



The best way to leverage new image formats

After researching image sizes ([Revisiting images formats for the web](#) and [Image formats for the web: HEIC and AVIF](#)) and how to [add new mime types to WordPress](#) there is one final question to ask. How do we leverage responsive images and the new image formats?

Background

We have four image formats to experiment with:

Format	Extension	Notes
JPEG	jpg	Default where no other formats are supported We still need the format because Supported in all browsers
WebP	webp	Smaller file sizes than JPG. Supported in all modern browsers. (caniuse entry)
HEIF	heif	Part of HEVC MPEG family of specifications. Supported in Safari (macOS and iOS)
AVIF	avif	Part of AV1 specification. Supported natively in Firefox (behind a flag) and in Chrome 85 and later (chromestatus entry).

Responsive images provide a client-side solution for delivering alternate image data based on device capabilities to prevent wasted bandwidth and optimize display for both screen and print. (Description taken from the [Responsive Images Community Group](#) web site).

So the question is how we combine responsive images with all the formats we have available?

We will use the `picture` element. In its most basic form it includes one or

more source attributes and an `img` that will work as a default when none of the image formats are supported.

In the example below we use two source child elements. The first one uses a media query to specify the argument that has to match and a `srcset` to describe the images that we use based on screen resolution.

```
<picture>
  <source media="(min-width: 40em)"
    type="image/jpeg"
    srcset="big.jpg 1x,
          big-hd.jpg 2x">
  <source
    type="image/jpeg"
    srcset="small.jpg 1x,
          small-hd.jpg 2x">
  
</picture>
```

So how do we combine the four formats that we want to use in a single picture element?

We take each format and use it in a source element and use it with the corresponding type attribute to indicate the mime type for the content of the source.

```
<picture>
  <source type="image/heif" src="large.heif">
  <source type="image/avif" src="large.avif">
  <source type="image/webp" src="large.webp">
  
</picture>
```

This will work fine but in Retina displays the image will look awful because there

are not enough pixels in the image to display it at the higher resolution. Instead of having a single src attribute we add a srcset and add the images we need to accommodate our resolution.

```
<picture>
  <source type="image/heif"
    srcset="large.heif 1x
      large-2x.heif 2x">
  <source type="image/avif"
    srcset="large.avif 1x
      large-2x-avif 2x">
  <source type="image/webp"
    srcset="large.webp 1x
      large2x.webp 2x">
  <source type="image/jpeg"
    srcset="fallback.jpg 1x
      fallback-2x.jpg 2x"
    loading="lazy"
    alt="Large image of lake Tahoe">
</picture>
```

This is the basic code and the minimal number of images we need. We can use these as the starting point for other responsive images use cases as described in many articles and posts. My reference guide is [Responsive Images Done Right: A Guide To And srcset](#).