusion of optional parameters in the form of an object.

This approach is useful where the showImage function is called in a context where the '&' character cannot be used to join multiple parameters together, or when using the zOnReady parameter to assign a function to execute upon initialization of the Viewer.

See this feature in action in the example web page “47-Parameters-InObjectForm.htm” (Pro and Enterprise products), and see the Parameter List in the Documentation folder for more information.

Viewing annotations in an image (Enterprise edition)

Zoomify's Image Viewer can be used to display annotations in a Zoomify Image. These annotations are enhanced hotspots that include – in addition to icons, captions, and tooltips – point of interest groupings, notes, and more. No Flash is required.



Annotations are stored in a small XML or JSON formatted text file named annotations.xml or annotations.json. This includes the coordinates, note, and other data needed for the annotations display within the Zoomify Image as well as for the multiple choicelists of the Annotations panel: Points of Interest, Notes, and Labels (hotspots).

A single parameter is used to implement annotation viewing. Detailed information on annotations and the annotations.xml data format can be found in the Zoomify Enterprise product documentation.

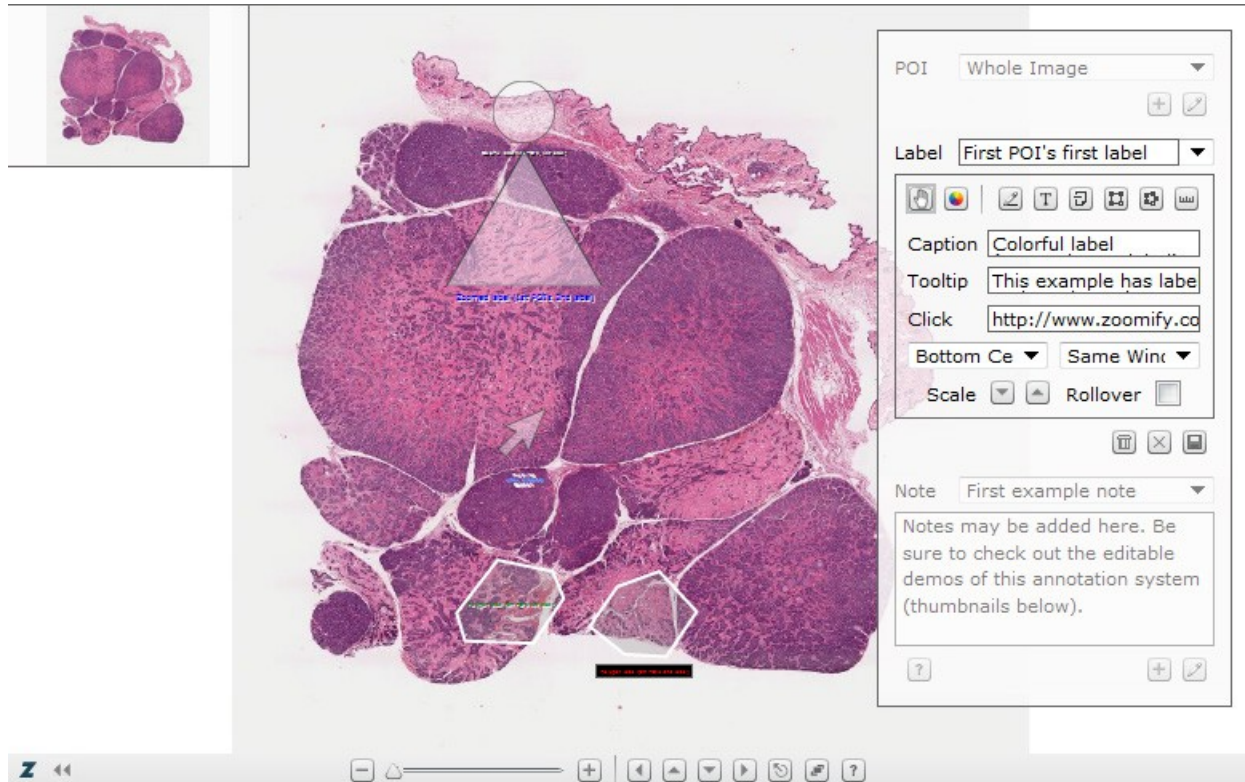
```
<script type="text/javascript"> Z.showImage("myContainer", "ZoomifyImageExample",
"zAnnotationPath=Assets/Annotations"); </script>
```

See the example web pages “49-Annotations.htm”, “57-Editing-MarkupMode.htm”, “58-Editing-AnnotationMode.htm”, and more (Enterprise product only).

Additionally see the examples “62-Storage-JSON-AnnotationsInFile.htm”, “63-Storage-JSON-AnnotationsInFileAtPath.htm”, and “64-Storage-JSON-AnnotationsInObject.htm” using JSON formatted storage.

Editing annotations (Enterprise edition)

Zoomify's Image Viewer includes support for annotation editing. The ability to create and edit Points of Interest, Labels, and Notes enables digital pathology, defense, and other customers to easily notate areas and items of interest within a Zoomify Image.



For example, clicking the Edit button (pencil icon) below the Label area of the Annotation Panel enables the label editing features of the Panel as displayed above. Click one of the Label type buttons and then click in the image to create a text or icon label. Click-drag in the image to create a freehand drawing or rectangular label. Click more than once to create a polygon or measurement label. Alt-click the first control point of a polygon to close it, or Alt-click in the image to leave it open (option-click on Macintosh). Alt-click-drag any existing label to reposition it.

The Label choicelist of the Annotation Panel can be used to select any existing label for editing. Clicking Save 'posts' updated XML (or JSON) to the webserver.

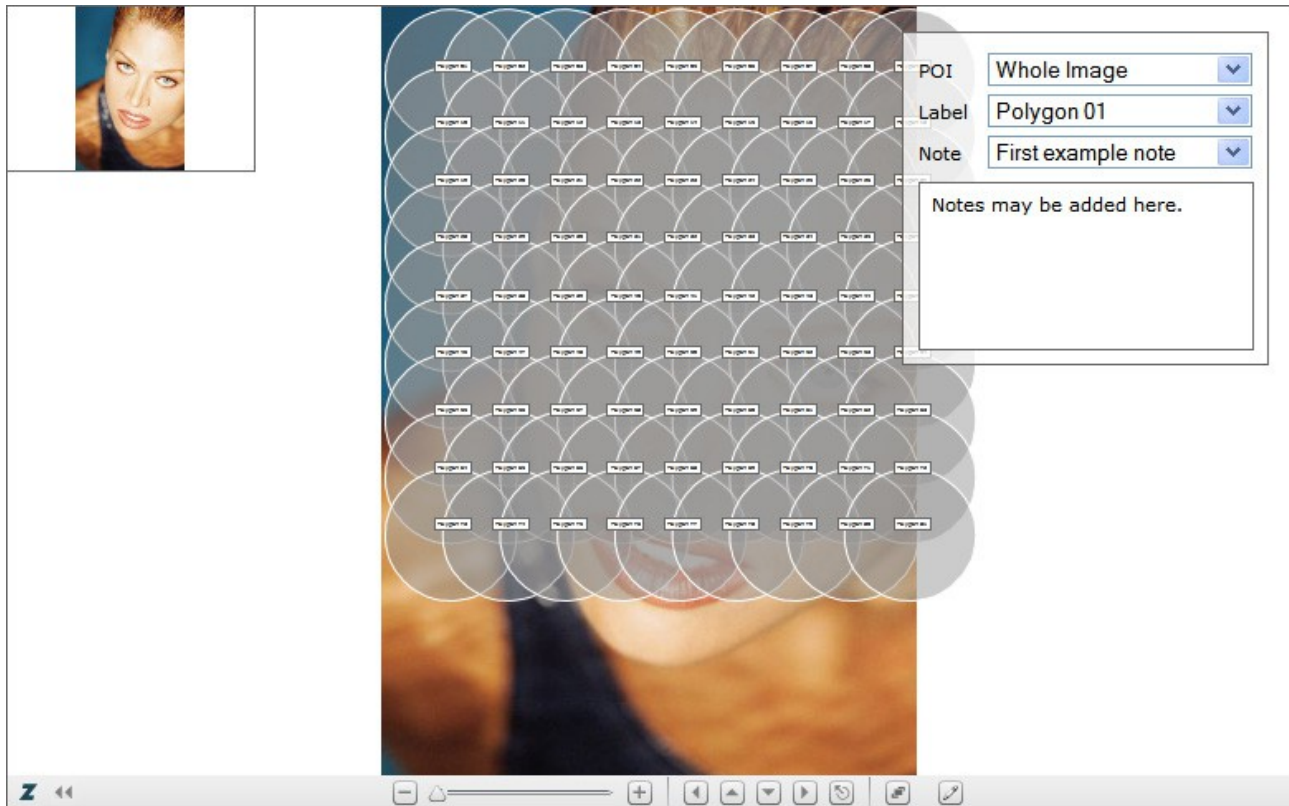
Three HTML parameters are used to enter Edit mode:

```
<script type="text/javascript"> Z.showImage("myContainer", "ZoomifyImageExample",
    "zAnnotationPath=Assets/Annotations/Enterprise&zEditMode=1
    &zSaveHandlerPath=upload.asp"); </script>
```

See the example web page “56-Editing-Polygons.htm” (Enterprise product only).

Testing polygon drawing speed (Enterprise edition)

Zoomify's Annotation and Hotspot support depend upon the HTML5 Canvas feature to enable the polygon drawing and editing features. Most users will require five to ten hotspots (labels) in an image and, if using polygons, perhaps 5 to 20 control points per polygon. Some users, however, will push these features much further. Support in popular browsers will allow inclusion of many more hotspots and many more polygons with many more control points per polygon.



The example above displays 81 polygons, each consisting of 129 control points, comprising 10,449 points in total. On common browsers new enough to support the HTML5 Canvas feature, initial drawing is rapid.

Selecting an item in the Label choicelist to zoom-in on a polygon, or navigating within the image using the Toolbar buttons or keys provides an adequate experience. Note that only the polygons within the current view area are drawn to optimize performance.

Clicking the Edit button to enable the current polygon's control points for dragging, provides very fast editing interaction. Only the current subset of polygons and control points is interacted with. Every effort has been made to streamline drawing to improve the viewing and editing experience.

See the example web page [“52-Annotations-SpeedTest.htm”](#) (Enterprise product only).

Viewing a Zoomify Image stored as one file

The Zoomify Image Format (ZIF) allows storage of high-resolution, zoomable images megabytes or gigabytes in size in one single file. ZIF files can be viewed using the Zoomify Image Viewer with no need for any special server side setup (no Java 'servlet' as required with the PFF format) and no special client technology (no proprietary plugin to download). For customers and partners with large collections of large images, this provides simple, fast viewing with convenient, efficient content management. Note that the Zoomify Converter can easily convert Zoomify Image Folders and PFF files to the ZIF format.

Note that this approach does not require a 'helper application' on the web server, as single-file storage with a '.PFF' file did in prior versions of Zoomify products. JavaScript is able to request the image tiles from within a single file from the webserver (byte-range requests) in recent versions of all popular browsers.

When hosting ZIF or PFF images a Windows server using Internet Information Services (IIS) Manager it may be necessary to set a 'MIME type'. Simply open IIS from the Start menu, click the MimeTypes button, click Add, then enter File name extension: .zif and MIME type: image/zif.

Detailed information on the single-file storage option can be found in the product folder 'Developer Resources | One File Storage Options'.

See the example web page “9-Storage-ZIF-OneFile.htm” (all products), “30-Storage-ZIF-AndHotspots.htm” (Pro and Enterprise products), and “59-Storage-ZIF-AndAnnotations.htm” and “61-Storage-PFF-OneFile.htm” (Enterprise product only).

Local Viewing

The ZIF and PFF single-file storage alternatives will not work on a local machine. They must be used over the Internet, off of a web site. This is because of the standard security protections of web browsers.

Additionally, local viewing may be blocked browsers other than Firefox regardless of the storage approach implemented. Recent versions of Chrome and Safari as well as Internet Explorer 11 all block local loading of certain types of dynamic content. As of the release date of this product, Mozilla Firefox does not. It is possible in some cases to use browser setting to allow local access.

Setting up a 'localhost' test server simulates access to a web server over a network. This can serve as a workaround for the local testing blocks described above. For detailed information see the folder Developer Resources | Testing Tools | Local Host.

Viewing, creating, editing, and saving annotations

The Zoomify Image Viewer displays Zoomify Images with annotation graphics, captions, and tooltips presented directly in the Zoomify Image and with annotation Points Of Interest, Notes and Label details displayed in the Annotations Panel. All data for annotations is stored in an annotations XML file. In the many examples of this product this file is typically named “annotations.xml” or a variation of this name. This annotations XML file includes parameters for each type of information to be stored.

Viewing annotations

The Zoomify Annotation Viewer makes viewing annotations a seamlessly integrated part of the experience of viewing a Zoomify Image. When an annotations XML file parameter is provided in the web page HTML or XML along with the image path parameter, the Annotations Panel is automatically populated and the specified graphics are placed within the Zoomify Image as appropriate.

Further, if a parameter is provided for a slides XML file, then multiple Zoomify Images will be listed on the Menu Panel and each may have its own annotations. Using the Menu to open a different Zoomify Image automatically clears and repopulates the Annotations Panel and the new annotation graphics are added within the new Zoomify Image.

In addition, if the save handler and edit mode parameters are also provided the annotations become editable (the Annotations Panel's Add and Edit buttons appear).

Note: if the save handler specified is not actually present or is in some way faulty, the annotations will appear editable within the Annotations Panel and the view of the Zoomify Image, however, when a save is attempted the annotation edits will only be saved in memory, will only be seen by the user doing the editing, and will only be saved for the duration of that user's browser session.

The Annotation Viewer provides display, navigation, and editing fields, buttons, and choicelists for every aspect of an annotation. This includes graphics, captions, tooltips, notes, positioning, color settings, and more. Because all features are supported by intuitively labeled interface elements users are able to immediately make use of annotations to explore and understand the subject Zoomify Image. No special instruction or documentation should be necessary for users beyond that related to the policies or practices of a specific organization.

Creating and editing annotations

Creating an annotations XML file can be accomplished in several ways: manually in a text editor, automatically via server-side script or database macro, or visually within the Zoomify Annotation Viewer itself when it is displayed in a web page in Edit Mode.

Manual creation and editing using a text editor

An annotations XML file can be created directly in a text editor because it is simply text with XML 'tags' – as seen in the above example. This is especially easy because one can take advantage of the many defaults and create a simple minimal XML file. See the Parameters List document in the Documentation folder for the minimum possible tags in an annotations XML file. When manually creating the desired XML, create an initial POI Label, and/or Note, then simply duplicate that entry and modify the Name , X and Y coordinates, Caption, and Tooltip.

Automatic creation using a database

Customers working with a database backend can easily output XML using the scripting language of the database software in use. The script can easily wrap each set of field values in the appropriate tags.

In this way, website traffic impacts can be limited the web server, which would respond to initial XML file load requests as any site visitor's browser loads a page including the Zoomify Annotation Viewer. The database is then utilized as the ultimate data repository.

Visual creation and editing using the Annotation Viewer

The Zoomify Image Viewer allows convenient, intuitive data entry. The Add and Edit buttons presented in Edit Mode permit rapid creation of Points of Interest, Notes, and Labels. In addition, the new Markup Mode allows rapid creation of labels of any type simply by clicking a button then clicking in the image. Note that the Viewer does not implement multi-use database record locking. If the website providing access via the Annotation Viewer to a Zoomify Image and its associated annotation data does not prevent multiple users from accessing the same image, more than one user can load the same data. In this event, it is the last call to Save rather than first call to Add or Edit, that determines the final overwrite of the annotation XML file.

Saving annotations using a 'save handler'

When the Add or Edit button is clicked within the Annotation Viewer, several new buttons are presented: Cancel, Delete, and Save. The Save button permits 'posting' – overwriting of the old annotations XML file with new additions and modifications. This is accomplished using whatever 'save handler' has been specified in the HTML or XML parameters of the calling web page.

The relevant parameters are “zoomifyXMLSaveHandlerPath” in HTML and “XMLSAVEHANDLERPATH” in XML. For an HTML example, see the example file “ZAS-SetMode-Edit.htm” in the “Edit or View-Set Mode”. For an XML example see the “ZASXMLParams-OneImage-Edit.xml” example file in the “Parameters-All XML”. Both examples are subfolders within the “Zoomify Feature Examples” folder.

The save handler itself can take many different forms, a PHP, JSP, ASP or other script file, or even a Java servlet or other server-side helper application. With respect to script-based posting, several example implementations are provided (“upload.asp”, “upload.php”, etc.). These can be found in the “Annotation - Posting Examples” subfolder of the “Website Publishing” folder. This form of posting support can be extremely simple to set up – no Servlet to set up *and* no servlet container to set up. In addition, because these scripts are totally focused on posting and don't need to include code for PFF tile handling, they are extremely simple – the simplest is just 20 lines in length while the most involved is around 100 lines. They are therefore quite compact and, as an added benefit, easy to customize. Further, they can be ported to another scripting language quickly.

With respect to a servlet, the Zoomify Enterprise product includes the Zoomify Servlet. Using the Zoomify Servlet for posting is convenient if it is already being used for viewing Zoomify Images stored in PFF single-file format. Under any other circumstances it is not recommended due to the greater setup requirements as compared to any of the script options. For setup instructions see the supplemental information in the Zoomify website Support FAQ at <http://www.zoomify.com/support.htm>.

Updating annotation data from prior versions

The annotations XML data saved by the HTML5-based and Flash-based versions of the Viewer differ due to the complexity and optimizations supported. The current Annotation Viewer is able to read older XML. It will invoke defaults for all values the legacy XML does not include. If loaded in Edit mode and saved, the XML will be saved in the current format - overwriting the pre-existing version of the XML.

*See the section 'Updating Annotation XML Creating In Flash' in the READ ME FIRST.txt file in the main product folder for information about **automatically** updating annotations.xml files created in Flash.*

Chapter V. Advanced Topics

Customizing the Zoomify Image Viewer JavaScript

The Zoomify Image Viewer provides substantial functionality that addresses the core requirements of important imaging applications. It is expected, however, that JavaScript developers will identify additional functionality and even entirely new imaging applications. The Zoomify License Agreement permits source code modifications - please see the Agreement in the Documentation folder or on the Zoomify website for complete details (www.zoomify.com/zoomifyLicenseAgreement.htm)

Code organization

To support development by JavaScript developers, Zoomify HTML5 Pro includes complete JavaScript source code in a single file: `ZoomifyImageViewer.js`. As JavaScript developers will expect, functions are organized in clear component groups: those related to the Viewer as a whole, those relevant to the Viewport, the Toolbar, and the Navigator, functions serving Networking needs, and general Utility functions useful for working with the DOM to identify element dimensions and positions, and perform other basic tasks in a manner compatible with many versions of many browsers.

The code groupings of the Zoomify Image Viewer are demarcated by comment banners:

```
//:.....
//:.....: INIT FUNCTIONS :.....
//:.....
```

Groupings include the following: Initialization, `ZoomifyImageViewer`, `ZoomifyViewport`, `ZoomifyToolbar`, `ZoomifyNavigator`, `NetConnector`, and `Utils`.

Special additions or implementation issues are be called out as well, as follows:

```
// DEV NOTE: note text here...
```

Code documentation

The JavaScript source file `ZoomifyImageViewer.js`, included in the Zoomify HTML5 Pro and Enterprise products, is heavily commented. Note that all standard 'minification' tools remove such comments to ensure they have no impact on download speeds during actual use. The Pro and Enterprise products also include the Zoomify Functions List – a listing of all functions that make-up the Zoomify Image Viewer complete with detailed descriptions.

Variable and function names

In an effort to reduce the need for narrative comments, variable and function names are descriptive and, in most cases, self-explanatory.

Errors, strings, defaults, and resourcing

The Zoomify Image Viewer's error messages, Toolbar tooltip strings, and default values are resourced for easy modification and/or international localization in a single function in the Utils code group that is named "getResource". Similarly, handling of XML parameters has been centralized in the "setParameters" function.

Zoomify is dedicated to making the use of high-quality images commonplace worldwide. Accordingly, every effort has been made to support efficient translation of the Zoomify JavaScript codebase, documentation, and online materials. As additional Zoomify and partner-delivered solutions become available notice will be provided on the Zoomify website at <http://www.zoomify.com>.

Publishing a modified Zoomify Image Viewer

Prior to placing the ZoomifyImageViewer.js file on a public web site - whether modified or not - it should be 'minified' - that is, it should be 'compiled' to remove all white space, comments, unused functions (if any), and other text that is not required for browsers to run the JavaScript. Additionally - depending on the level of compression desired - compilation may shorten or replace variable and/or function names and it may even modify the code to make it as concise as possible.

Minifying reduces the size of the Viewer JS file significantly - almost 60% - allowing for faster downloads. The whitespace, comments, and long variable and function names can be helpful when editing the JavaScript but they are unnecessary when using it on a website. Minification by reputable compilers has no impact on how the Viewer functions.

Zoomify uses the Google Closure Compiler to minify the Zoomify Image Viewer. Closure can be found here: <http://closure-compiler.appspot.com/home>. Zoomify uses the 'Simple' setting, rather than 'Whitespace Only' or 'Advanced'.

Minifying can also provide some small measure of protection for source code by making it less convenient to download and use without purchasing a license. While unscrupulous web developers can work around this to convert the minified code to an editable form, they cannot do so without intentionally violating the copyright and use rights retained by Zoomify.

For this reason, please always use the minified form of the ZoomifyImageViewer.js file on your production web sites, and if you customize the full source version in this folder to suit your own needs, please minify it prior to using it on your web site. Your effort helps us preserve the value of the product you have purchased and thereby assists us in continued development and support of Zoomify HTML5.

Please note: Zoomify also 'obfuscates' the ZoomifyImageViewer.js file prior to minifying it. This means that Zoomify performs batch search and replace operations on function and variable names in the JavaScript code in order to provide additional protection of the value that our customers pay for. For this reason, your minified ZoomifyImageViewer.js file will look different than the obfuscated and minified ZoomifyImageViewer.js file included in the product.

Current information about minification web services can be found in the READ ME FIRST file in the folder Developer Resources | Viewer Source in the Zoomify HTML5 Pro and Enterprise products.

Thumbnail support

The Zoomify solution is completely compatible with common website designs involving a simple thumbnail that links to a pop-up providing a more detailed view.

To implement this site design using any Zoomify Image folder simply refer to the first JPEG tile within the image folder. This file will be named "0-0-0.jpg" and will be located in the folder "TileGroup0". It will be between 128 and 256 pixels in height and width - a common scale for website thumbnails. A standard JPEG reference to this file can easily be embedded within any HTML page. For example:

```
<img src = "../imageFolderName/TileGroup0/0-0-0.jpg" border="0">
```

Additional resources

The Zoomify HTML5 products include the Zoomify Parameters List document which details all parameters supported by the Zoomify Image Viewer version available in each product.

The Zoomify HTML5 Pro & Enterprise products additionally include the Zoomify Functions List – a listing with descriptions for all functions that make-up the Zoomify Image Viewer.

Finally, the Zoomify Support page provides an extensive Frequently Asked Questions listing with the latest information from Zoomify and our customers and partners. It is available on the Zoomify website: <http://www.zoomify.com/support.htm>.

Chapter VI. Troubleshooting

Troubleshooting image conversion

1. If you receive an error message or the Zoomify Converter unexpectedly quits, you may wish to check the following:
 - a. Does the directory you're attempting to write the Zoomify image to (the target directory) have enough disk space? Generally the Zoomify Converter requires two to three times the size of the source image for temporary files when encoding an image. This disk space is freed up after successful conversions.
 - b. Is the target directory writeable (i.e., not a CD or a directory / disk with limited write permissions). If so, select a target directory that is writeable before proceeding.
2. If conversion is slower than you expected:
 - a. Are you running the Zoomify on an older OS such as OS 9 or Windows 95, 98, ME, 2000, or Vista? We strongly recommend that you run the Zoomify Converter on OS X or Windows XP, 7 or 8. (Versions supporting Unix/Linux are available separately.)
 - b. Are you trying to convert an image that is not native to the Zoomify Converter? The Converter supports TIFF, BMP, JPEG, and PNG files natively.²
 - c. Is your source image on a network drive? We recommend that you copy all images you'd like to encode to a local drive.
 - d. Is your target directory a network drive? As the Zoomify encoding process is disk-intensive, we recommend that you convert on a local drive.
 - e. Do you have enough memory?
3. If you still experience problems image conversion, please send an email to support@zoomify.com with details about your environment, including OS, amount of free disk space, available RAM, Zoomify product and version number, the source image format and size, and any other information you believe to be relevant. If we determine that the issue is related to the image itself and not the environment, we may request the image for testing purposes.³

² For optimal results working with source images in an uncompressed (BMP, TIFF) or 'losslessly' compressed (PNG) source images is recommended rather than working with images in a 'lossy' format such as the popular JPEG format. The Zoomify Converter itself will apply JPEG compression to the tiles created during conversion. When working from lossy compressed source images this would effectively represent a second lossy compression and would introduce the possibility of visible artifacts.

³ Zoomify respects intellectual property rights and any image provided to Zoomify will not be used for any other purpose than testing unless otherwise indicated by the copyright holder.

Troubleshooting image viewing

1. If you receive an error message when attempting to view a Zoomify Image in a web page the error will typically provide information that will help identify the source of the problem. Examples will include an incorrect image path or an incorrect path to another needed resource such as an XML file for the copyright feature or a graphic file for the watermarking feature.
2. Please note that in some cases browsers will function differently when viewing local content than when viewing content over the web from a web server. In particular, Microsoft Internet Explorer may present an alert when viewing web content locally and require the user to verify that they wish to do so, and Internet Explorer, Google Chrome, and Apple Safari may block dynamic loading of local content unless the user's browser settings are modified. In both cases, viewing content from a web site will not be affected - though viewing the example web pages after downloading this product may require extra steps. For additional details on enabling local viewing please see the Frequently Asked Question on the topic on the Zoomify website at www.zoomify.com/support.htm.
3. If the initial display of a Zoomify Image is incorrectly sized or positioned when viewed with Microsoft Internet Explorer v11, please note that IE11 is especially sensitive to the 'DOCTYPE' used in web pages. Please see the example pages in this product for the DOCTYPE that best supports proper display in all recent versions of all major browsers.
4. In unusual cases an error may result from a very old browser version being in use – a version several versions prior to the current version in most cases.
5. In other cases, a problem may occur without an error message being presented. For example it is possible that an image tile may fail to load, or, it may load but fail to fully fade-in. In these cases the problem could result from a busy or faulty network connection. In cases of faulty functionality, such as when an incorrect tile loads or interactivity functions incorrectly or fails to function, the problem could be a similar to those noted above – viewer confusion due to configuration or network access.
6. Unusual display characteristics such as half-visible Toolbar buttons or a badly sized Navigator can be caused by CSS in the web page. The Viewer includes protections against effects of style settings outside itself, however, conflicts are still possible.
7. Alternatively, the problem could be the result of a genuine bug in the Zoomify Image Viewer – in which case the cause will need to be isolated and addressed by Zoomify. If you experience such a problem, please send an email to support@zoomify.com with details. Your support and patience will be greatly appreciated as we do our best to address the issue.

Troubleshooting annotations

The Zoomify Annotation Viewer displays a Zoomify Image, annotation labels (graphics, captions, tooltips) within that image, and annotation data within the Annotation Panel (Points Of Interest, Notes, Label details). The display of the annotation labels and annotation Panel information depend on access to the annotations XML file and to that file being properly formatted.

Symptom: no annotations displayed or error on initial loading of an Image:

1. Verify the web page HTML parameters specify an annotations XML file (for example: “zoomifyAnnotationsXMLPath=ZoomifyExampleAnnotations.xml”).
2. Verify the specified annotations XML file exists in the specified location.
3. Compare the format of the specified annotations XML file to that of the many product examples. If no difference can be found, substitute an example annotations XML file for yours as a test of the formatting.

Symptom: annotations appear but records are missing or incorrect:

1. Compare the formatting of the specific annotations that appear correctly with those that do not. If differences cannot be identified, duplicate the working items and verify the duplicates work as intended, then modify them field by field to match the desired values, testing after each modification.

Symptom: annotations appear correctly but the Annotation Viewer is not displayed in Edit Mode (that is, with Add and Edit buttons):

1. Verify the web page HTML parameters specify edit mode should be enabled (eg. “zoomifyEditMode=1”).
2. Verify the web page HTML parameters specify an annotation 'save handler' (eg. “zoomifyXMLSaveHandlerPath=/servlet/zoomifyServlet.ZoomifyPostServlet”).

Symptom: annotations appear correctly but edits cannot be seen by other users when saved, or disappear after the browser is closed and reopened.

1. Verify the web page HTML parameters specify an annotation 'save handler' (eg. “zoomifyXMLSaveHandlerPath=whateverDirectory/upload.asp”).
2. Verify the specified save handler script exists in the specified location.

Please also note that the latest information from Zoomify and our customers and partners is available on the Support page on the Zoomify website: <http://www.zoomify.com/support.htm>.

Chapter VII. Appendices

Tips and Tricks

The Zoomify Image Viewer supports numerous features to speed navigation and improve the viewing experience. These will increase over time – and suggestions are welcomed!

Keystroke shortcuts:

Click to zoom *and* double-click to zoom-out**

Alt-click* to zoom to 100%

Alt-double-click to zoom fully out (to initial display)

Alt-click the Reset button to return to the prior view

Alt-click-drag any annotation label to reposition it when editing annotations

Alt-click the Zoom-In button to toggle between View and Edit displays of the Annotation Panel when editing annotations

* Macintosh users: option-click substitutes for alt-click.

** Click-zoom in and out move to the point clicked, with the exception that, if enabled, the pan constraint feature will limit pan centering and will auto-center the image when zooming out.

Special Characters in Annotation XML Parameters

The use of special characters is supported in hotspot captions and in annotation fields including POI Name, Note Name, Note Text, Label Name, Label Caption, and Label Tooltip. Useful special characters include those listed below:

apostrophe %27 example: NAME="First POI%27s first label"

carriage return %0D example: CAPTION="First line%0DSecond line"

ampersand %26 example: TOOLTIP="Here%26there"

NOTE: The tab character (URL encoded with %09) is not supported in annotation Name, Note, and other values as it is reserved for switching focus between form fields.

Custom tile support

PNG tiles for transparency and lossless compression

The Zoomify Image Viewer can support PNG tiles (rather than JPG tiles) to support transparency and/or lossless compression. To enable PNG support simply add the optional HTML parameter `zTilesPNG=1` in the calling web page.

The most important step in supporting PNG tiles and transparency is the creation of the tiles themselves. A series of steps is described below which enables rapid conversion of JPG tiles to PNG tiles using Photoshop's Action and Batch features. Additional PNG support notes follow the list.

Steps to convert Zoomify Image tiles from JPEG to PNG format in Photoshop:

1. Open the Window | Actions panel;
2. Create New Action (flipping page button at bottom of);
3. In the 'New Action' dialog enter a name like 'convertJPGToPNG';
4. Select File | Open... and pick the 0-0-0.jpg tile from the TileGroup0 subfolder of the ZoomifyImageExample folder in Photoshop;
5. Select File | Save As... ;
6. Pick PNG-24 format and check Transparency;
7. Select Save and then Save again when presented with the filename dialog;
8. Click the Stop Recording button on the Actions panel;
9. Select File | Automate | Batch to open the batch processing dialog;
10. On the Action dropdown list pick 'convertJPGToPNG';
11. Choose a Source Folder and Destination Folder (the same Zoomify Image folder);
12. Be sure to check Override Action 'Open' Commands as well as Include All Subfolders,

Suppress File Open Options Dialogs, Suppress Color Profile Warnings, and Override

Action 'Save As' Commands;

13. Click OK to run the batch. If presented with a Cancel/Replace dialog due to the preexisting 0-0-0.png file created during the Action recording steps choose Replace;
14. When the Action completes, open each TileGroup subfolder in your Zoomify Image folder, sort by file Type, and delete all the JPEG files;
15. You now have a Zoomify Image folder with PNG tiles.

Additional PNG Support Notes

It is important to be aware that the current Zoomify Converter's support for source files include transparency is untested. It is possible that the original process of creating JPG tiles will eliminate any included transparency. Conversion of the tiles to PNG would therefore require reimplementing of the transparency in each tile (as implemented in this example).

An additional tile creation alternative available to designer/developers with familiarity with scripting languages is the 'zoomify image' converter available on sourceforge.net. While not directly supporting PNG tile output, the project is script-based and therefore may serve as a model for custom development.

Zoomify is evaluating the inclusion of PNG output in its Converter application, however, a delivery timeline has not been set. Zoomify will continue to pursue feature development as rapidly as resources permit, according to priorities that reflect customer needs to the greatest extent possible.

Notes regarding transparency support in Photoshop: 8-bit PNG files support index transparency (like GIF files) while 32-bit PNG files support alpha channel transparency (enabling variable or partial transparency in a full color image with 8 bits for each color and 8 bits for an alpha channel). Essentially, exporting a 32-bit PNG file means exporting a 24 bit color image with an 8-bit alpha channel. Note that in Photoshop a 32-bit PNG file is therefore referred to as a 24 bit PNG file. Also note that while 8 bit PNG support does include alpha transparency, it may be inconsistent in what it applies as the alpha (it uses a color in the palette - white is typically the default).

Rectangular tiles for compatibility with microscopy tools

The Zoomify Image Viewer also supports tiles whose widths differ from their heights – so long as all the tiles in a Zoomify Image are of the same width and all are of the same height. Such rectangular tiles are sometime created by sophisticated scanning microscopes used in digital pathology and other fields.

Use the optional custom tile dimension parameters zTileW and zTileH to adjust Viewer functionality to match custom tiles.

For example:

```
<script type="text/javascript"> Z.showImage("myContainer", "ZoomifyImageExample",  
"zTileW=512&zTileH=128"); </script>
```

Complementary Zoomify products

1. Zoomify Converter Upgrades

Three additional Converter editions are available for separate purchase – all with support of source images of unlimited size.

The *Zoomify Quad-Thread Converter With Tiled Input* improves performance by increasing support from 2 processing threads to 4. This Converter also adds support for tile input source images including tiled TIF images as well as multiple input source files using the ACI format. This enables the creation of massive Zoomify Images from image sets - exceptionally useful given the size limits of many capture devices and image formats.

The *Zoomify Command-Line Converter* enables scripted conversion! Use PHP, Python, JSP, and other scripting languages to instruct the Converter to convert an image. This powerful functionality enables the integration of conversion within server-side systems and full automation of the conversion process.

The *Objective Pathology Universal Converter* delivers support for the most important microscopy formats with enhancements for maximum performance including 64-bit processing and unlimited threads. One solution to handle the output of all of the most popular microscopy tools and the contents of the most respected microscopy slide collections. Supported formats include: tiled ACI, Aperio SVS, Bacus CWS, Mirax & 3D Histech MRXS, Motic CS, BioImagene & Ventana TIF, Hamamatsu NDPI, JPEG 2000, Leica SCN, Objective Imaging & MikroScan SWS, and Zeiss CZI.

Please note: If you are using an operating system that pre-dates Windows XP or OS X v10.5 you may need an earlier version of the Zoomify Converter. Simple contact us at support@zoomify.com and we'll provide one to you.

Detailed steps for using the Zoomify Command-Line Converter and the Zoomify Unlimited Converter including instructions for Tiled Input support are included in the READ ME FIRST file in the Converter folder of the Zoomify HTML5 Pro and Enterprise products.

For general Converter upgrade information, please visit:
<http://www.zoomify.com/converters.htm>.

2. **Zoomify Photoshop Export**

Adobe Photoshop includes built-in support for creating Zoomify Images. This capability is included in Photoshop versions CS3 and newer. Watch the Zoomify website for updates enabling new Export features.

For more information, please visit: <http://www.zoomify.com/photoshop.htm>.

3. **Destiny Toolbar Skins**

Zoomify's partner Destiny Custom Toolbar Skins delivers Toolbar graphics sets that are elegant or edgy and always of professional quality. Easy to implement and inexpensive to purchase, these skins give the Zoomify Toolbar a new level of polish and give web designers exciting new options.

For more information, please visit: <http://www.zoomify.com/extras.htm>.

Uninstalling Zoomify HTML5

Uninstalling Zoomify HTML5 is easy – simply delete the Zoomify download folder and any files that have been manually copied to other locations. No special uninstaller or 'add/remove programs' steps are necessary.

Glossary

Compression:	Use of software algorithm to reduce file size while retaining file information by substituting representations or approximations of the file data for the actual data.
HTML5:	Combination and extension of web technologies including HTML and JavaScript intended to support web page markup and multimedia delivery.
JavaScript:	Standard scripting language supported by all popular web browsers.
Lossy / lossless:	Compression that permanently removes image data to reduce file size as compared to lossless compression, that keeps all original image data intact.
Metadata:	Additional information embedded in an image format (such as a TIFF). May include copyright, author, data, and equipment information.
Minification:	Compilation to shorten function and variable names, remove white space, and remove unused functions (if any). Reduces file size and speeds downloading.
Multi-resolution:	Data file containing multiple levels of quality.
Obfuscation:	Substitution of function and variable names with shorter versions to protect functionality, reduce file size, and speed downloading.
PPF:	Pyramidal File Format: Zoomify's single-file, pyramidal , tiled image, storage format.
Pixel:	The smallest unit of screen display. A point of color.
Pre-fetching:	Intelligent retrieval of data from a web server in anticipation of immediately subsequent views.
Pyramidal image:	A data file format containing multiple levels of resolution.
Resolution:	The amount of image data available for display. One measure of image quality. Screen equivalent of printer 'dpi' (dots per inch).
Tile:	Piece of an image at a given resolution.
Wavelet:	Highly effective image compression approach enabling maximum reduction in file size while preserving image quality.
ZIF:	Zoomify Image Format: Zoomify's optimized single-file, pyramidal, tiled image, storage format accessible without any special server-side setup.

System requirements and additional resources

Authoring

Hardware, memory, and operating system

Windows XP or newer, OS X 10.5 or newer
Image conversion on Unix/Linux requires alternative converter
(separate purchase)

Software applications

Any standard text or HTML editor

Viewing

Hardware, memory, and operating system

Any platform and operating system supporting browsers listed below

Software applications

Recent versions of Google Chrome, Mozilla Firefox,
Microsoft Internet Explorer, Opera Software's Opera, or Apple Safari

Additional resources

Companies & Organizations

Zoomify, Inc..... www.zoomify.com
Objective Pathology..... www.objectivepathology.com
Adobe, Inc..... www.Adobe.com

Software

Zoomify *HTML5 Express*..... www.zoomify.com/html5.htm
Zoomify *HTML5 Pro*..... www.zoomify.com/flash.htm
Zoomify *HTML5 Enterprise*..... www.zoomify.com/enterprise.htm
WinZip Computing Inc.'s WinZip..... www.winzip.com