



IMAGE ENGINE GETTING STARTED

Support

The [ScientiaMobile Support Forum](#) is open to all WURFL users, both commercial license holders and evaluation users. It represents the combined knowledge base for the WURFL community. Commercial licensees are invited to post questions in the forum using the account to which their licenses are associated. This may mean faster handling of those posts by ScientiaMobile's personnel.

For commercial license holders, there are tiered support levels to address a variety of business support needs. After logging into your account, commercial licensees with support options can access the [Enterprise Support](#) portal to post tickets. These tickets will receive expedited attention.

To inquire about support plans, use our [License Inquiry](#) or our [General Inquiry form](#).

Update Notifications

If you would like to be notified of our API updates, major data updates, and other technical changes, please [subscribe](#) to our ScientiaMobile Announcements list

Getting Started with ImageEngine

ImageEngine is an "intelligent" Content Delivery Network for images. Mobile now accounts for most of the traffic in many countries. Naturally, ImageEngine has a special emphasis on optimizing images for the mobile usecase (apps and browsers), but also demonstrate significant value for traditional desktop based browsing.

Using ImageEngine will reduce the image payload significantly and increase website performance.

The easiest way to understand ImageEngine is to see it as an intelligent Content Delivery Network (CDN). The intelligence built in helps you to automatically (or explicitly) optimize images for the requesting device or browser.

Getting Started

The first step is to [register an account](#) in order to receive your unique hostname. Your hostname will follow this template:

```
{token}.imgeng.in
```

For example:

```
abcd.imgeng.in
```

Once registered, you'll also get access to a dashboard where you can manage settings and see statistics. The dashboard is available from [your account page](#).

If you do not want your resized images to be accessible from outside domains, you will need to enable hotlinking for your account. In order to enable this, you will need to add and verify your whitelist of hostnames. Once verified you will be able to enable hotlinking in your account.

The Image Engine service will continue to resize images as long as your account is in good standing order. If you would like to set up billing alerts you may configure this. You may also use this to estimate how much you expect to use the service for any given billing period. Once the billing alert has been exceeded, you will receive an email notification to the account on file letting you know you are approaching the estimated costs.

Note: You will need to remove references to your ImageEngine images in order to halt service.

How does it work?

ImageEngine combines explicit settings, Client Hints and WURFL device intelligence to make sure that the image is as optimized as it can be for any device or browser.

This makes ImageEngine the ideal choice for automating your implementation of responsive images.

In order to begin using ImageEngine, you will need to fetch your "subdomain" token received during the registration process and prefix all your images your ImageEngine url (eg: try.imgeng.in).

If you have an `img` tag like this:

```

```

You will then need to prepend the url with your unique token:

```

```

While prefixing is an easy way to get started and to give ImageEngine a spin, for some it may not be the best solution. If desired, ImageEngine also offers a CNAME option. For example, if your site serves images from `images.example.com`, you may point that hostname to the ImageEngine hostname you got assigned using a CNAME DNS record. In this scenario the `img` tags would look like this:

```

```

Please contact our support team for more information for this option.

The examples above demonstrate the fully automatic optimization, meaning that the size and format is determined by either WURFL or Client Hints.

Parameters

Below is a list of parameters that will explicitly instruct ImageEngine on how to treat the image.

ImageEngine takes commands from the URL preceding the original image URL so in the example below, `w_320` will set the width of the image to 320px. See the below diagram:

```

```

Section	Values	Function
ImageEngine URL	URL	This is the URL of the ImageEngine service.
Image Manipulation Parameters ¹	parameter_value pairs ²	These settings instruct ImageEngine what transformations are expected.
Image URL	URL	This is the image that is to be displayed. The image itself does not need to be on the client site.

¹ See the respective product sections for a complete parameter list.

² Some parameters may have multiple values and have the form `setting_value_value1_value2`

Available Parameters in ImageEngine

Parameter	Name	Value	Description	Example
w	Width	Integer	Desired width of resized picture. 1 3	/w_200/

Parameter	Name	Value	Description	Example
h	Height	Integer	Desired height of resized picture. ^{1 3}	/h_200/
pc	Percentage of screen	1-100 (float)	This field will use WURFL to calculate the screen width and then scale the image accordingly.	/pc_50/
m	Fit method	stretch, box, letterbox, cropbox ²	This setting determines how the engine will resize the images. This can include stretching, cropping, and/or padding out the canvas.	/m_box/
f	Format	jpg, gif, png, bmp, webp	Desired format of picture. If no format is provided, the format of the original picture is used.	/f_webp/
cr ⁴	Crop	Integer	Crop by coordinates	/cr_10,10,100,100/
cmpr ⁴	Compression	Integer	Amount of compression to apply	/cmpr_50/
in ⁴	Inline	Boolean	Return the image as a base64 encoded string for use in data urls	/in_true/

*¹ If a fit method is not specified and only width or height is specified it will select "box". If both width and height are specified by fit is not explicitly specified, then it will assume the fit method is "stretch".

² Fit method has additional optional parameters.

³ If defined, this will take precedence over any automatic setting.

Client Hints

Client Hints enable the browser to send more detailed information about what size and image should be, relative to the viewport size. To enable Client Hints you'll have to explicitly enable it by adding the below <meta> tag to your <head> element:

```
<meta http-equiv="Accept-CH" content="DPR, Viewport-Width, Width, Save-Data">
```

or adding a Link HTTP header in the response of the html:

```
Accept-CH: DPR, Width, Viewport-Width, Save-Data
```

Available Client Hints

Client Hint	Description
DPR	Device Pixel Ratio of the device
Width	Intrinsic size of the image in pixels
Viewport-Width	Size of Viewport
Downlink	Indicates the client's maximum downlink speed. Additional compression may be applied.
Save-Data	Indicate client's preference for reduced data usage. More compression will be applied.

Using client hints and ImageEngine with responsive images will generate a perfectly sized and optimized image. The Client Hints [specification can be found here](#). For more details, please read this [article on Client Hints, Responsive Images and ImageEngine](#).

Image Formats

The formatting parameter allows images to be converted between formats.

Supported Formats

The following formats are accepted by ImageEngine for both input and output.

- PNG ¹
- GIF ²
- JPG ³
- BMP
- WEBP

¹ Both 8-bit and 24-bit PNG files are supported including both index and alpha transparency.

² Not all features of ImageEngine are available for animated GIF files.

³ JPEG can also be used as an alias of JPG.

Input ColorSpaces Supported

Image Format	Greyscale	Indexed Color	RGB	CMYK
JPG	Y	N/A	Y 3	Y 3
GIF	Y 2	Y	N/A	N/A
8-Bit PNG	Y	Y 1	N/A	N/A
24-Bit PNG	Y	N/A	Y 3	N/A
BMP	Y	N/A	Y	N/A

Output ColorSpaces Supported

Image Format	Greyscale	Indexed Color	RGB	CMYK
JPG	Y	N/A	Y 3	Y 3
GIF	Y 2	Y	N/A	N/A
8-Bit PNG	Y 2	N/A	N/A 1	N/A
24-Bit PNG	Y	N/A	Y 3	N/A

Image Format	Greyscale	Indexed Color	RGB	CMYK
BMP	Y	N/A	Y	N/A

¹ 8-bit PNG's are converted to 24-bit by most operations. This will not result in quality loss.

² 8-bit images can be treated like indexed color images for ImageEngine operations. Greyscale images in this case will behave identically to 8-bit indexed color images where the colors available are all greys.

³ Both 8-bit and 16-bit per channel depths are supported

Examples:

Fully Automatic

https://try.imgeng.in/http://upload.wikimedia.org/wikipedia/commons/8/82/2011_Trampeltier_1528.JPG

□

All Images are 300px Wide

https://try.imgeng.in/w_300/http://upload.wikimedia.org/wikipedia/commons/8/82/2011_Trampeltier_1528.JPG

□

Create 200x200px Thumbnails with Black Letterboxes / Pillarboxes

https://try.imgeng.in/w_200/h_200/m_letterbox_000000_100/http://upload.wikimedia.org/wikipedia/commons/8/82/2011_Trampeltier_1528.JPG

□

License

2017 ScientiaMobile Incorporated All Rights Reserved.

NOTICE: All information contained herein is, and remains the property of ScientiaMobile Incorporated and its suppliers, if any. The intellectual and technical concepts contained herein are proprietary to ScientiaMobile Incorporated and its suppliers and may be covered by U.S. and Foreign Patents, patents in process, and are protected by trade secret or copyright law. Dissemination of this information or reproduction of this material is strictly forbidden unless prior written permission is obtained from ScientiaMobile Incorporated.