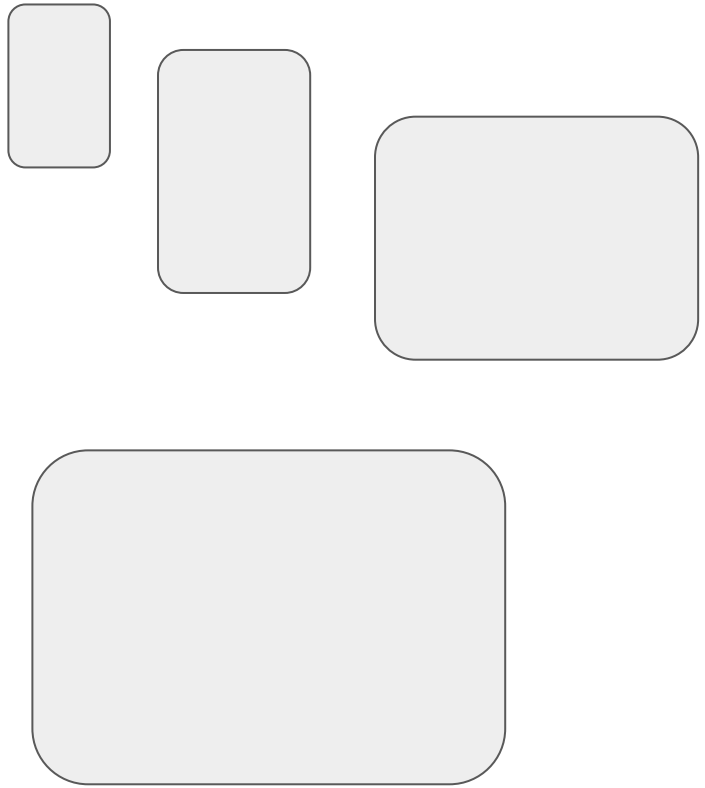
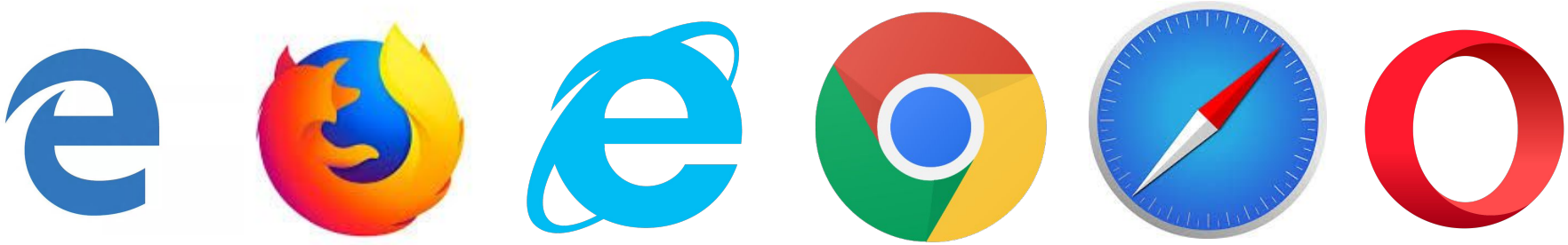
Automated Visual Testing

GovTech QE Chapter
December 2018

Responsiveness

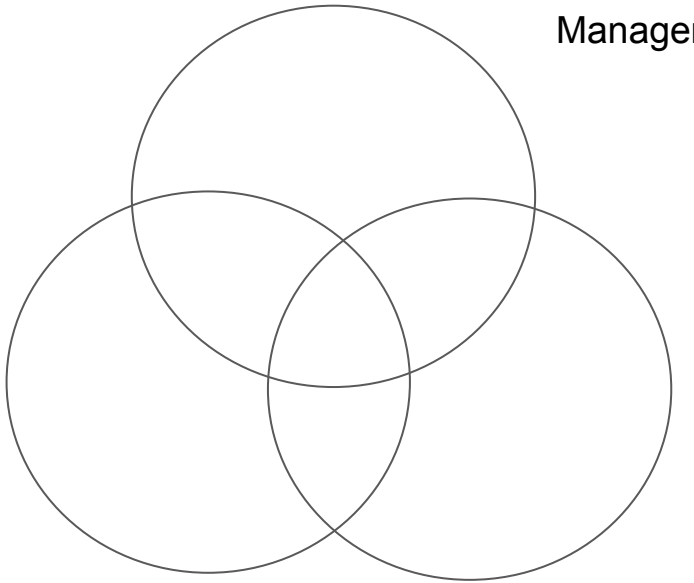


Browser Compatibility



Why Applitools?

Dashboard and Project Management



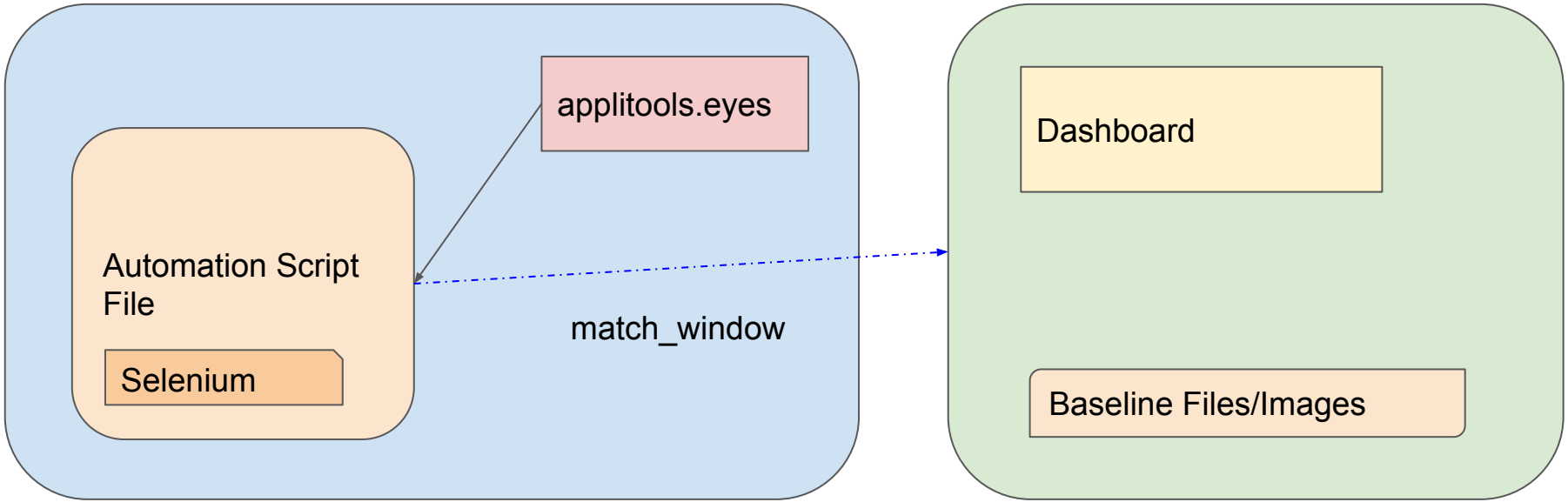
DevOps Integration

Visual Automated Testing

Visual ~~Automated~~ AI Testing

- Applitools

Applitools Overview



Applitools Demo

Need Benchmarks
Need Use Cases
Need User Pain Points

Other Visual Automated Testing Options

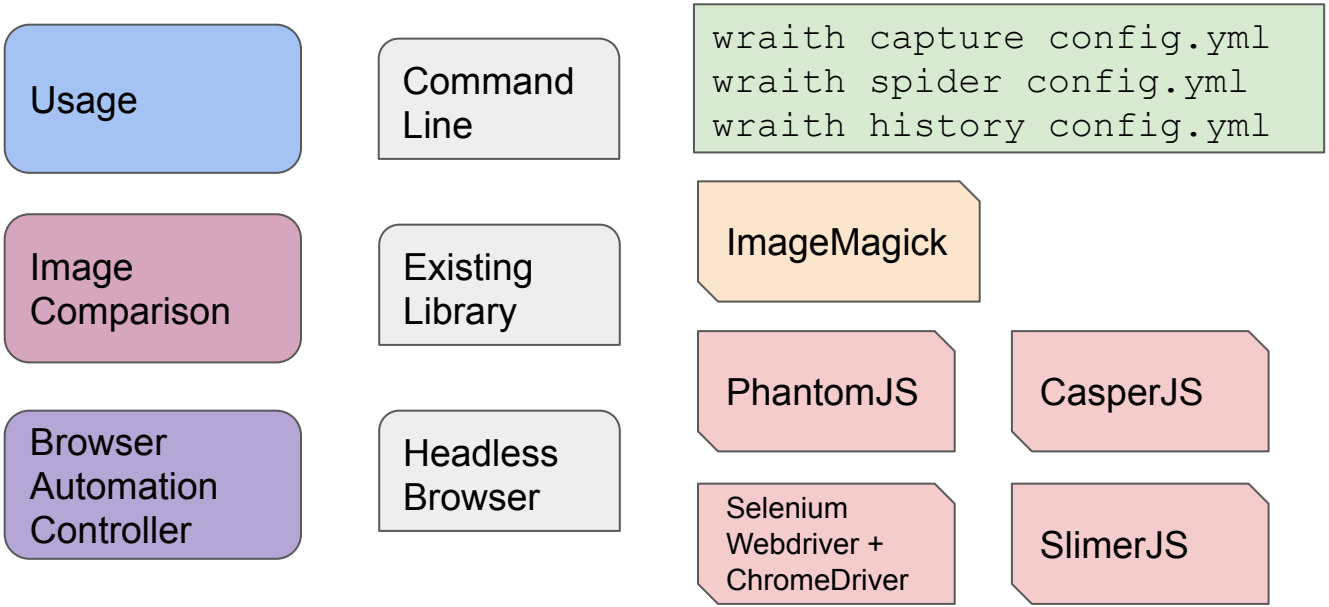
- Wraith
- WebdriverCSS
- Galen Framework
- Phantom CSS
- Gemini

Visual Automated Testing

- **Wraith**
- WebdriverCSS
- Galen Framework
- Phantom CSS
- Gemini

Wraith

- Developed at BBC
- Used for checking CSS changes; intentional and unintentional



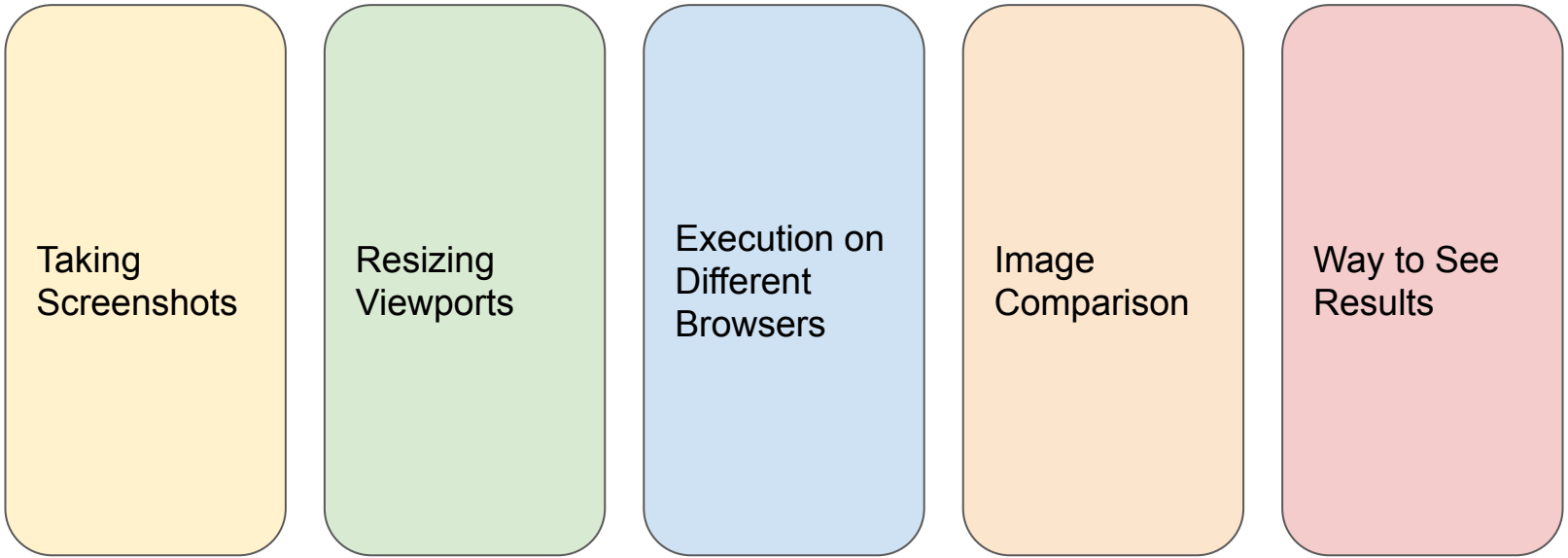
```
#Headless browser option
browser:
  phantomjs: "phantomjs"
  # slimerjs: "slimerjs"

#If you want to have multiple snapping files, set the file name here
#snap_file: "javascript/snap.js"

# (optional) JavaScript file to execute before taking screenshot of every path. Default: nil
#before_capture: 'javascript/disable_javascript--phantom.js'
before_capture: '../javascript/wait--phantom.js'

# Type the name of the directory that shots will be stored in
directory: 'gallery_bbc_shots'

# Add only 2 domains, key will act as a label
domains:
  #default: "http://www.bbc.co.uk/news"
  russian: "http://www.bbc.co.uk/russian"
  #english: "http://www.bbc.co.uk/english"
  burmese: "http://www.bbc.co.uk/burmese"
  #chinese: "http://www.bbc.co.uk/chinese"
  #asia: "http://www.bbc.co.uk/asia"
  #chinese: "http://www.bbc.co.uk/french"

#Type screen widths below, here are a couple of examples
screen_widths:
  - 320
  - 600
  - 768
  - 1024
  - 1280
```

```yaml
#Type screen widths below, here are a couple of examples
screen_widths:
  - 320
  - 600
  - 768
  - 1024
  - 1280

#Type page URL paths below, here are a couple of examples
paths:
  home: /
  uk_index: /uk

#Amount of fuzz ImageMagick will use
fuzz: '20%'

#Set the number of days to keep the site spider file
spider_days:
  - 10

#Choose how results are displayed, by default alphanumeric. Different screen widths are always grouped.
#alphanumeric - all paths (with, and without, a difference) are shown, sorted by path
#diffs_first - all paths (with, and without, a difference) are shown, sorted by difference size (largest first)
#diffs_only - only paths with a difference are shown, sorted by difference size (largest first)
mode: diffs_first

threshold: 5
```

Wraith Promise

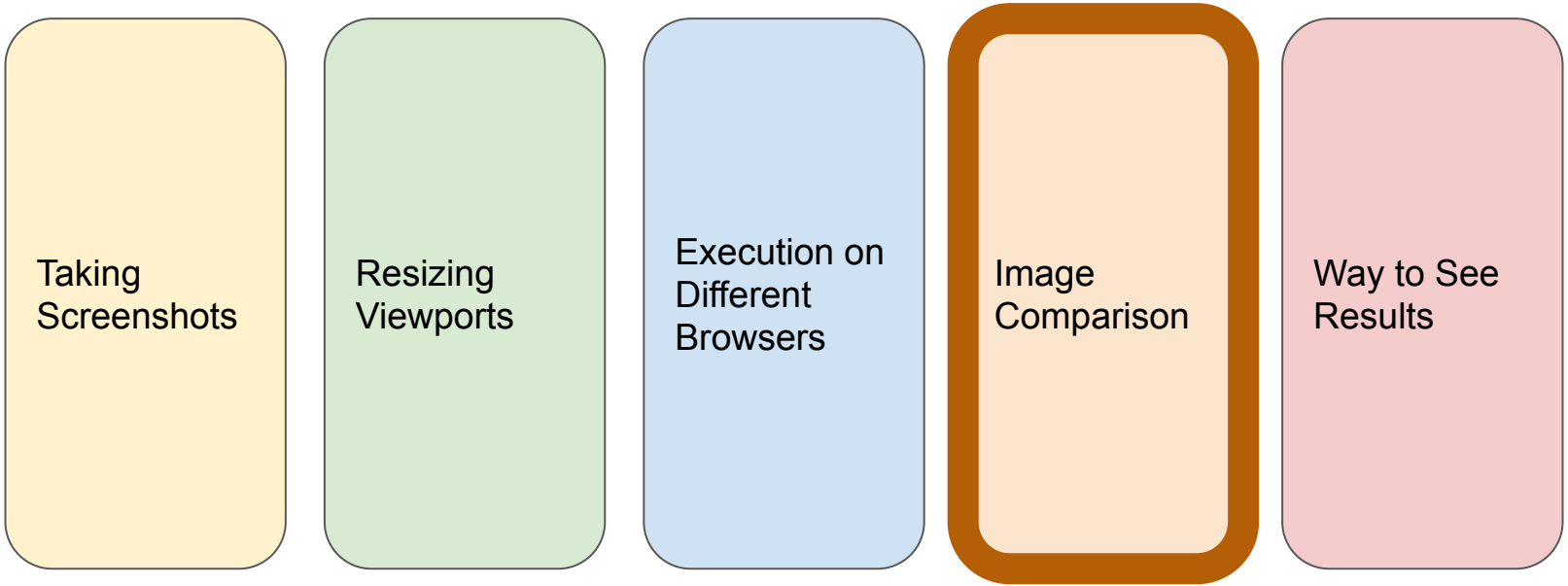
- takes screenshots of your webpages
- runs a comparison task across them
- outputs a diff PNG file comparing the two images, and a data.txt file which contains the percentage of pixels that have changed
- packages all of this up into a gallery.html, ready for you to view
- if any screenshot's diff is above the threshold you specified in your configuration file, the task exits with a system error code (useful for CI)
- the failed screenshot will also be highlighted in the gallery

Essentials


Taking Screenshots
Resizing Viewports
Execution on Different Browsers
Image Comparison
Way to See Results

Ownself develop ownself?

Essentials


Taking Screenshots
Resizing Viewports
Execution on Different Browsers
Image Comparison
Way to See Results

Similarity Measures

+ Mean Squared Error

$$MSE = \frac{1}{m\,n} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [I(i,j) - K(i,j)]^2$$

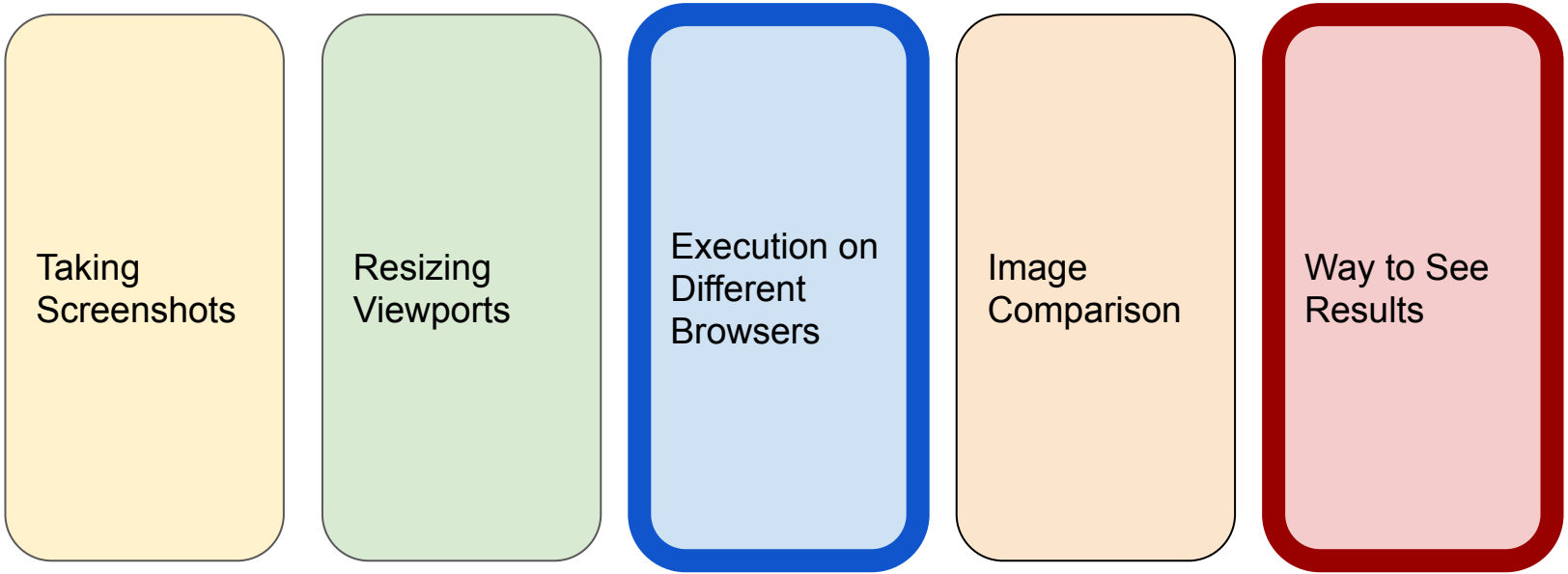
+ Structural Similarity Index

$$\text{SSIM}(x,y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}$$

Reference:
https://www.pyimagesearch.com/2014/09/15/python-compare-two-images/

Freezer demo

Essentials


Taking Screenshots
Resizing Viewports
Execution on Different Browsers
Image Comparison
Way to See Results

That’s all! Thank you and feedback welcomed ~