PERFORMANCE

LEISTUNG

INTRO

WHY IS PERFORMANCE IMPORTANT

AMAZON

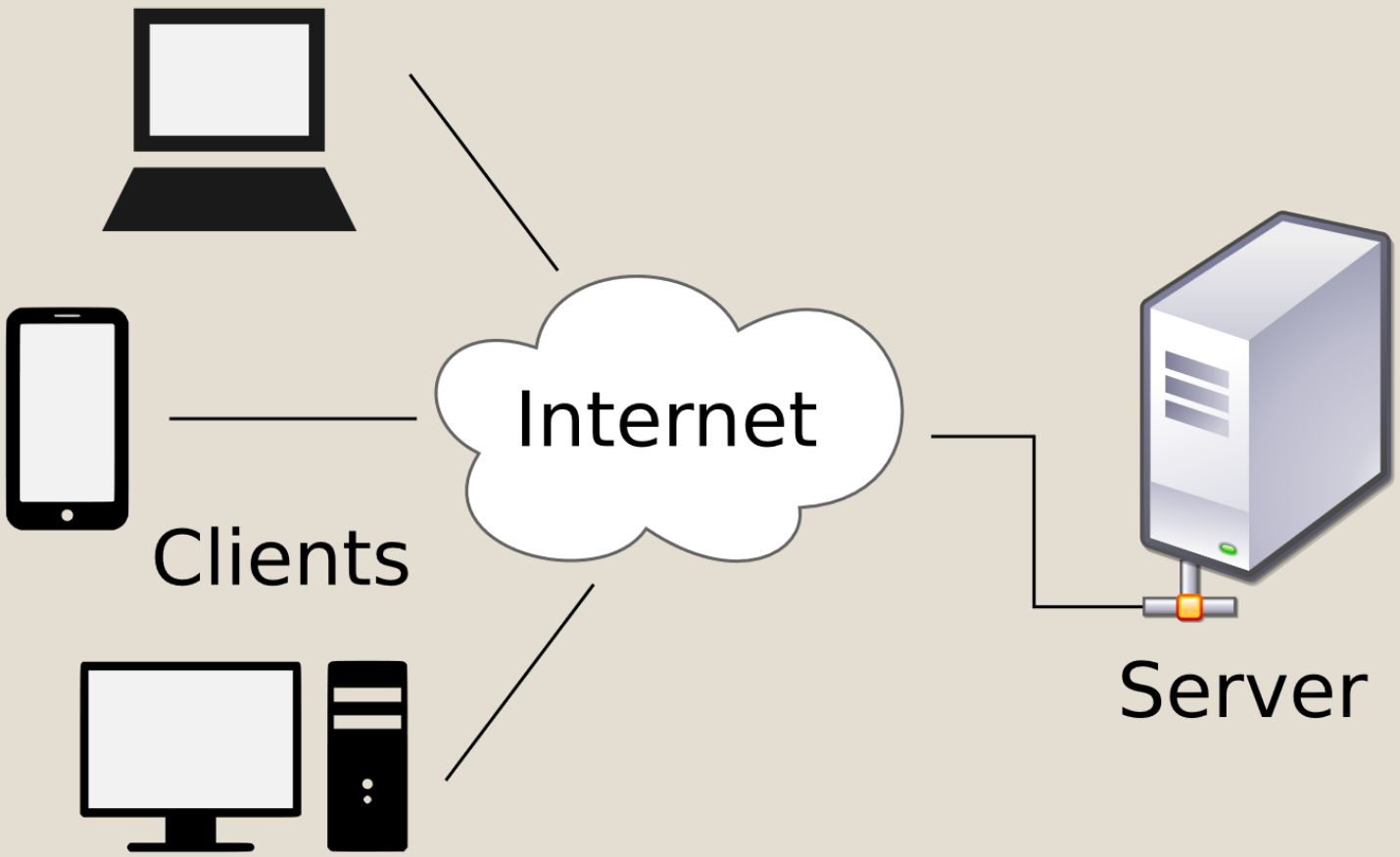
- " Amazon's calculated that a page load slowdown of just one second could cost it $1.6 billion in sales each year."

GOOGLE

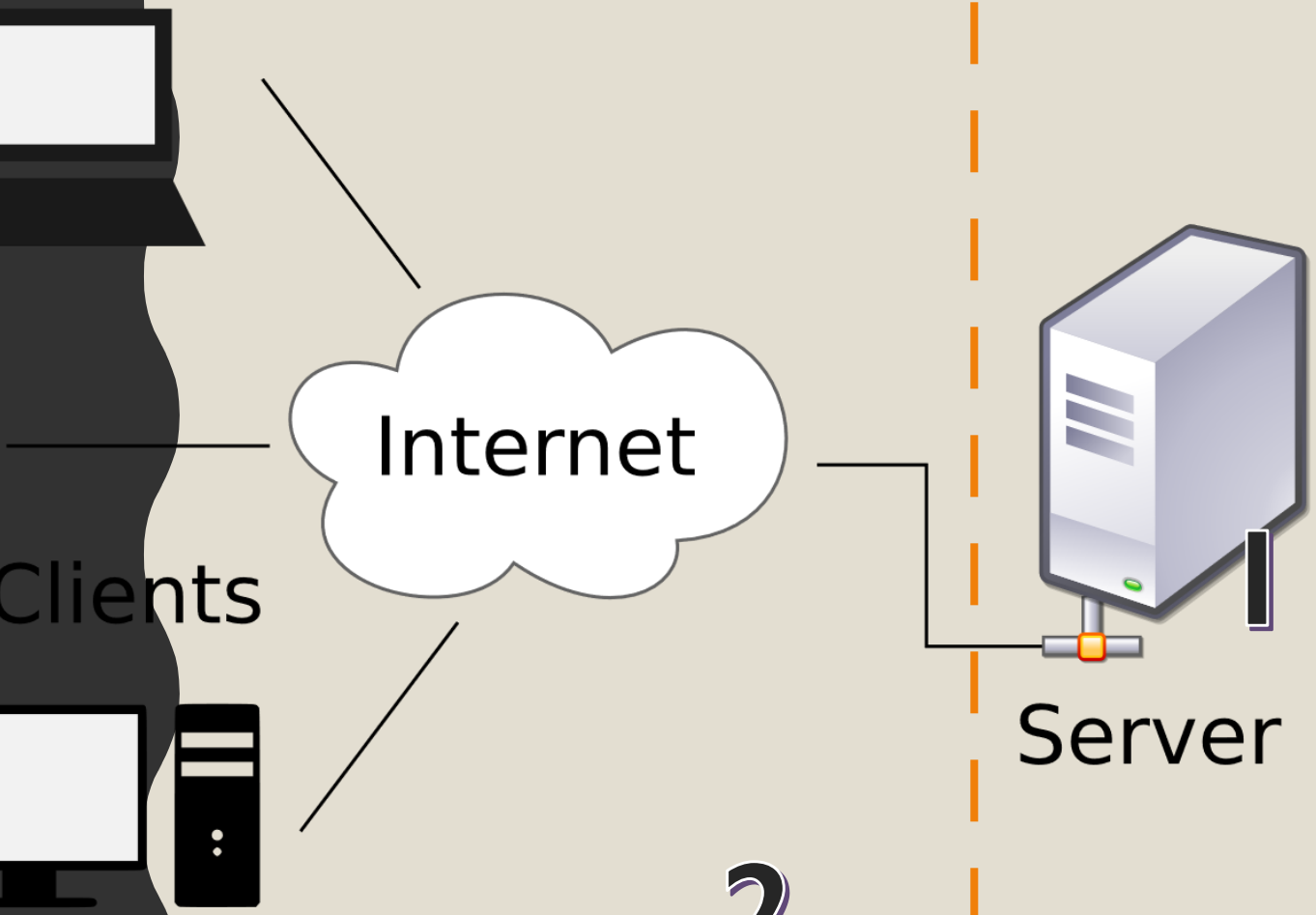
- “Google has calculated that by slowing its search results by just four tenths of a second they could lose 8 million searches per day–meaning they’d serve up many millions fewer online adverts.”

MORE INFORMATION

- https://github.com/zenorocha/browser-diet/wiki/Impact-of-performance




```
<link rel='stylesheet' href='style.css'>
<img src='img/profile.png' alt='profile picture'>
<script src='code.js' ></script>
```



2

3

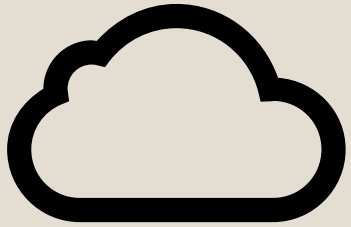
2

FRONT-END

- Critical Render Path
- Optimized Code
- Progressive Web App

1

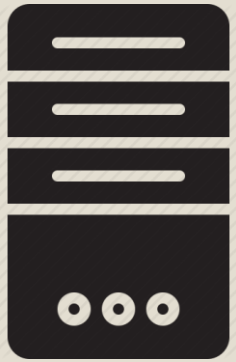
NETWORK



- Minimized Files
- Minimized Delivery

3

BACK-END



- CDNs
- Caching
- Load Balancing
- DB Scaling
- GZIP

NETWORK PRRFORMANCE

FASTER DELIVERY

MINIMIZING TEXT

Manually

- Use online minifiers like uglifyjs / minifier.org

Automatically

- Use tools like webpack

IMAGE FORMATS

OPTIMIZING IMAGES

- Transparency ? PNG
- Animations? GIF
- Colourful Images? JPG
- Simple Icons / Logos / Illustrations? SVG

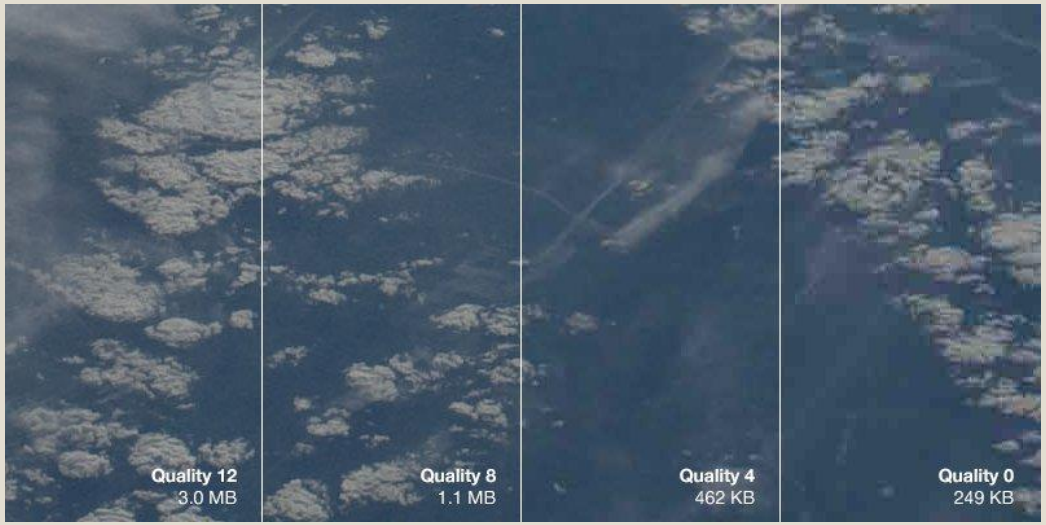
OPTIMIZING IMAGES

- Reduce PNG with TinyPNG 
- Reduce JPG with JPEG-optimizer 
- Simple illustrations > Highly detailed images
- Lower JPG image quality to 30-60% if you can!
- Rezise images based on size needed!
- Different Media Queries for different backgrounds
- imgix
- Remove img metadata (verexif.com) 

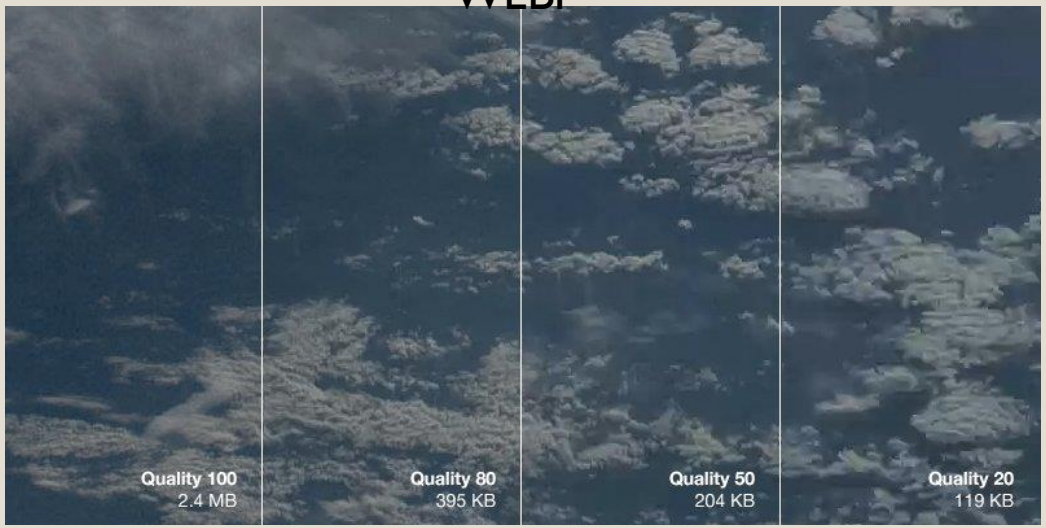
WEBP & JPEG 2000

JPEG2000

JPEG



WEBP



CANIUSE?

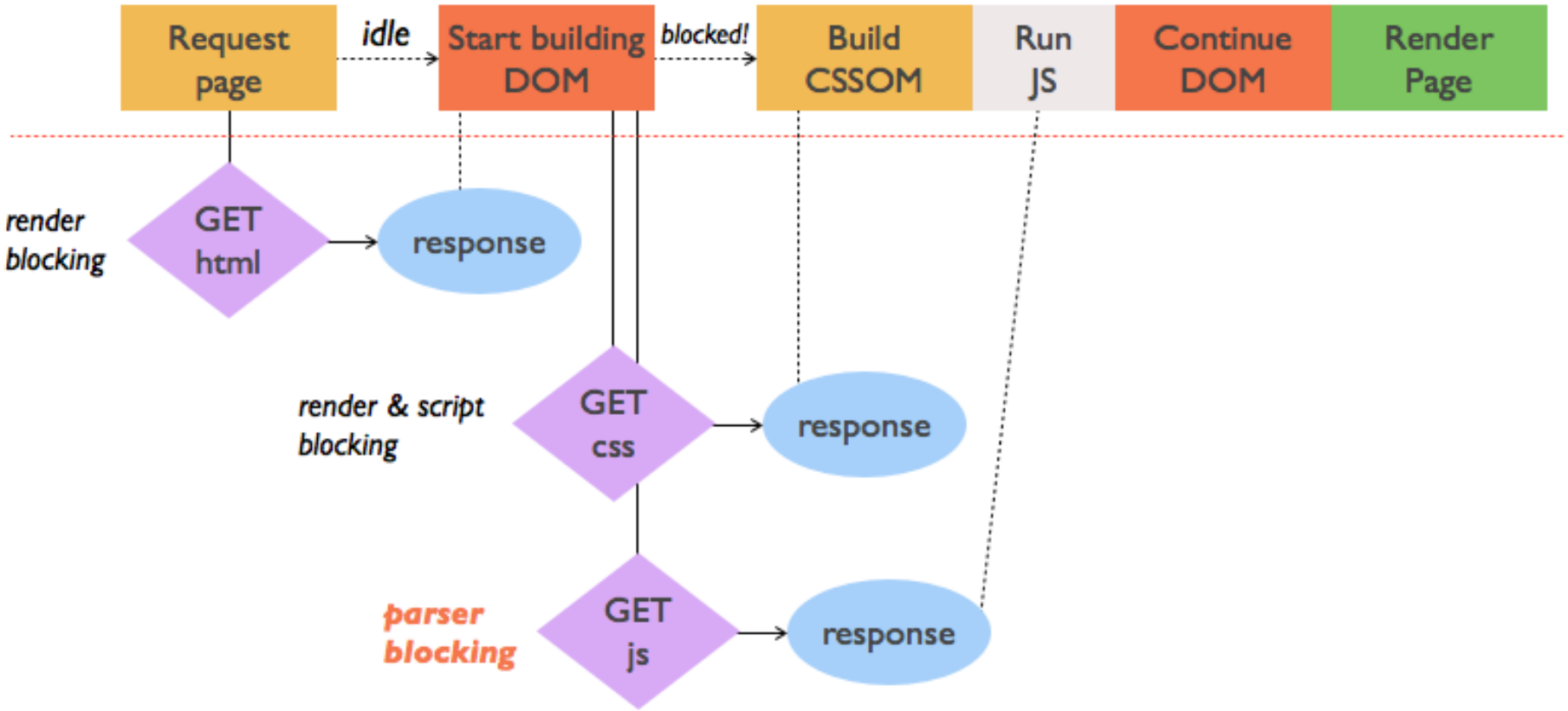
- https://caniuse.com/#search=jpeg2000
- https://caniuse.com/#search=webp

LESS TRIPS

- Is it really necessary to use that library?
  - CSS - Bootstrap, foundation ...
  - JS - jQuery ...
- Don’t saturate your website with images
- Less overall files = faster speed
- HTTP has a limit of files it can handle at once.

CRITICAL RENDER PATH

FRONT-END OPTIMIZATION


Request page
idle
Start building DOM
blocked!
Build CSSOM
Run JS
Continue DOM
Render Page
render blocking
GET html
response
render & script blocking
GET css
response
parser blocking
GET js
response

HTML OPTIMIZATION

- Load JS as late as possible
- Load CSS as early as possible

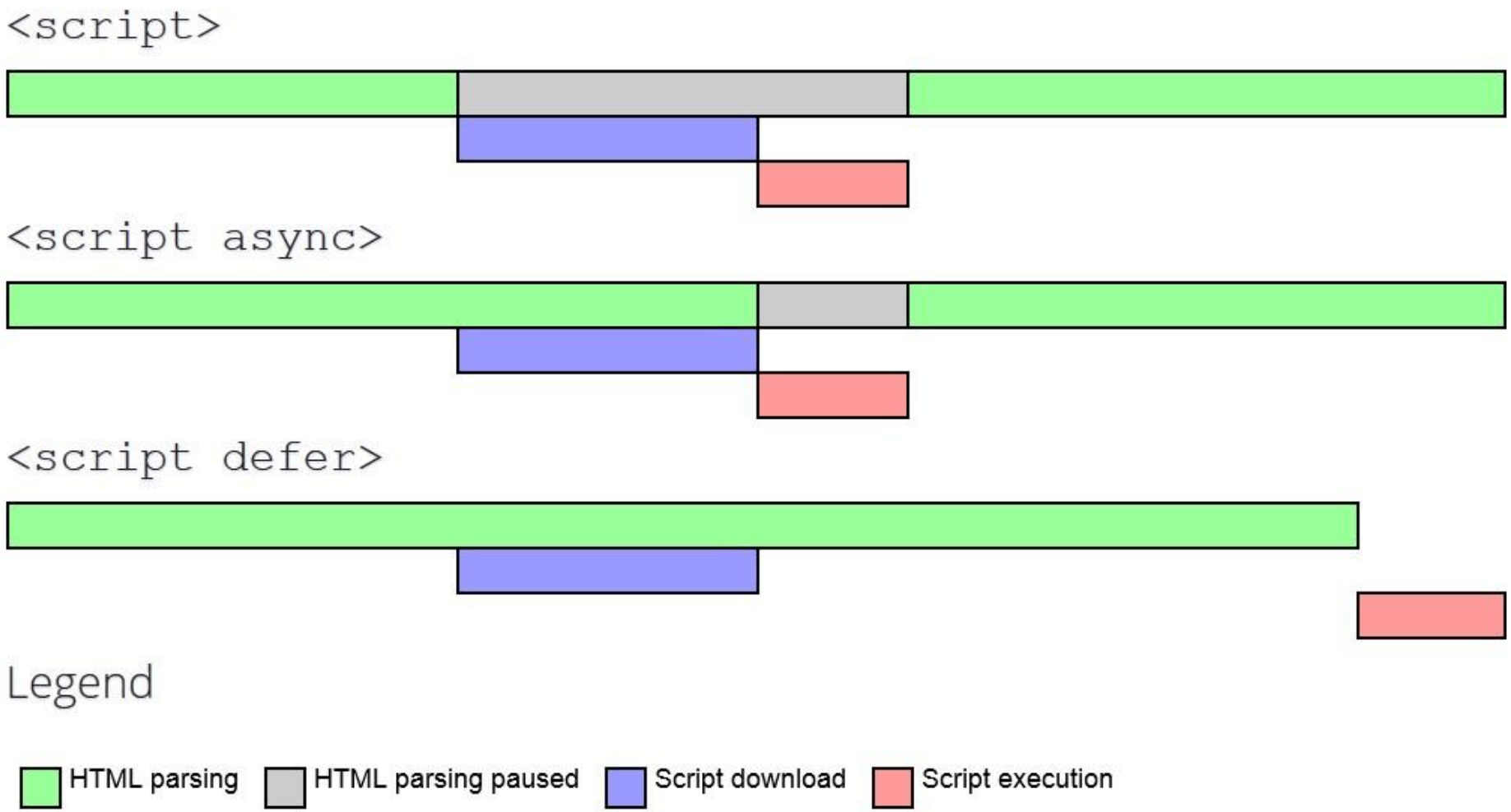
CSS OPTIMIZATION

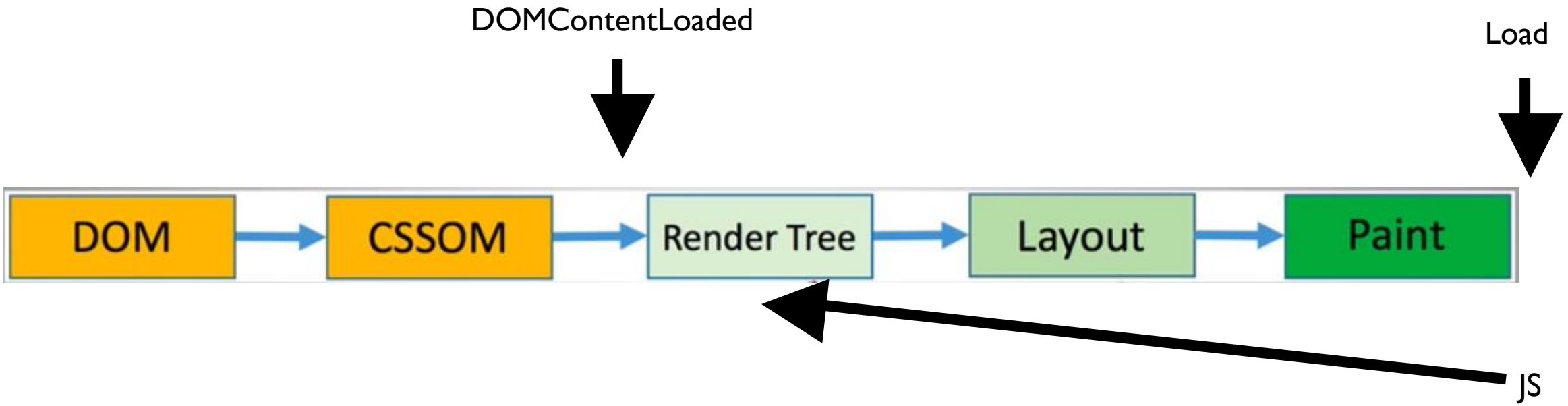
  - CSS is render-blocking
- Only load whatever is needed
  - Get rid of unused rules in your stylesheet.
- Use above the fold loading
  - A user is only interested in what they see
- Media Attributes
  - Html media attributes
- Less Specificity
  - Specificity takes time to calculate

JS OPTIMIZATION

- Once a script tag is dicovered, DOM construction is paused
- Script can't be executed until CSSOM is loaded.
- Thus JS is called a 'Parser Blocker'

  - We can
- Load Scripts asynchronously
- Defer loading of script
- Minimize DOM Manipulation
- Avoid Long running JS


<script>
<script async>
<script defer>
Legend
HTML parsing
HTML parsing paused
Script download
Script execution


DOMContentLoaded
Load
DOM
CSSOM
Render Tree
Layout
Paint
JS