Adding images

Web development I: Front-end engineering

Supported image formats

Format type	MIME type	File extension	Comments
JPEG	image/jpeg	.jpg, .jpeg	Photography
PNG	image/png	.png	Photography, allows transparency
GIF	image/gif	.gif	8 bit (256 colors), animations, transparency
SVG	image/svg+xml	.svg	Vector graphics, resolution-independent
WebP	image/webp	.webp	Both for photos and animations
AVIF	image/avif	.avif	Photos and animations, not widely supported

The <img> element

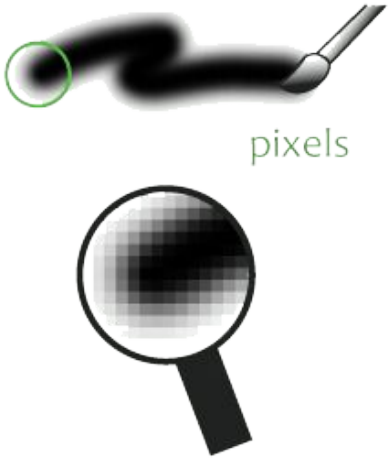
Inline by default

Not affected by CORS policy

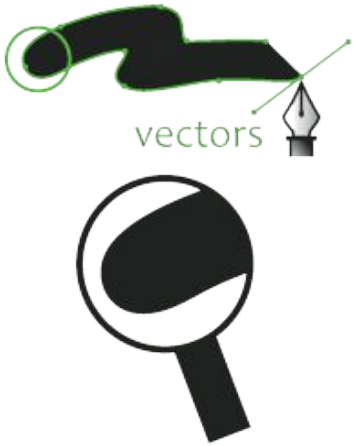
Attributes: `src` (required), `alt` (critical for **accessibility**), `width`, `height`, and more: https://developer.mozilla.org/en-US/docs/Web/HTML/Element/img

Raster *vs* Vector

Raster graphics are stored as a matrix of pixels



Vector graphics are stored as mathematical expressions

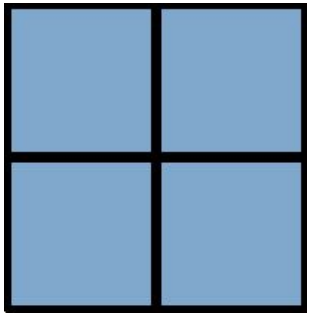


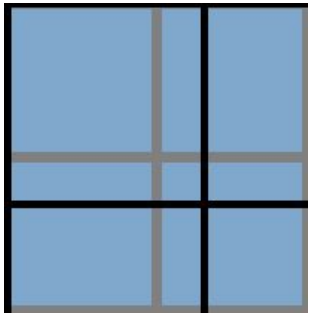
Pixel madness

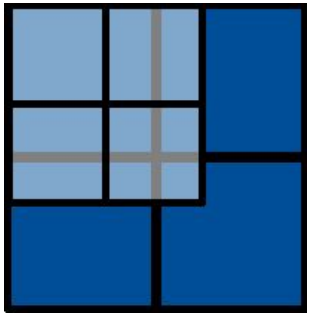
Hardware pixels are indivisible

Reference pixels are device-independent

Px density: “a pixel is a pixel is a pixel” or not?

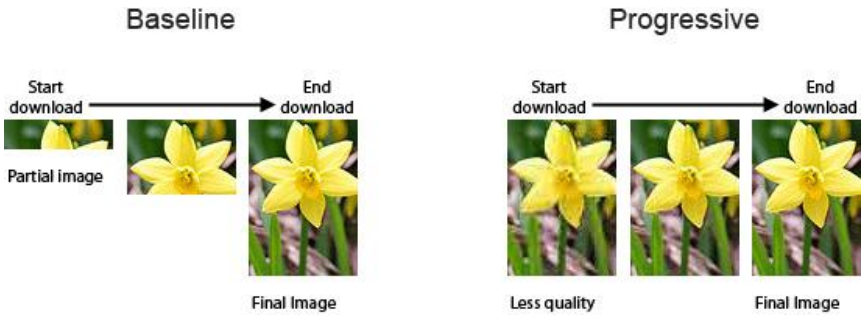






Interlacing

Interlaced graphics load in multiple passes (progressive download)



Compression

UNIVERSITÉ DU LUXEMBOURG

Lossless compression

Is reversible

Preserves original quality

Larger file size

Lossy compression

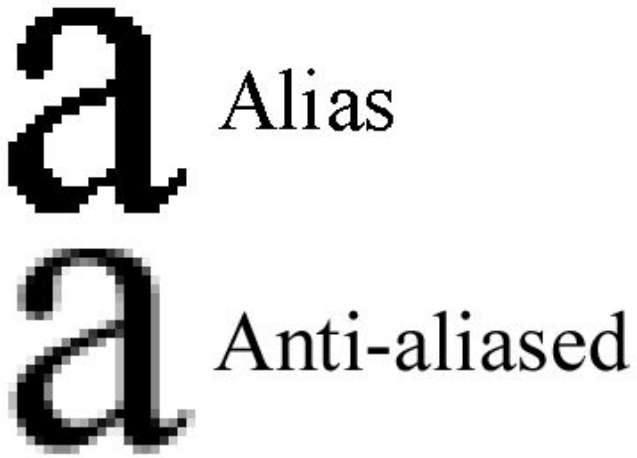
Irreversible

Eliminates redundancies but creates artifacts

Smaller file size

Anti-aliasing

Anti-aliasing



Indexed color palette

Dithering

UNIVERSITÉ DU LUXEMBOURG

Reducing the color range

Original full-color photograph

Dithered to 256 colors

Transparency

Alpha channel *vs* indexed color

Image optimization

General tips:

- Limit dimensions
- Reuse and recycle
- Design for compression
- Use web graphics tools

https://www.smashingmagazine.com/2021/04/image-optimization-pre-release/

Optimizing GIFs

Limit the palette (number of colors)

Reduce dithering

Use flat colors, no gradients

Apply a “lossy” filter

Optimizing JPGs

Be aggressive with compression, but not too much

Soften the image: Blur/Smoothing

Try weighted (selective) optimization

Optimizing PNGs

uni.lu
UNIVERSITÉ DU
LUXEMBOURG

Limit the number of colors

Reduce dithering

Use flat colors

Avoid details

Further tips

UNIVERSITÉ DU LUXEMBOURG

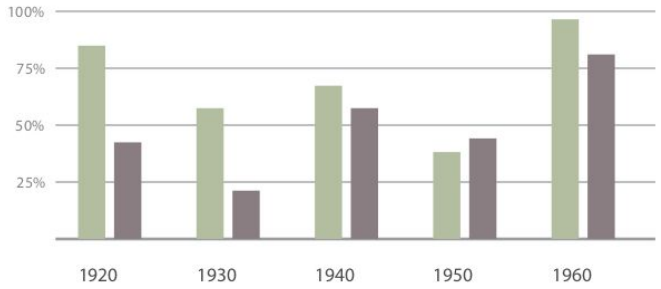
Eagerly vs **lazy** loading

Async decoding

CSS placeholders

Scalable Vector Graphics

The Noun Project



"ben", Open Clip Art

Ozer Kavak, Open Clip Art

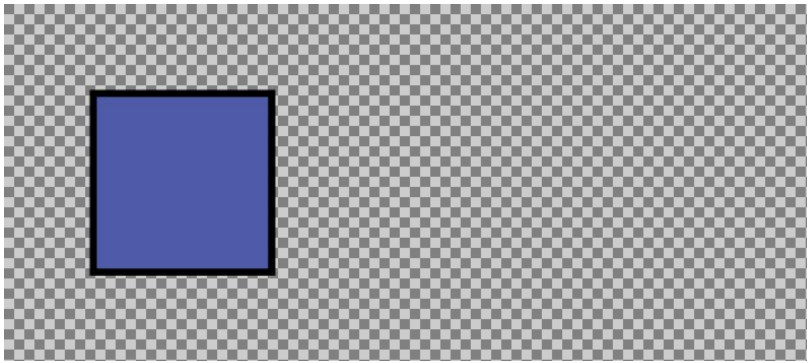
Ghostscript tiger

Scalable Vector Graphics

```
<?xml version="1.0" encoding="utf-8"?>
<svg version="1.1" width="450px" height="200px"
xmlns="http://www.w3.org/2000/svg"
xmlns:xlink="http://www.w3.org/1999/xlink">

  <rect x="50" y="50" width="100" height="100"
fill="#4F5AA8" stroke="#000000" stroke-width="4" />

</svg>
```



Scalable Vector Graphics

```
<?xml version="1.0" encoding="utf-8"?>
<svg version="1.1" width="450px" height="200px"
xmlns="http://www.w3.org/2000/svg"
xmlns:xlink="http://www.w3.org/1999/xlink">

  <rect x="50" y="50" width="100" height="100"
fill="#4F5AA8" stroke="#000000" stroke-width="4">
    <animate attributeName="width"
values="0%;50%;0%" dur="2s"
repeatCount="indefinite" />
  </rect>

</svg>
```

Scalable Vector Graphics

UNIVERSITÉ DU LUXEMBOURG

```
<?xml version="1.0" encoding="utf-8"?>

<svg version="1.1" width="450px" height="200px"
xmlns="http://www.w3.org/2000/svg"
xmlns:xlink="http://www.w3.org/1999/xlink">

  <defs>

    <filter id="my-blur" x="0" y="0" width="200%" height="200%">

      <feOffset result="offOut" in="SourceGraphic" dx="20" dy="20" />

      <feGaussianBlur result="blurOut" in="offOut" stdDeviation="10" />

      <feBlend in="SourceGraphic" in2="blurOut" mode="normal" />

    </filter>

  </defs>

  <rect x="50" y="50" width="100" height="100" fill="#4F5AA8"
stroke="#000000" stroke-width="4" filter="url(#my-blur)" />

</svg>
```