

# Dump {less} and Sass

Dynamic CSS Manipulation

*Powered by JavaScript*



**What I am showing you today is**

**“Impossible”**

- not completely anonymous co-worker

# What's wrong with {less}, Sass and static CSS?

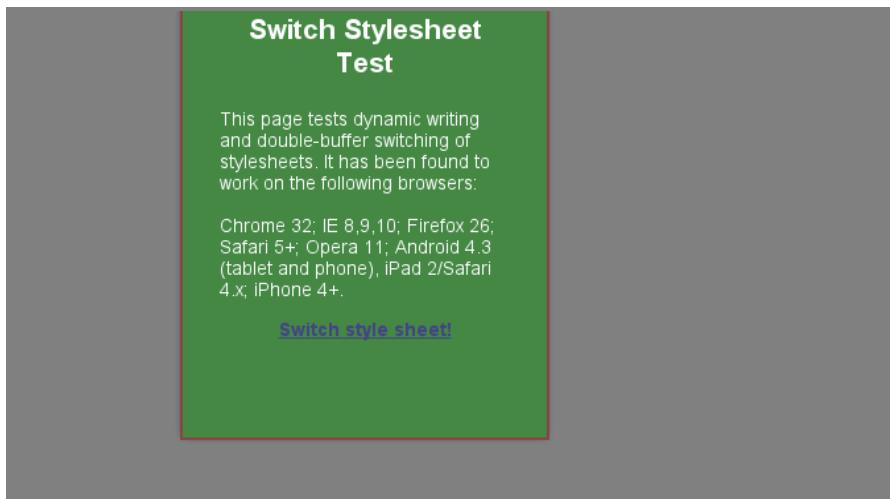
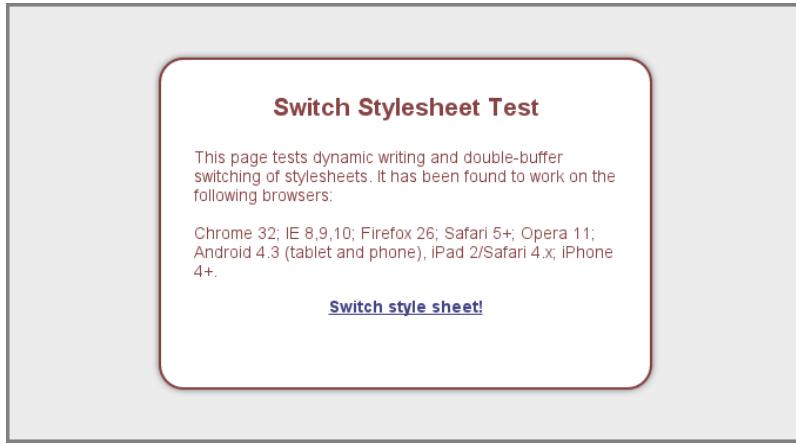
- “I used Sass once, and it ate my dog!” - Louis, Manning reviewer
- “Just because everyone else is jumping into {less} and Sass doesn't mean you should too!” - your parents

another abstraction layer (DSL) to master which obscures the underlying technology

- All create CSS at the WRONG TIME

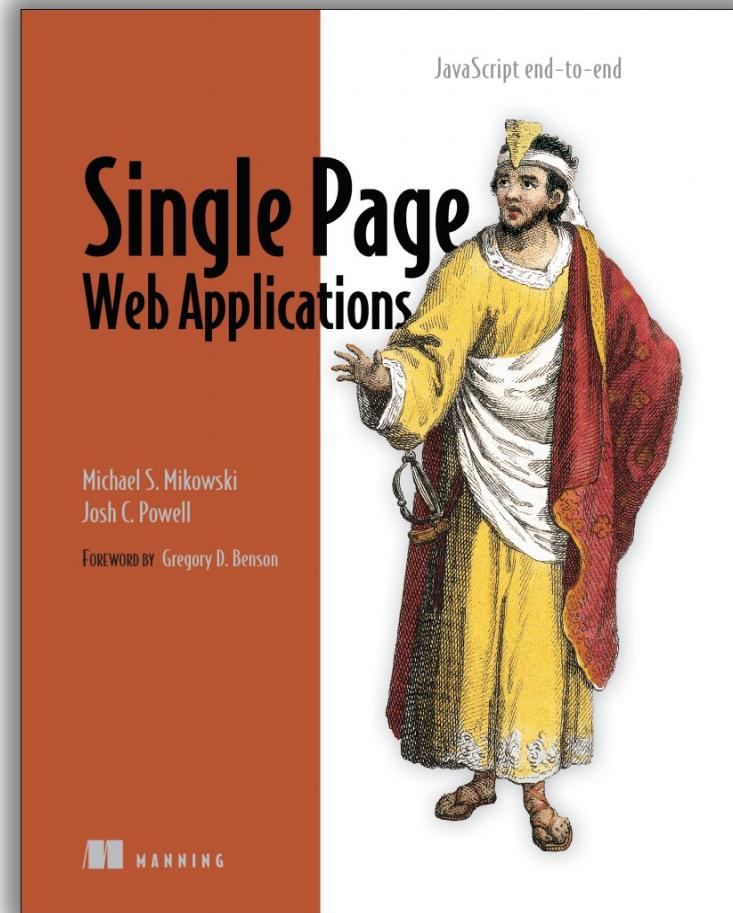
# Simple JS-CSS example

## Action served here



# About Michael S. Mikowski

- Single Page Web Applications – JavaScript end-to-end
- Senior Director, UX Engineering at Qualaroo
- Developer on Six production SPAs since 2006, Architect on all but one
- Previous dev manager on HP/HA mod\_perl clusters (~2B web transactions per week)
- First SPA: European and US AMD “wheretobuy.amd.com,” rel. 2007
- Favors boring convention....?



# What is JS-CSS?

- CSS **created** and **on demand** by JavaScript application
- Not static, compiled
- Not loaded from a file

JS-CSS



Created and  
*on demand* by a  
script application

# The JS-CSS Style Manager

- Developed and refined at Qualaroo for a year
- Today we are **sharing with the world**
- Can replace {less} and Sass in many cases
- We will use:
  - JavaScript
  - Browser features
  - HTML, DOM, and CSS



# Another JS-CSS Example

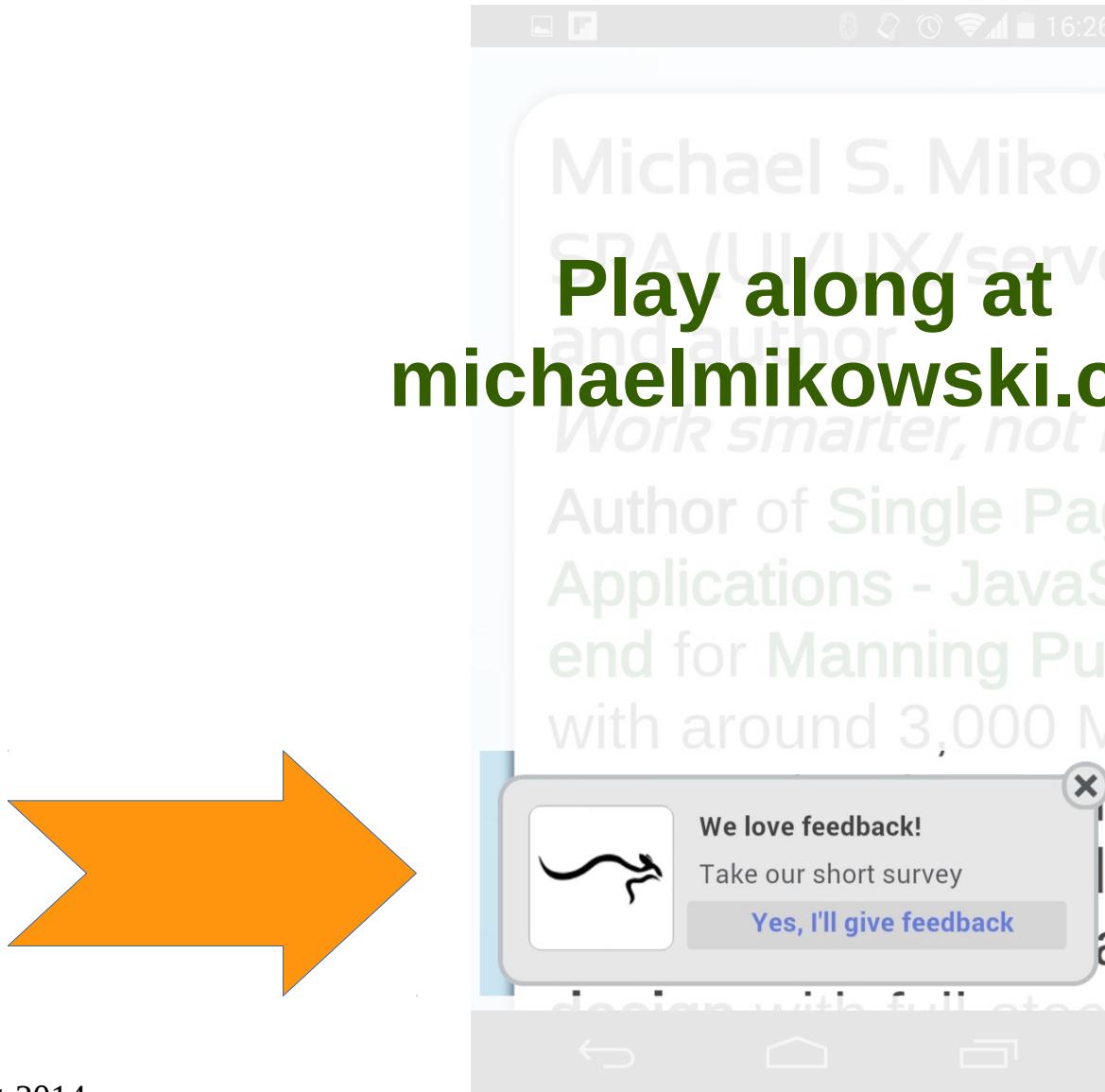
Over 100 million end users every day

- Qualaroo => web marketing surveys
- “Third party JS”
- Load only **one** file:
  - No jQuery
  - No external images
  - No external CSS
  - No external HTML
  - In-line data



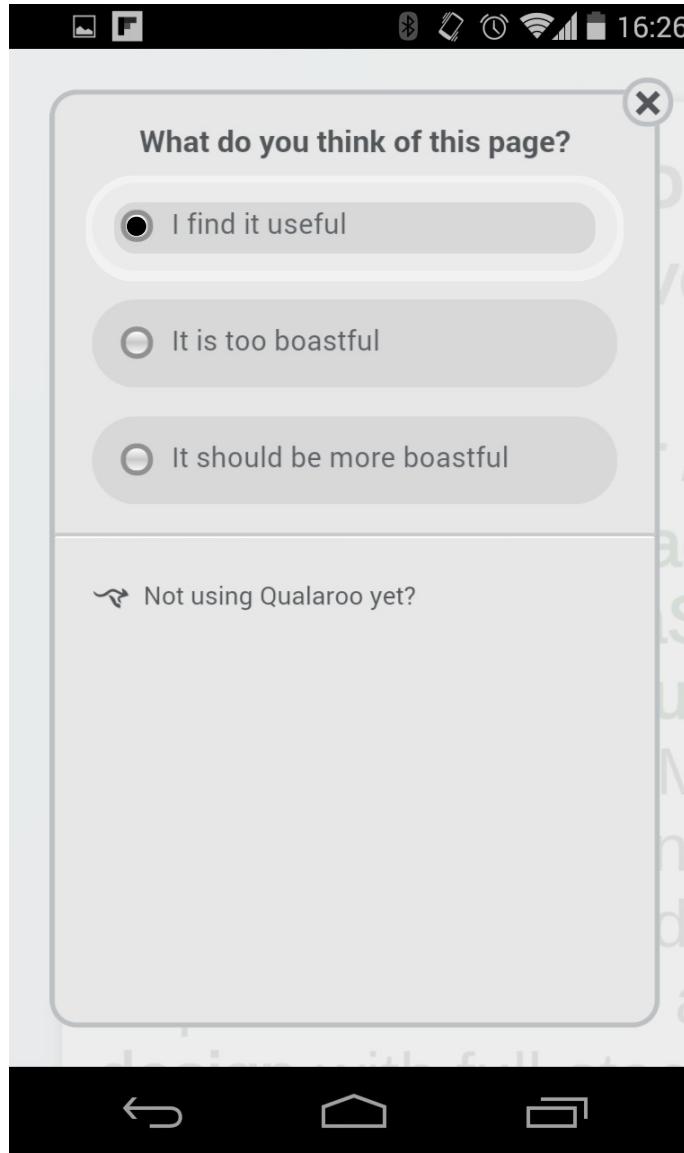
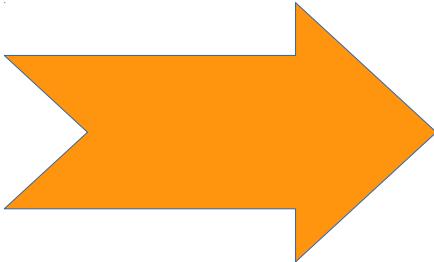
# Example Survey

## (1) Optional “screener”



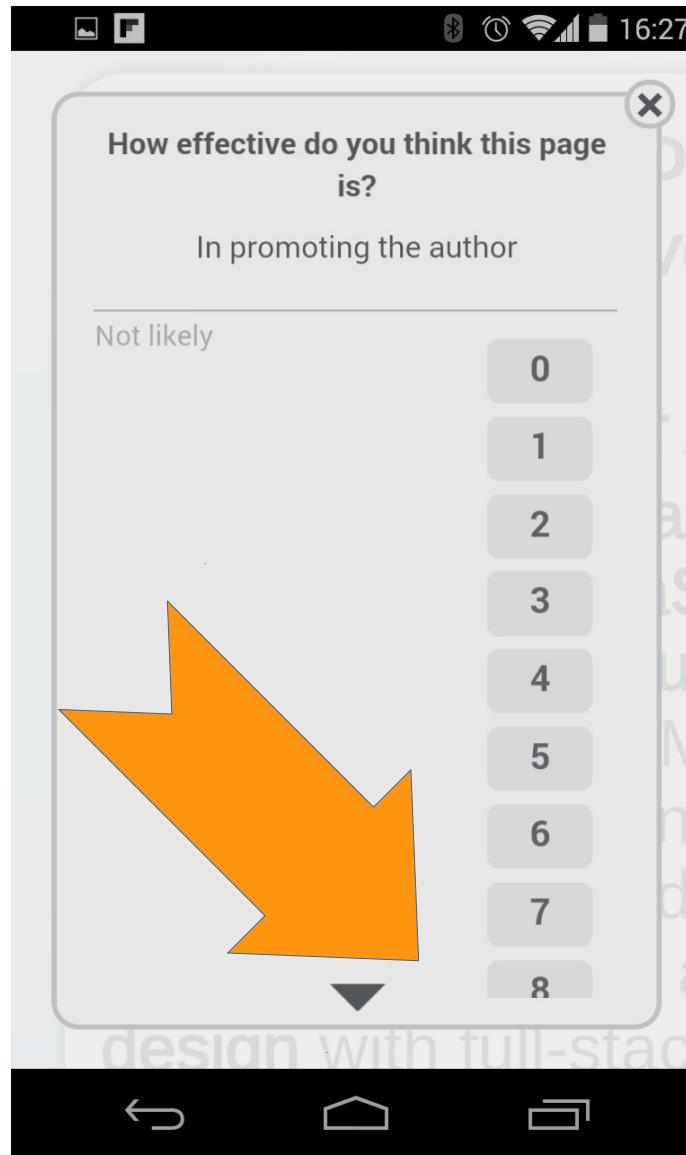
# Example Survey

## (2) multi-choice question



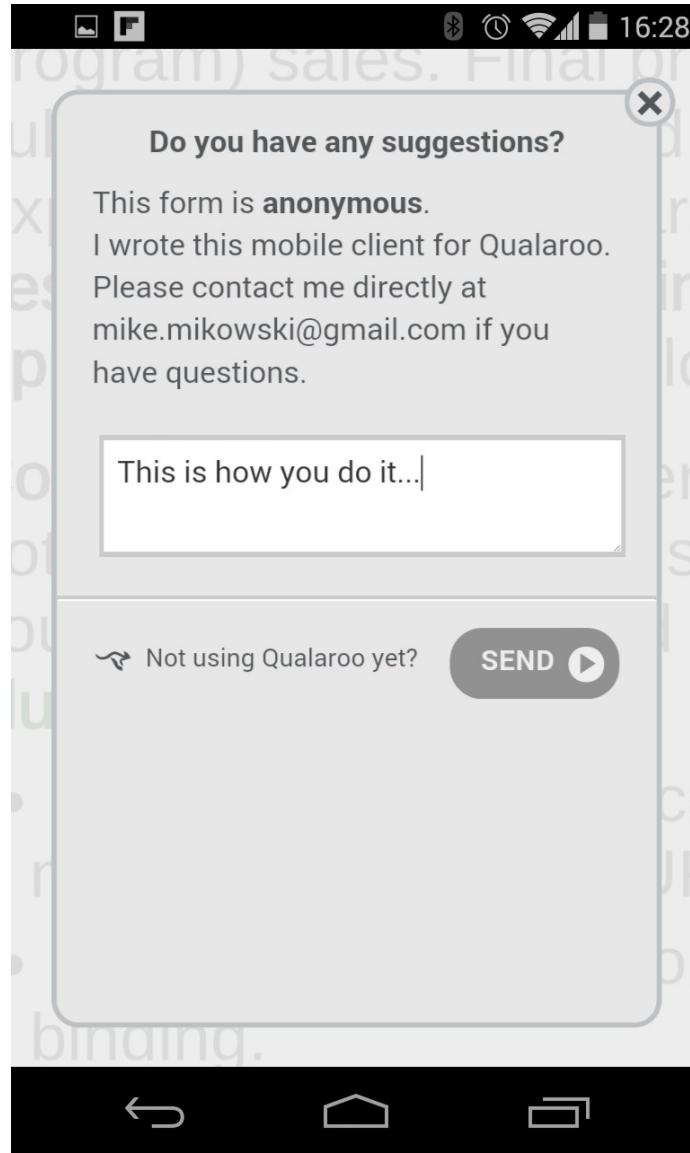
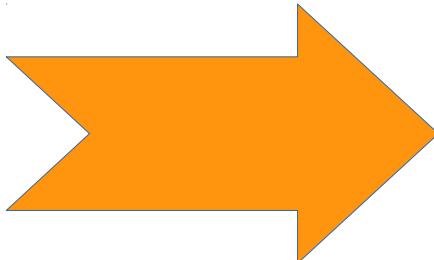
# Example Survey

## (3) Rating question



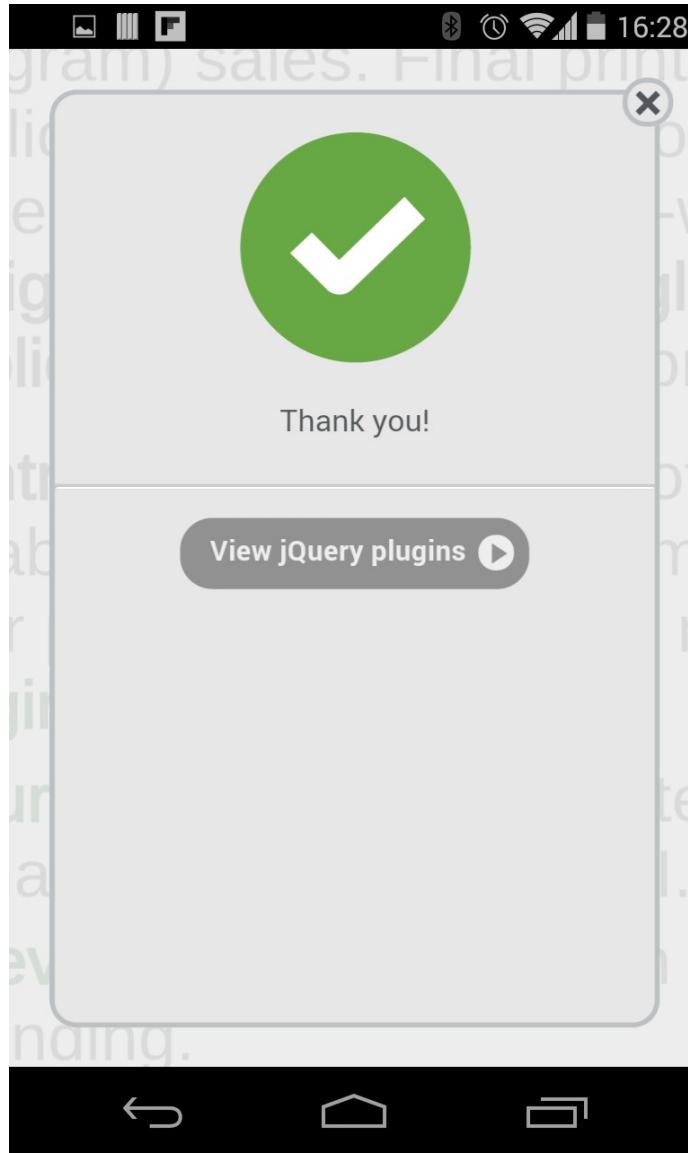
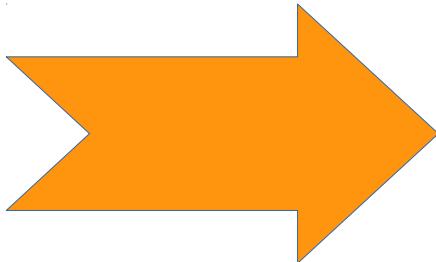
# Example Survey

## (4) Text input



# Example Survey

## (5) And “thank you”



# **Is it simple?**

Not on your life...

Numerous device sizes and capabilities

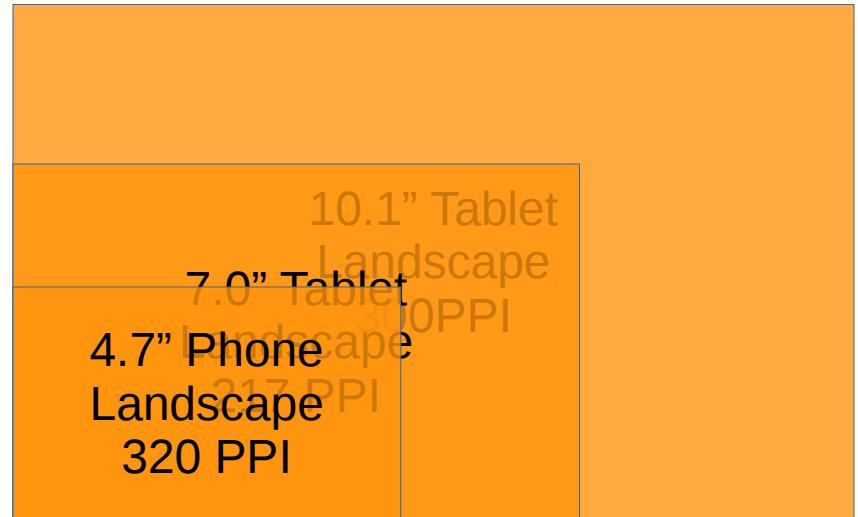
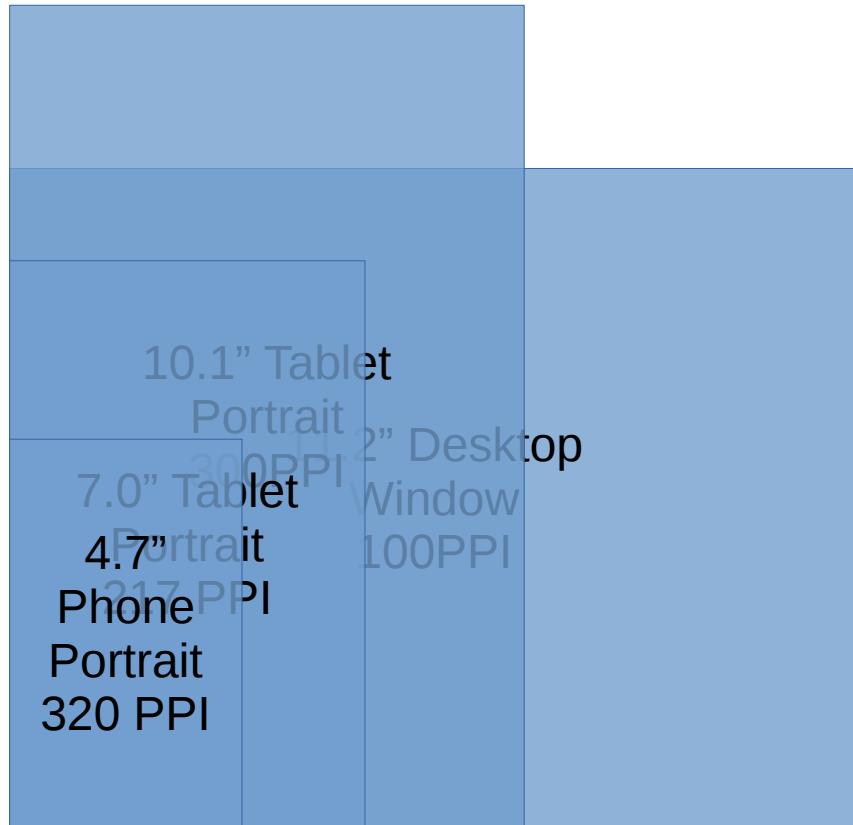
- Varying zoom and pan level
- Device orientation
- Host site color scheme

**Display needs dynamic generation to be  
meet our needs!**

**Create custom CSS for every viewer!**

# Just look at all the device fun!

This is just a small sample of clients





Contributor on a number of open source projects, notably [TaffyDB](#). Here is my [GitHub](#) account for your perusal.

Published a number of [jQuery plugins](#), including:

- [uriAnchor](#) - Sophisticated SPA state management in the URI.
- [event.gevent](#) - Custom global events for data binding.
- [event.ue](#) - Unified desktop and touch events.

# (1) Screeners

Researched many tools and emerging technologies such as [SVG](#) and [mobile](#) interfaces. Employed technologies such as frame rate compensation, proximity

Michael S. Mikowski - Mozilla Firefox

Michael S. Mikowski

Author of [Single Page Web Applications - JavaScript end-to-end](#) for [Manning Publications Co.](#) with around 3,000 MEAP (Manning Early Access Program) sales. Final print and electronic publication is scheduled for June 2013. Experience blends award-winning [industrial design](#) with full-stack [Single Page web Application \(SPA\)](#) development.

Contributor on a number of open source projects, notably [TaffyDB](#). Here is my [GitHub](#) account for your perusal. Published a number of [jQuery plugins](#), including:

- [uriAnchor](#) - Sophisticated SPA state management in the URI.
- [event.gevent](#) - Custom global events for data binding.
- [event.ue](#) - Unified desktop and touch events.

Researched many tools and emerging technologies such as [SVG](#) and [mobile](#) interfaces. Employed technologies such as frame rate compensation, proximity sensing, collision detection, and audio design to bring a highly engaging user experience to enterprise software. Researched and employed [Node.js](#) and [MongoDB](#). Have many years experience developing software for, and managing, large (100+ server) [Linux clusters](#) that served billions of web transactions per week using [MySQL](#). My favorite SCMS is [git](#) but also administered [Subversion](#), [Perforce](#), and [CVS](#).

Presented to thousands of people over the last nine months, including three presentations at the last two [HTML5 developer conferences](#), [Developer week](#), [University of San Francisco](#), and [Trendulate](#). All but one of the public presentations had an audience substantially over capacity<sup>1</sup> As a result I have many followers on [Google+](#), [LinkedIn](#), and [Manning Publications Co](#). The publicity has reached many thousands of influencers, potential employees, and customers.

The [Fog of SPA](#) is my most recent publication and presentation. It shares seven hard-won lessons from designing and delivering five commercial SPAs. It shows how design for quality, creative destruction, minimalism, and targeted testing techniques can remove complexity and confusion from SPA development.

### References

References are listed from most recent interactions. I respect the privacy of my references, and therefore please connect via [LinkedIn](#) if you wish to acquire contact details.

Presented to thousands of people over the last nine months, including three presentations at the last two [HTML5 developer conferences](#), [Developer week](#), [University of San Francisco](#), and [Trendulate](#). All but one of the public presentations had an audience substantially over capacity<sup>1</sup> As a result I have many followers on [Google+](#), [LinkedIn](#), and [Manning Publications Co](#). The publicity has reached many thousands of influencers, potential employees, and customers.

The [Fog of SPA](#) is my most recent publication and presentation. It shares seven hard-won lessons from designing and delivering five commercial SPAs. It shows how design for quality, creative destruction, minimalism, and targeted testing techniques can remove complexity and confusion from SPA development.

16

Received award-winning industrial design with full-stack Single Page web Application (SPA) development.



Contributor on a number of open source projects, notably

[TaffyDB](#). Here is my [GitHub](#) account for your perusal.

Published a number of [jQuery plugins](#), including:

- [uriAnchor](#) - Sophisticated SPA state management in the URI.
- [event.gevent](#) - Custom global events for data binding.
- [event.ue](#) - Unified desktop and touch events.

Researched many tools and emerging technologies such as [SVG](#) and [mobile](#) interfaces. Employed technologies such as frame rate compensation, proximity

Michael S. Mikowski - Mozilla Firefox

Michael S. Mikowski



Author of [Single Page Web Applications - JavaScript end-to-end](#) for Manning Publications Co.

with around 3,000 MEAP (Manning Early Access Program) sales. Final print and electronic publication is scheduled for June 2013. Experience blends award-winning industrial design with full-stack Single Page web Application (SPA) development.



16% 16:48

Contributor on a number of open source projects, notably [TaffyDB](#). Here is my [GitHub](#) account for your perusal. Published a number of [jQuery plugins](#), including:

- [uriAnchor](#) - Sophisticated SPA state management in the URI.
- [event.gevent](#) - Custom global events for data binding.
- [event.ue](#) - Unified desktop and touch events.

Researched many tools and emerging technologies such as [SVG](#) and [mobile](#) interfaces.

Employed technologies such as frame rate compensation, proximity sensing, collision detection, and audio design to bring a highly engaging user experience to enterprise software. Researched and employed [Node.js](#) and [MongoDB](#). Have many years experience developing software for, and managing, large (100+ server) [Linux clusters](#) that served billions of web transactions per week using [MySQL](#). My favorite SCMS is [git](#) but also administered [Subversion](#), [Perforce](#), and [CVS](#).

Presented to thousands of people over the last nine months, including three presentations at the last two [HTML5 developer conferences](#), [Developer week](#), [University of San Francisco](#), and [Trendulate](#). All but one of the public presentations had an audience substantially over capacity<sup>1</sup> As a result I have many followers on [Google+](#), [LinkedIn](#), and [Manning Publications Co](#). The publicity has reached many thousands of influencers, potential employees, and customers.

[The Fog of SPA](#) is my most recent publication and presentation. It shares seven hard-won lessons from designing and delivering five commercial SPAs. It shows how design for quality, creative destruction, minimalism, and targeted testing techniques can remove complexity and confusion from SPA development.

## References

References are listed from most recent interactions. I respect the privacy of my references, and therefore please connect via [LinkedIn](#) if you wish to acquire contact details.

- [Gaurav Dhillon, CEO, SnapLogic Inc.](#)

Guarav is the founder of Informatica and SnapLogic, and a great fan of product design in any field. Because of his position, his availability is limited.

- [John Schuster, former VP of Engineering, SnapLogic Inc.](#)

I reported to John during the initial design and research for the next-generation product. He is a very experienced manager who understands how to strategically align near-term requirements with long-term vision — perhaps anyone I've ever worked for. He was supportive and insightful and let his knowledge workers flourish with them. He rarely showed a strong opinions about our future design, but when he did he showed very good

What do you think of this page?

- It is too boastful
- I find it useful
- It should be more boastful



Unified desktop and touch events.

any tools and emerging technologies such as **SVG** and **mobile** interfaces. Employed technologies such as frame rate compensation, proximity sensing, collision detection, and audio design to bring a highly engaging user experience to enterprise software. Researched and employed **Node.js** experience developing software for, and managing, large (100+ server) **Linux clusters** that week using **MySQL**. My favorite SCMS is **git** but also administered **Subversion**, **Perforce**, and

## (2b) Multi-choice - Zoom

A screenshot of a Firefox browser window titled "Michael S. Mikowski - Mozilla Firefox". The main content area displays a list of bullet points:

- **unAnchor** - Sophisticated SPA state management in the UI.
- **event.gevent** - Custom global events for data binding.
- **event.ue** - Unified desktop and touch events.

The text below the list reads:

**Researched** many tools and emerging technologies such as **SVG** and **mobile** interfaces. Employed technologies such as frame rate compensation, proximity sensing, collision detection, and audio design to bring a highly engaging user experience to enterprise software. Researched and employed **Node.js** experience developing software for, and managing, large (100+ server) **Linux clusters** that week using **MySQL**. My favorite SCMS is **git** but also administered **Subversion**, **Perforce**, and

Presented to thousands of people over the last nine months, including three presentations at **conferences**, **Developer week**, **University of San Francisco**, and **Trenduate**. All but one of the audience substantially over capacity<sup>1</sup>. As a result, my **publicity** has reached many thousands of influencers.

**The Fog of SPA** is my most recent publication. It shows how design for five commercial SPAs. It shows how design for remove complexity and confusion from SPA development.

**References**

References are listed from most recent interactions.

A dark overlay window is visible in the bottom right corner, containing the text "What do you think of this page?" and three radio button options:

- It is too boastful
- I find it useful
- It should be more boastful

Presented to thousands of people over the last nine months, including three presentations at the last two **HTML5 developer conferences**, **Developer week**, **University of San Francisco**, and **Trendulate**. All but one of the public presentations had an audience substantially over capacity<sup>1</sup> As a result I have many followers on **Google+**, **LinkedIn**, and **Manning Publications Co**. The publicity has reached many thousands of influencers, potential employees, and customers.

**The Fog of SPA** is my most recent publication and presentation. It shares seven hard-won lessons from designing and delivering five commercial SPAs. It shows how design for quality, creative destruction, minimalism, and targeted testing techniques can remove complexity and confusion from SPA development.

# (3) Rating Question

Michael S. Mikowski - Mozilla Firefox

Michael S. Mikowski

Saving screenshot...

Highly engaging and effective. I have been using this technique for many years. I experience developing software for, and managing, large (100+ server) Linux clusters that served billions of web transactions per week using **MySQL**. My favorite SCMS is **git** but also administered **Subversion**, **Perforce**, and **CVS**.

Presented to thousands of people over the last nine months, including three presentations at the last two **HTML5 developer conferences**, **Developer week**, **University of San Francisco**, and **Trendulate**. All but one of the public presentations had an audience substantially over capacity<sup>1</sup> As a result I have many followers on **Google+**, **LinkedIn**, and **Manning Publications Co**. The publicity has reached many thousands of influencers, potential employees, and customers.

**The Fog of SPA** is my most recent publication and presentation. It shares seven hard-won lessons from designing and delivering five commercial SPAs. It shows how design for quality, creative destruction, minimalism, and targeted testing techniques can remove complexity and confusion from SPA development.

**References**

References are listed from most recent interactions. I respect the privacy of my references, and therefore please connect via **LinkedIn** if you wish to acquire contact details.

- Gaurav Dihlon, CEO, SnapLogic Inc.**  
Gaurav is the founder of Informatica and SnapLogic, and a great fan of product design in any field. Because of the demands of his position, his availability is limited.
- John Schuster, former VP of Engineering, SnapLogic Inc.**  
I reported to John during the initial design and research for the next-generation product. He is a very experienced senior manager who understands how to strategically align near-term requirements with long-term vision — perhaps better than anyone I've ever worked for. He was supportive and insightful and let his knowledge workers flourish without competing with them. He rarely showed a strong opinions about our future design, but when he did he showed very good sense of what was needed in the market. I would very much appreciate a chance to work with him again.
- Steve Goodwin, Senior UI Developer, SnapLogic Inc.**  
Steve is a senior member of the team I managed. He is very experienced with full-stack development, and always has creative ideas to solve almost any challenge. He is also cool under pressure, a great communicator, and a pleasure to work with.
- Joyce Lam, Senior UI Developer, SnapLogic Inc.**  
Joyce is a senior member of the team I managed. She is also very experienced with full-stack development and has an especially good eye for elegant design. She has a great sense of humor and is a great collaborator.
- Michael Lorton, VP of Engineer, GameCrush**  
Michael is the smartest manager I have ever worked for. Besides his deft technical skills, he has excellent product sense. I really enjoyed working for him and learned a lot.
- Anees Iqbal, CEO, GameCrush**  
Anees has the intelligence, experience, and pragmatism to work on any aspect of a project that is required for success. He is always accessible and helpful, and I found him a superb CEO.

How effective do you think this page is?

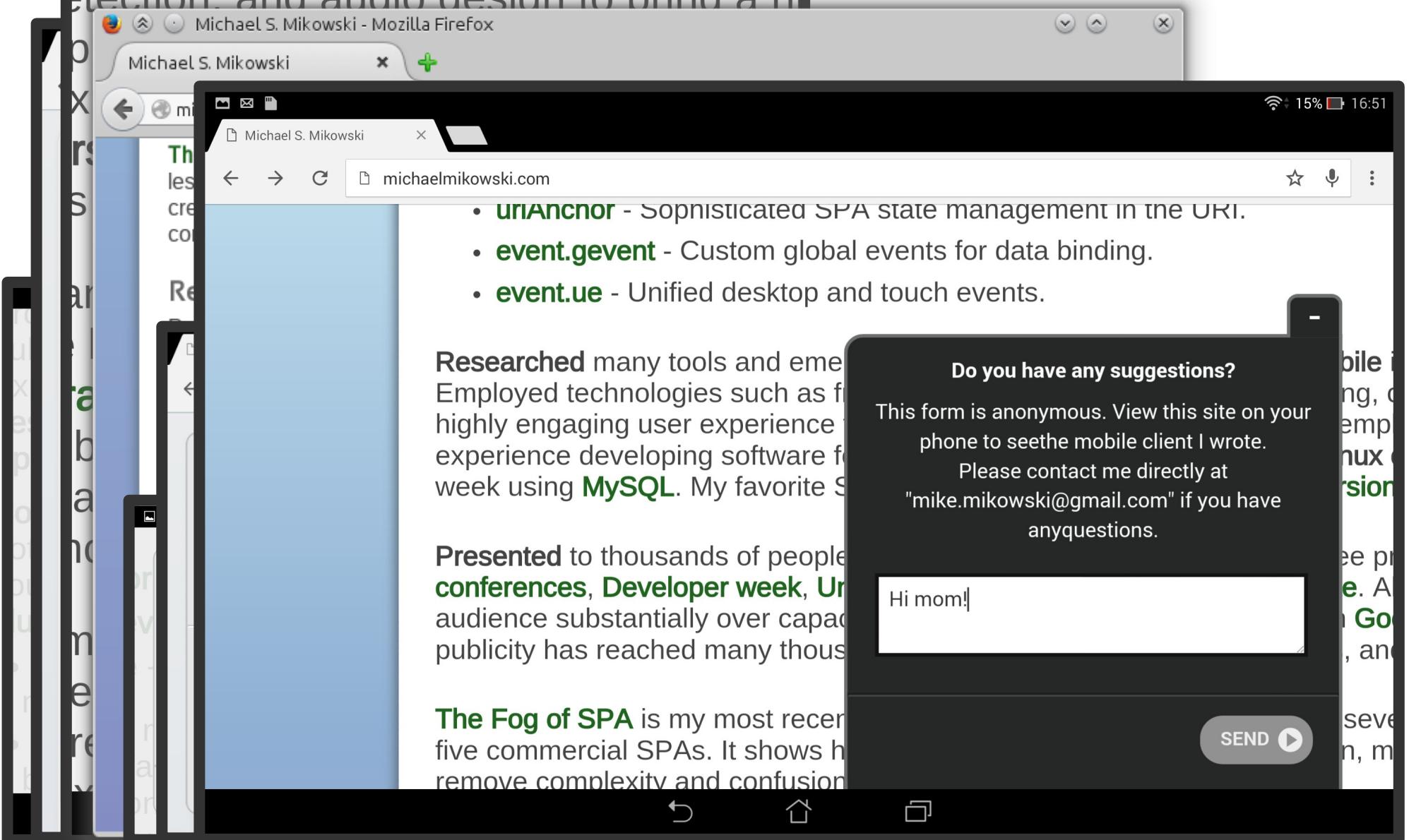
In promoting the author

0 1 2 3 4 5 6 7 8 9 10

Not likely      Most likely

ed desktop and touch events.

# (4) Text input



# (5) And thank you

A screenshot of a Firefox browser window. The address bar shows 'michaelmikowski.com'. The main content area displays a page with text about technologies used and a 'Thank you!' message.

Employed technologies such as frame rate compensation, proximity sensing, and audio design to bring a highly engaging user experience to enterprise software. Researched and developed software for, and managed, large (100+ server) systems. Worked billions of web transactions per week using MySQL. My favorite SCMS is git but also administered Subversion and Git.

Presented to thousands of people at conferences, Developer weeks, and user groups. Audience substantially over 100,000. Publicity has reached many news sites and forums.

The Fog of SPA is my most popular article, with over 5 million views. It shows how to build single-page applications using Backbone.js and jQuery. It removes complexity and confusion from building SPAs.

**Thank you!**

[Check out my jQuery plugins](#)

# Classic CSS hierarchy

Web 2.0 called and wants it strategy back

- Site rules
- Regional rules
- Section rules
- Page rules
- Device-class rules
- Specific device rules  
(iPhone, HD laptops, other popular outliers)

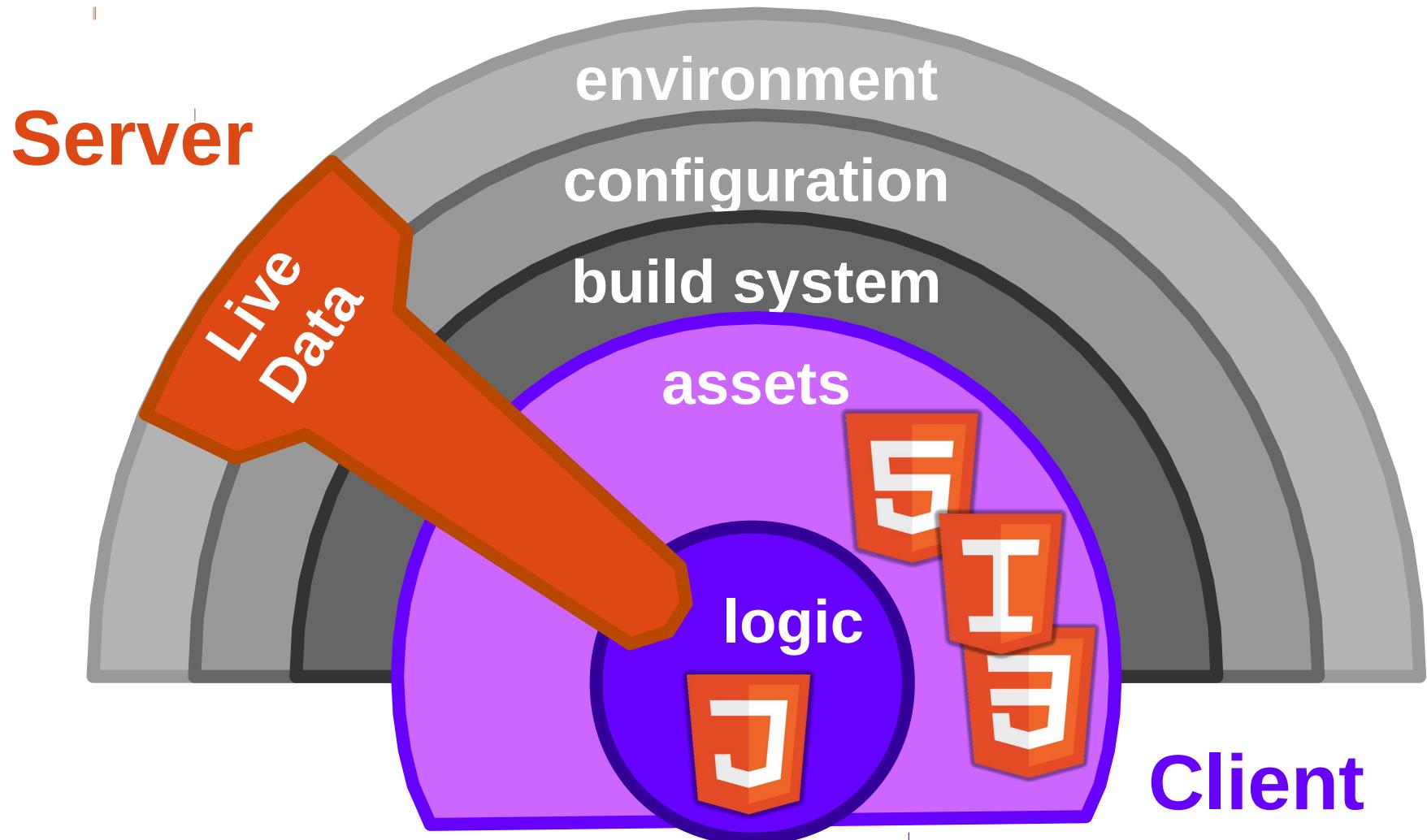
# But what about run-time data?

Static styles are at best a kludge

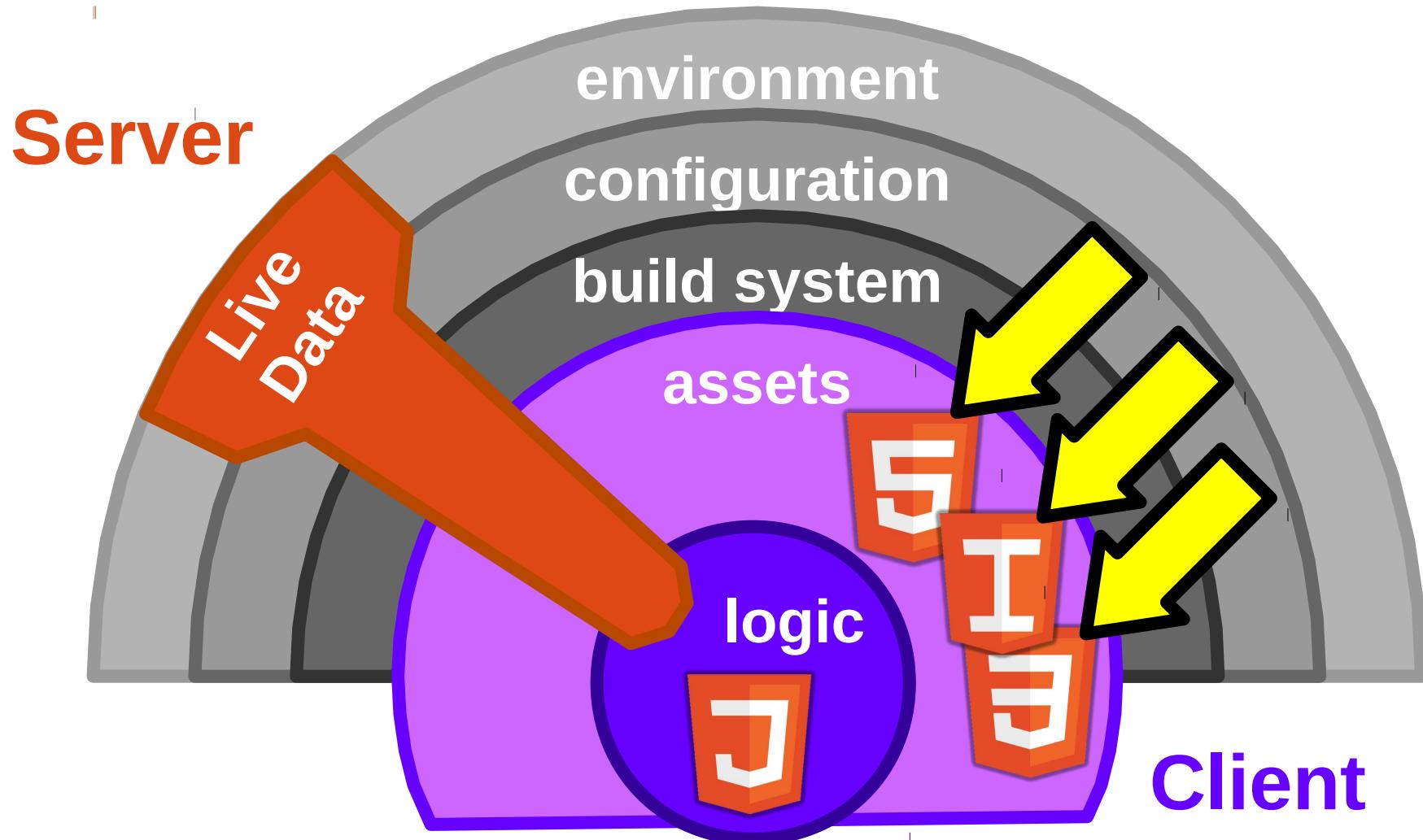
- Device capabilities?
- Orientation? Number of touch points?
- Zoom level and Pan?
- Ambient temperature? Ambient Light?
- Heartbeat? Acceleration?
- Time of day? Geo location?
- Stock market performance?
- Live RSS feed?

# The REAL problem

## Typical web app structure

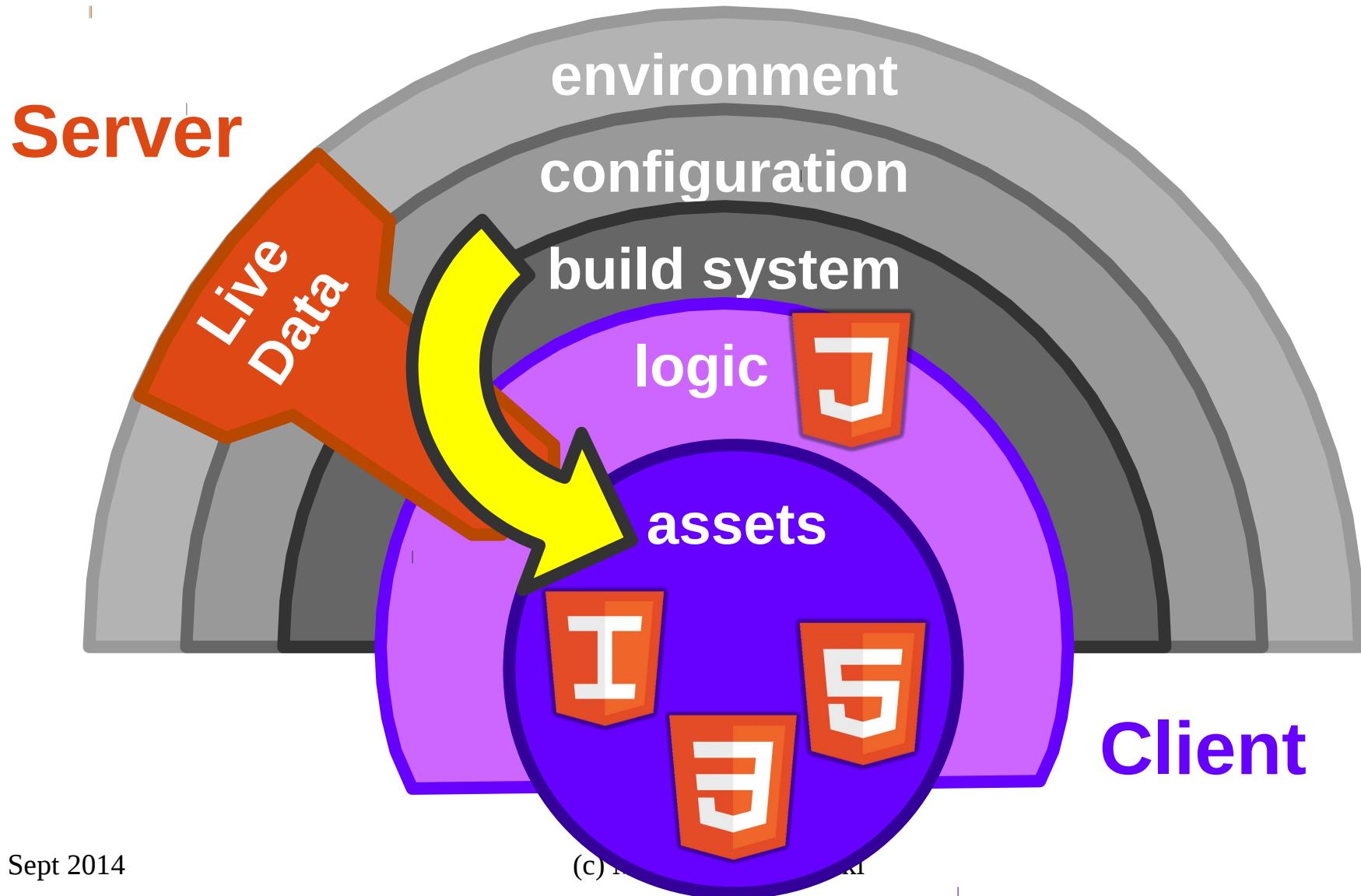


# Assets are controlled OUTSIDE of Logic by Config and Build System



# FIX the problem!

## Create assets with Client as needed!



# Create CSS for every visitor!

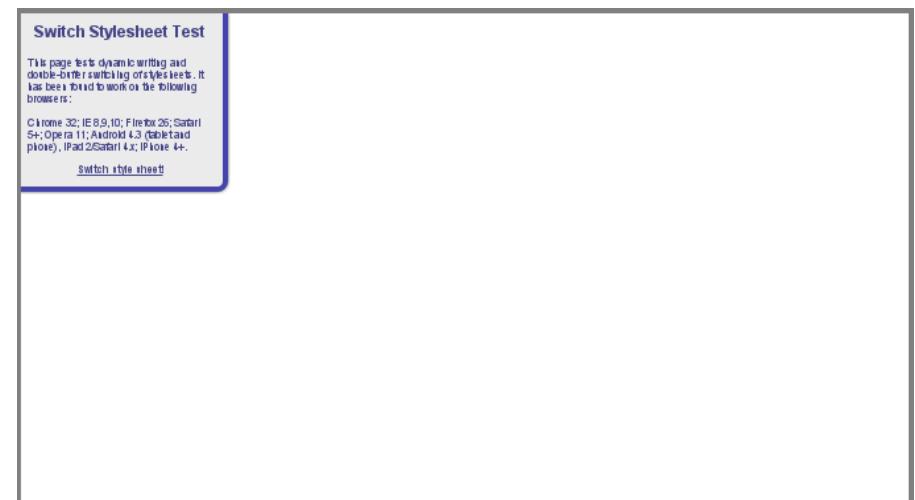
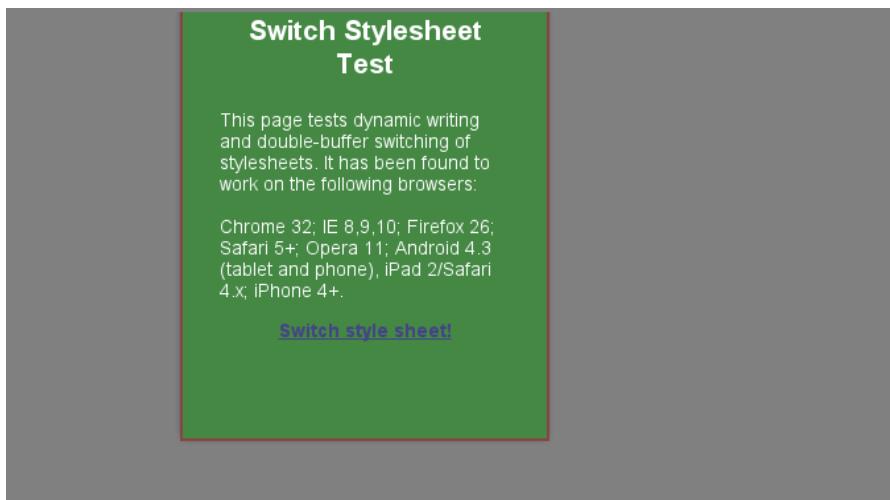
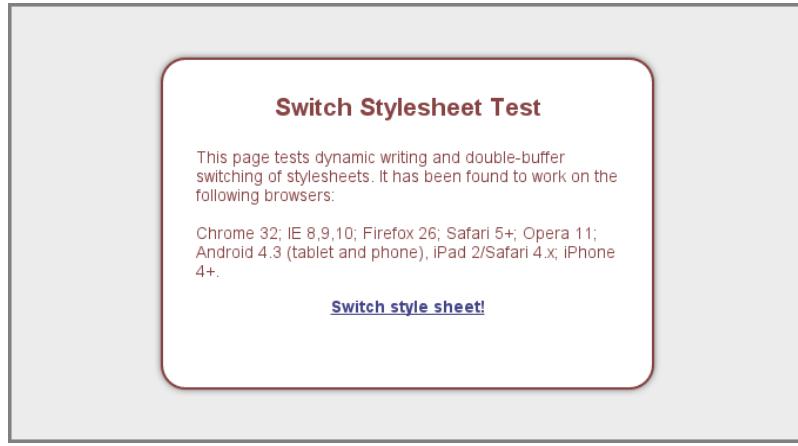
Use any dynamic data to drive style

- Ambient temperature?
- Heartbeat?
- Time of day?
- Stock market performance?
- Users mood?

**We create > 100,000,000 stylesheets every day tailored for every customer!**

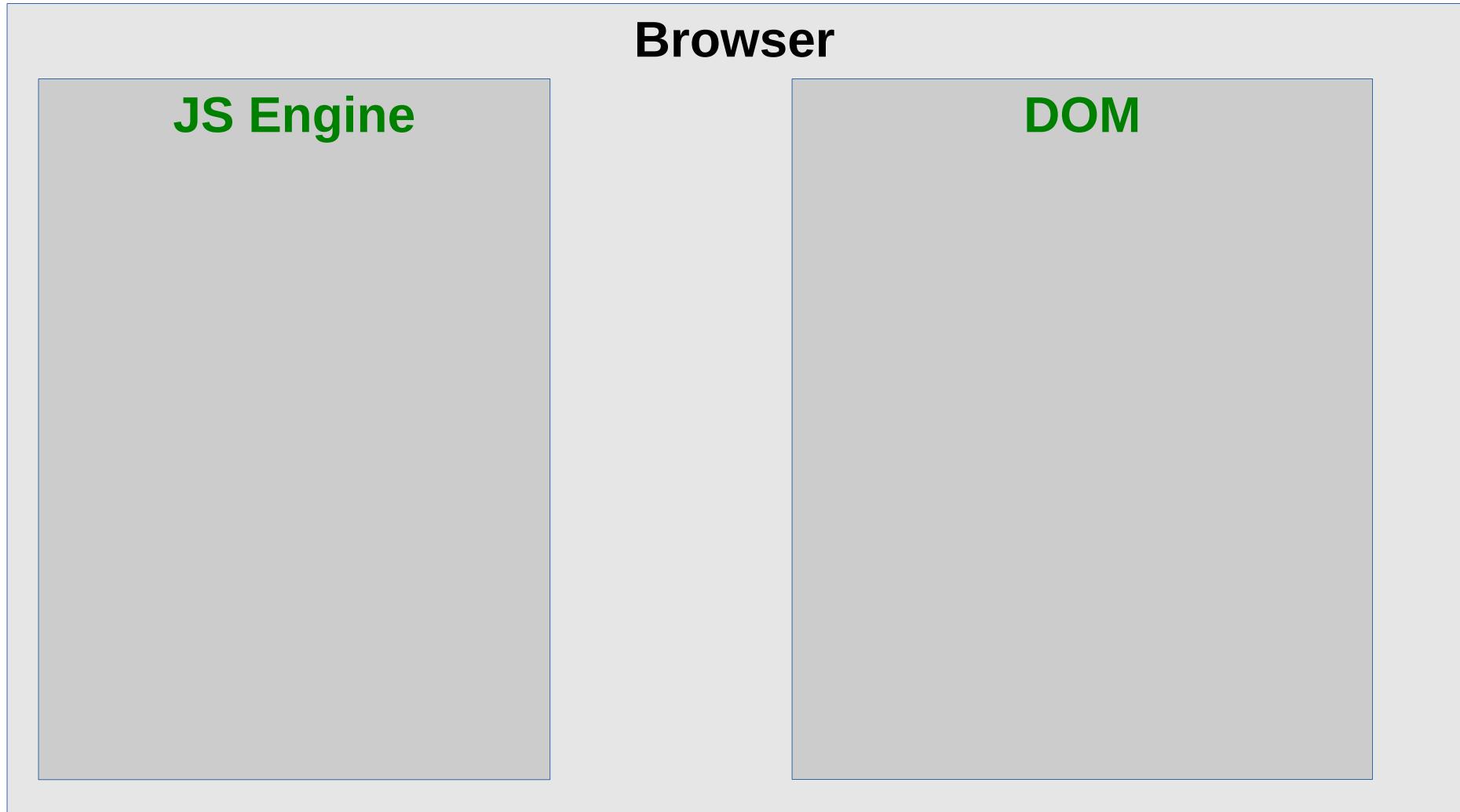
# Remember our JS-CSS example

## Uses JS-CSS Style Manager



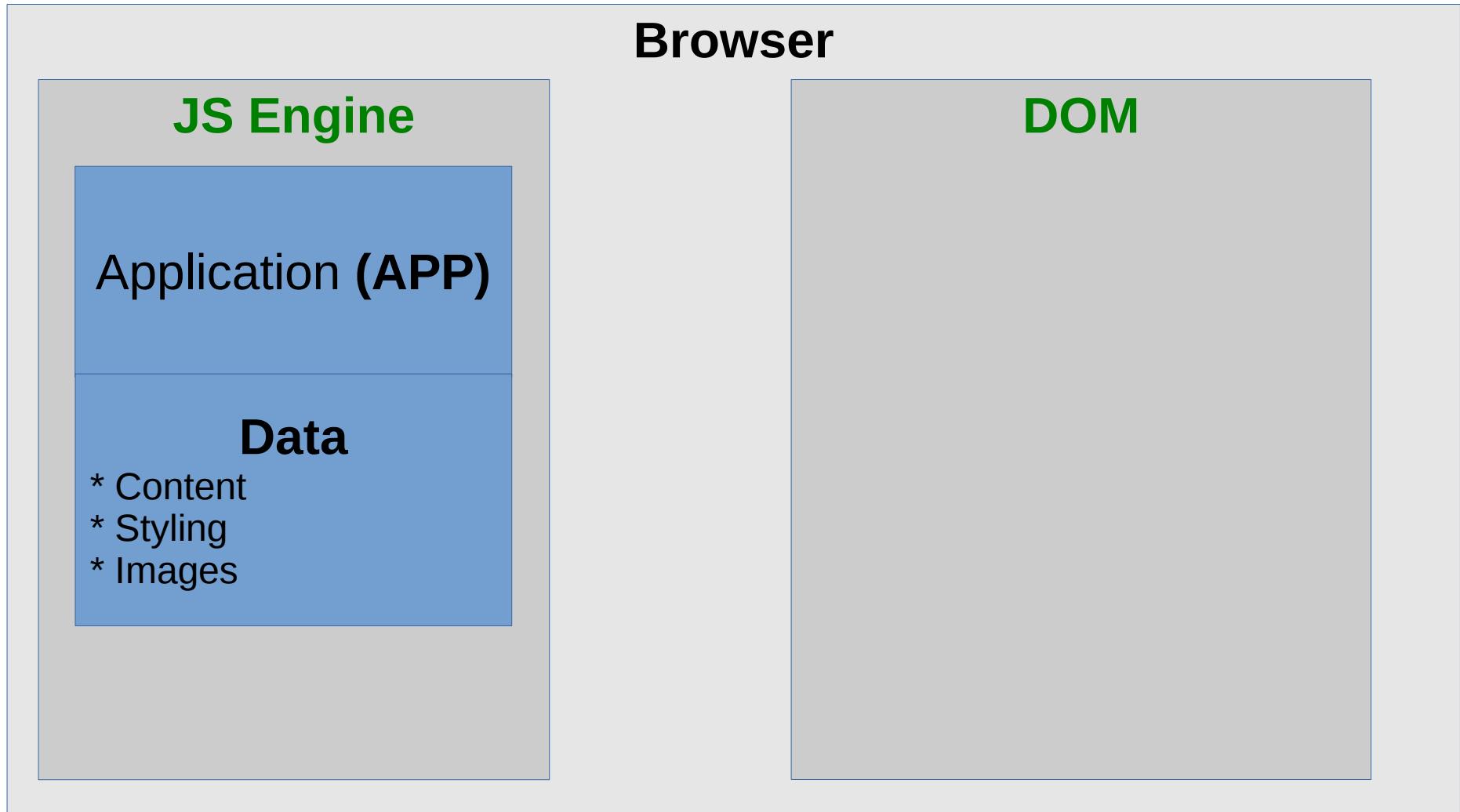
# How it works

A user is viewing a web page with a browser. The web page uses our App



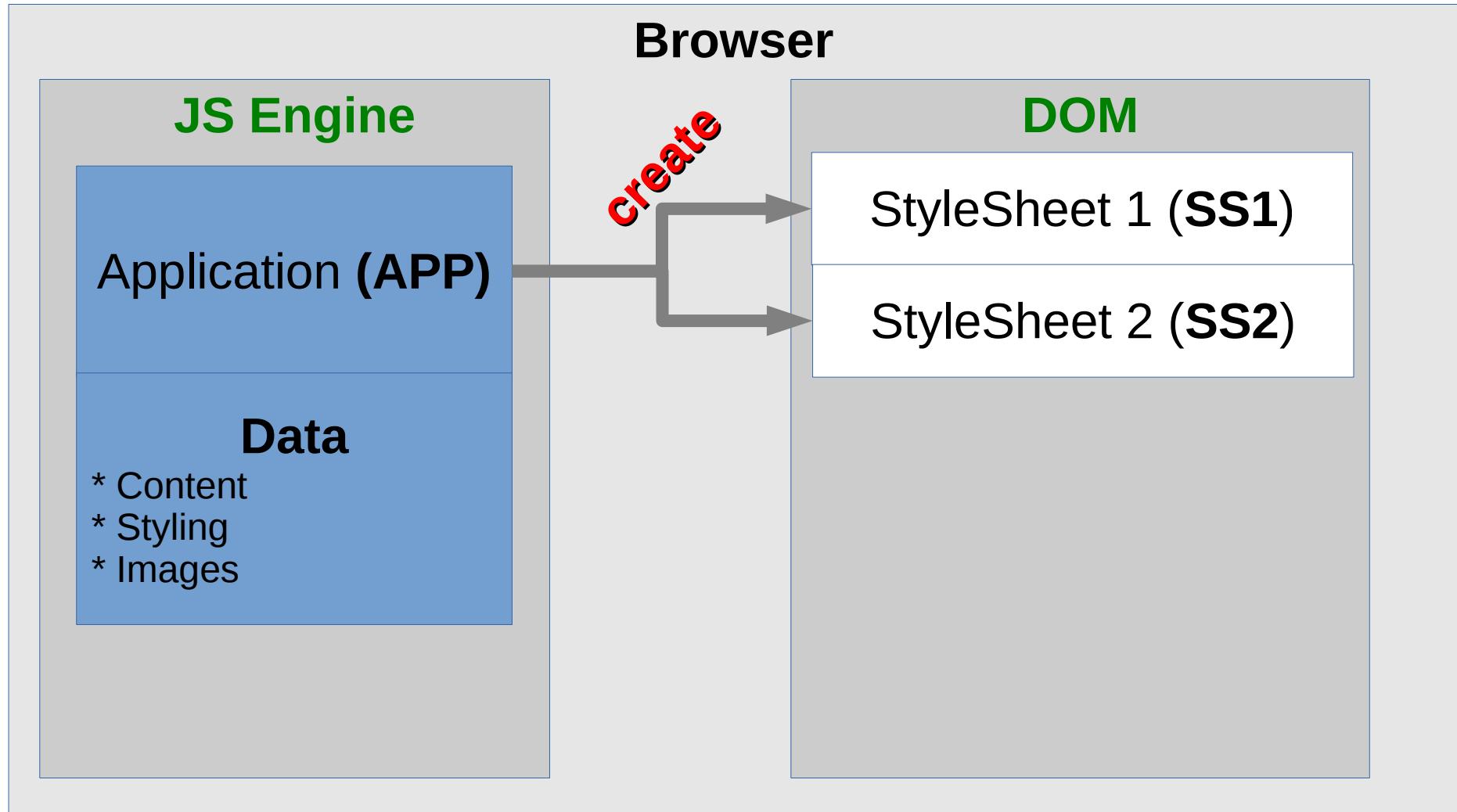
# How it works

And then a script tag loads our single package of Application and Data



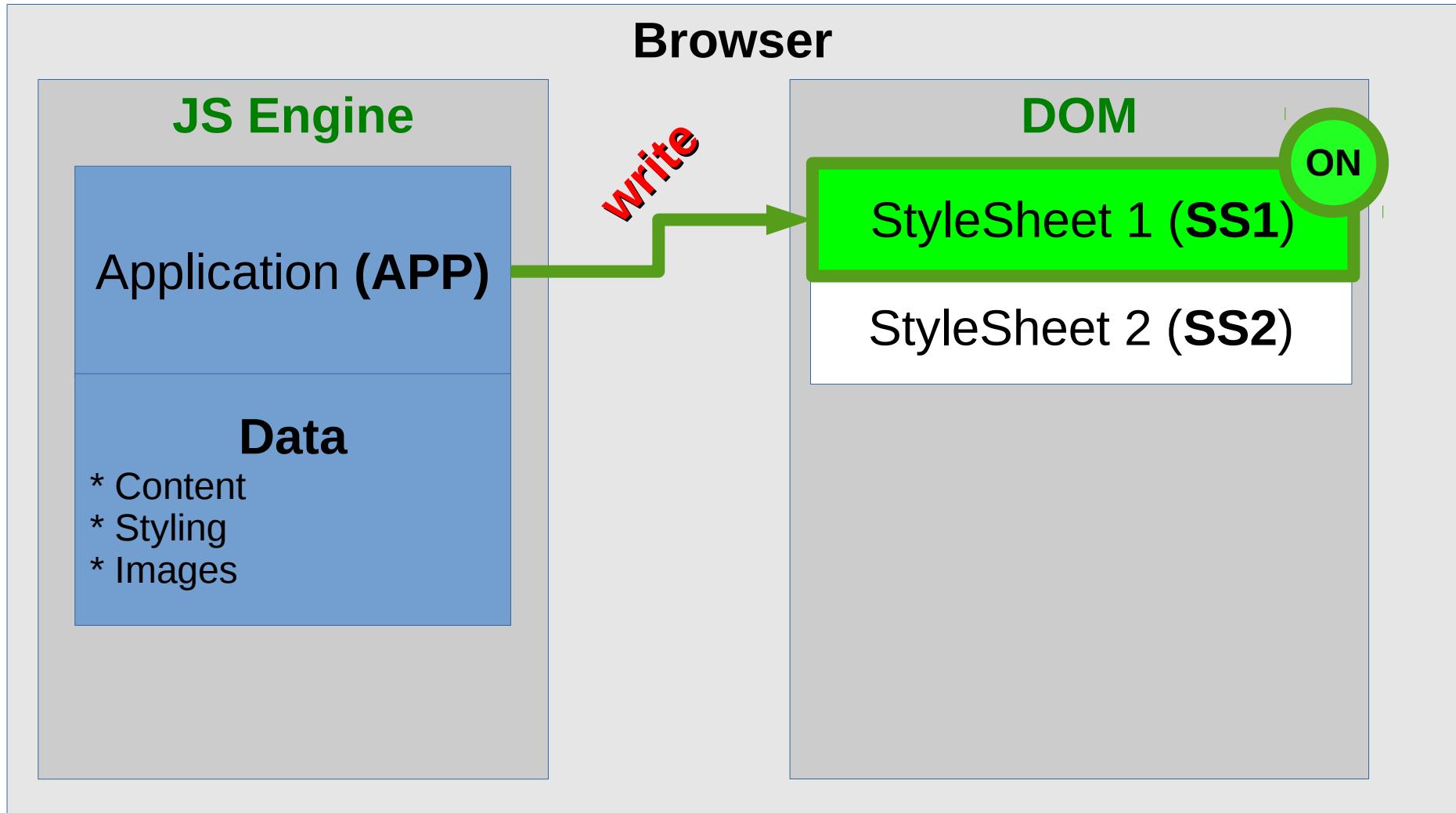
# How it works

1. On initialization, the JS APP creates two empty StyleSheets – **SS1** and **SS2**.



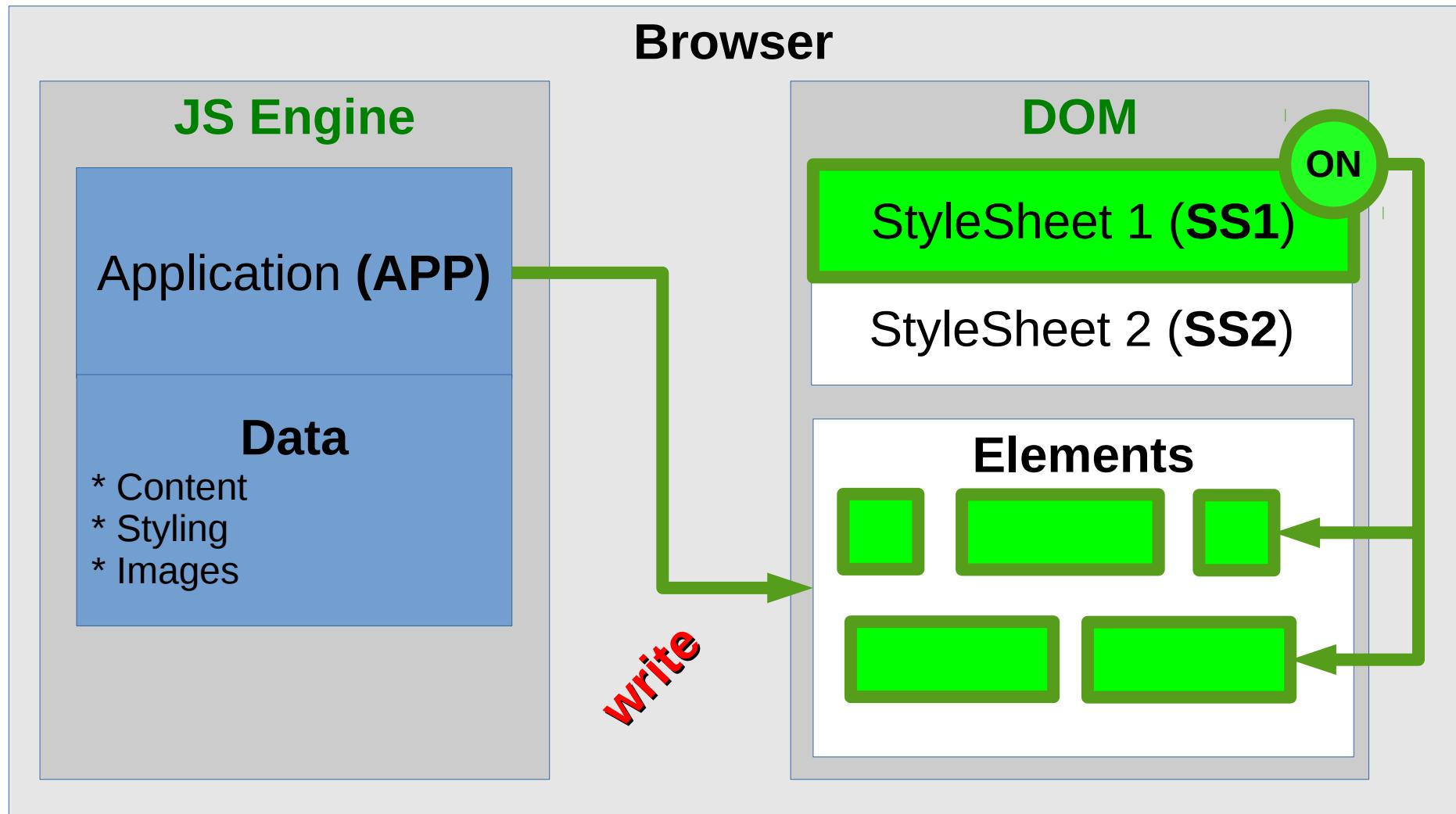
# How it works

2. On DOM ready, APP calculates and writes CSS to **SS1** and **enables** it.



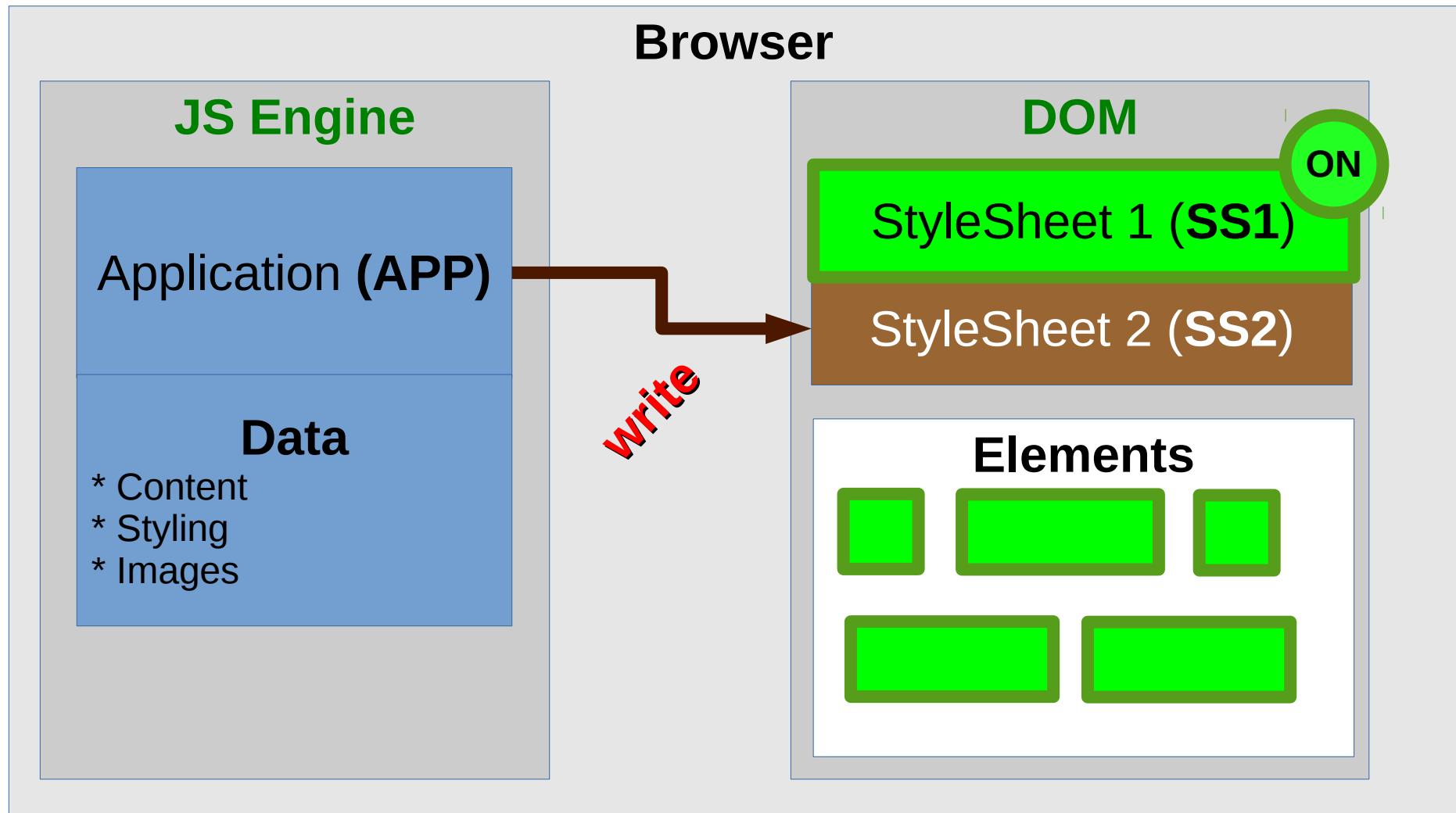
# How it works

3. APP then writes elements to the DOM which are styled by SS1.



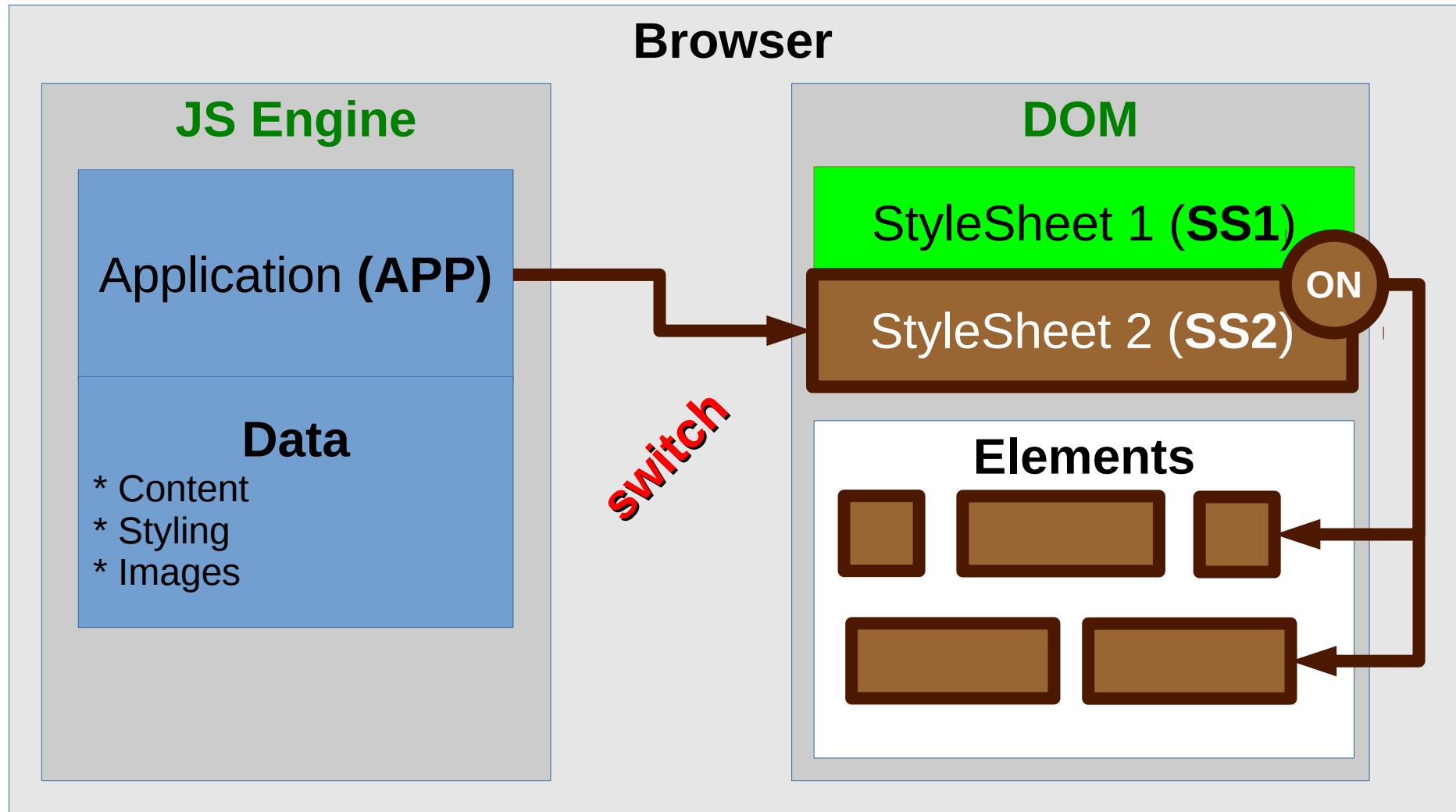
# How it works

- When APP needs to change styles, it starts by calculating and writing CSS to SS2.



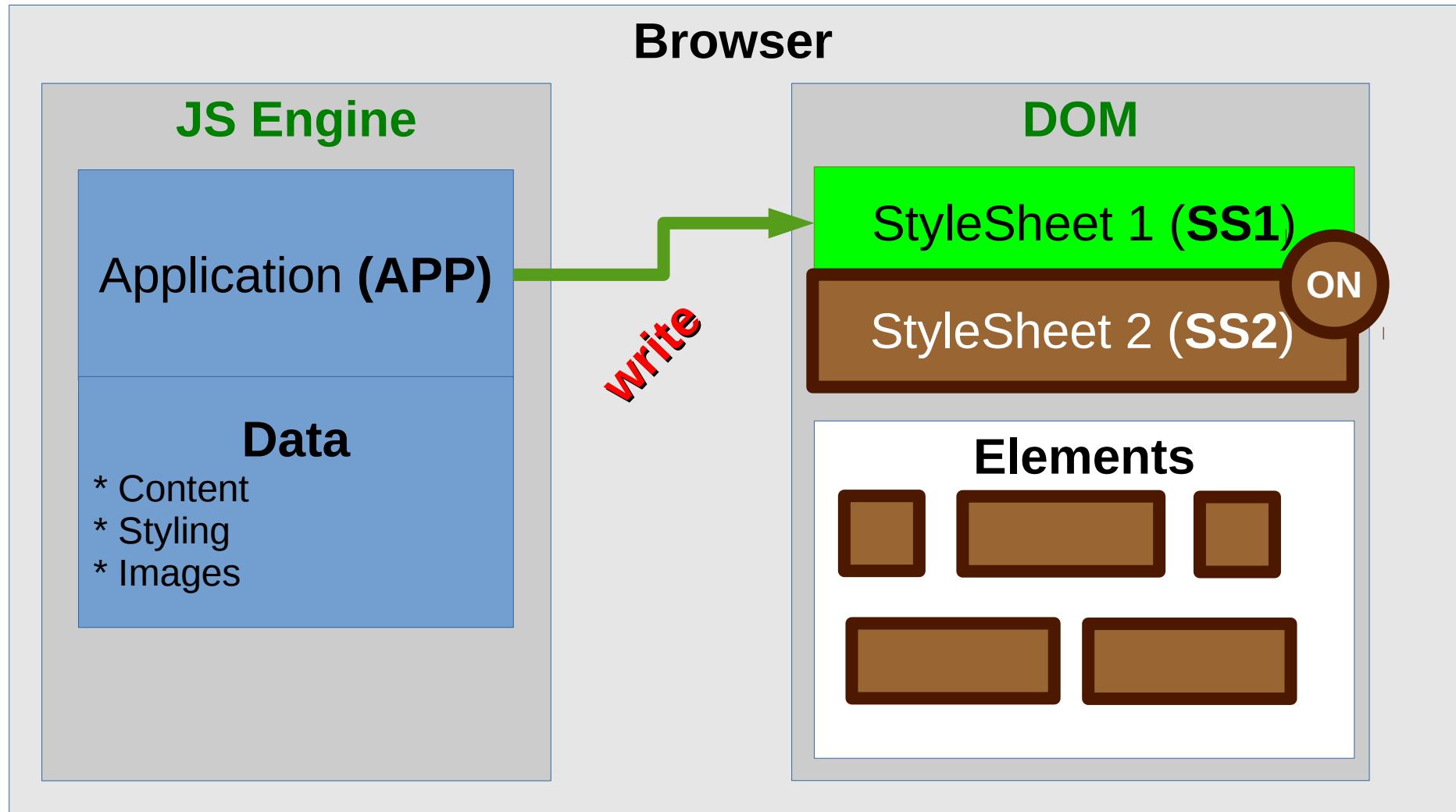
# How it works

5. When APP has completed writing SS2, it disables SS1 and enables SS2.



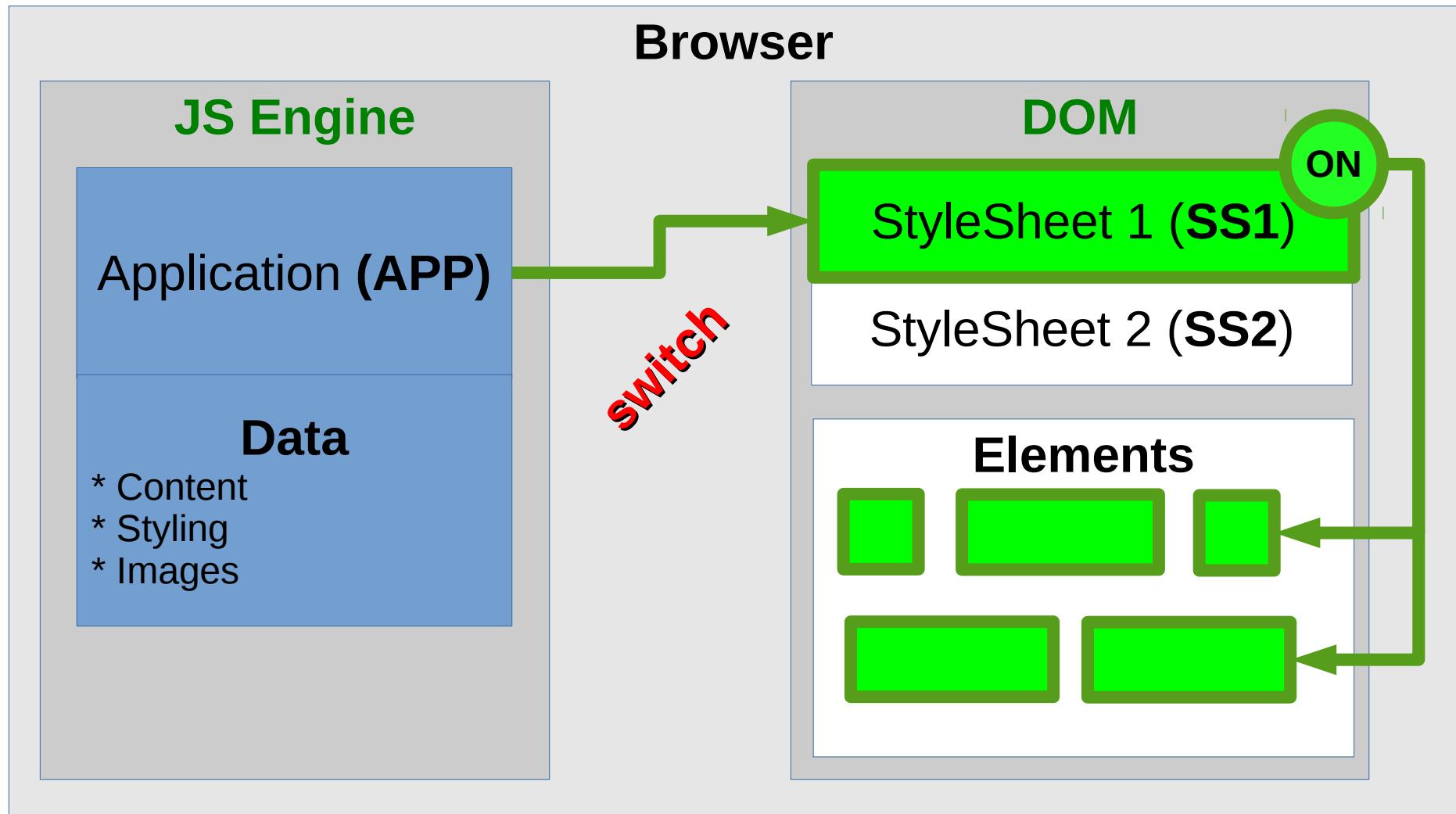
# How it works

6. When APP needs to change styles again, it overwrites SS1



# How it works

6. When APP needs to change styles again, it overwrites SS1



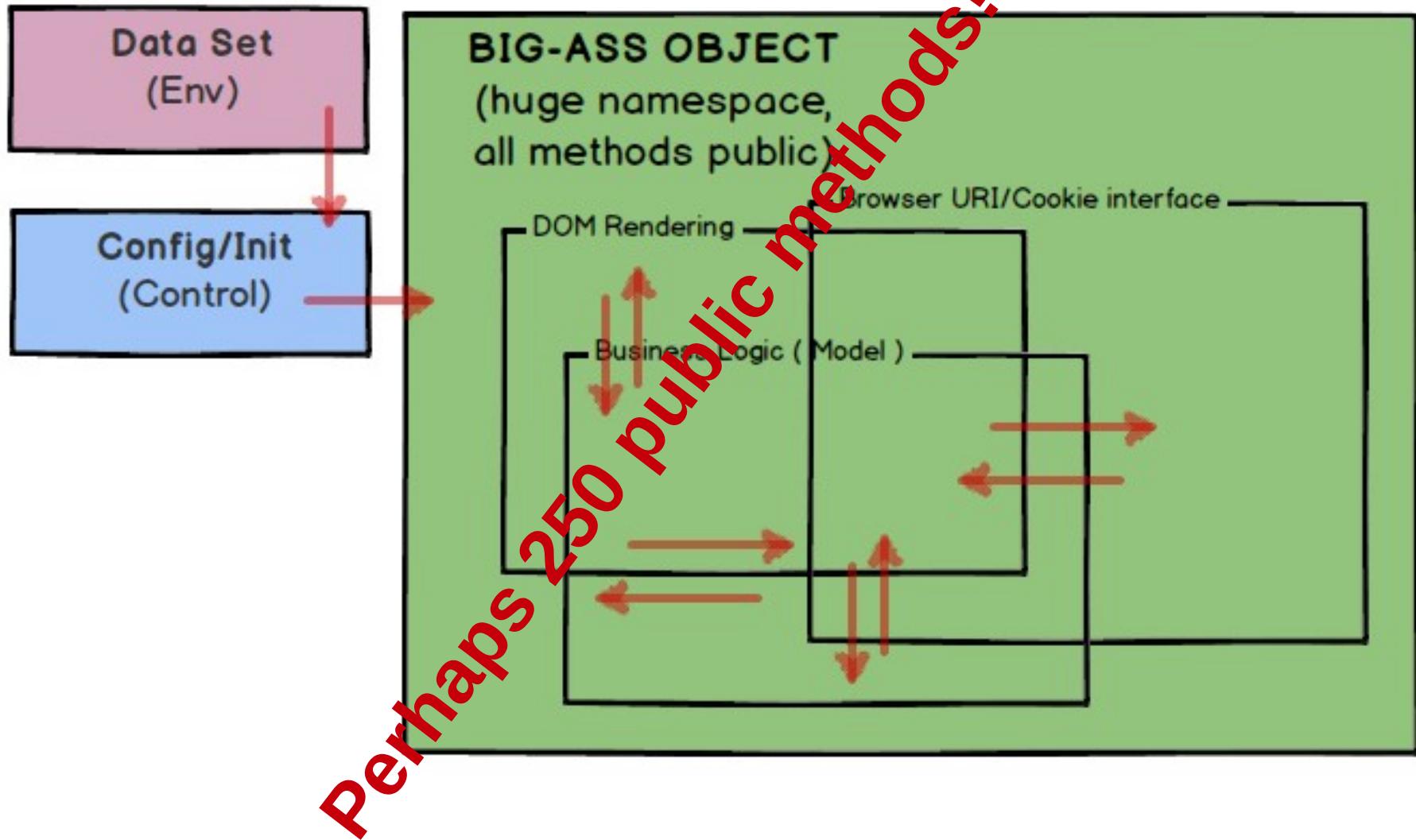
# Remember Qualaroo?

Over 100 million end users every day

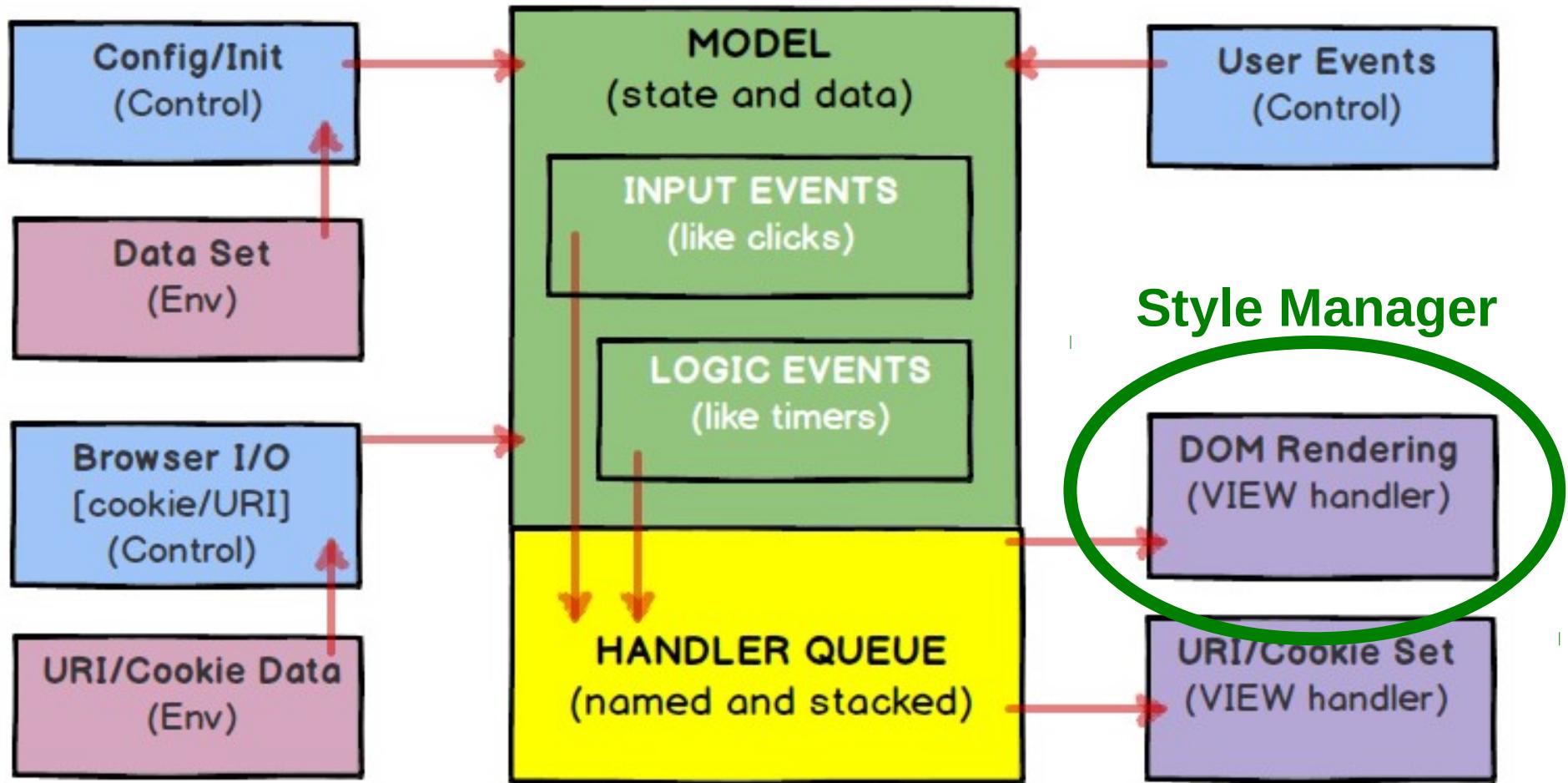
- We write a custom CSS for every customer
- Using JS-CSS Style Manager
- We first employed on Mobile Client, then unified clients



# Our 3<sup>rd</sup> party app arch was:



# And changed to this: 3 public methods!



# Compelling results

- Single style changes can be applied “in-situ”
- Larger changes are ***much faster*** with double buffering
  - The DOM must be scanned for each change
  - Fifty changes times 1,000 elems = 50,000 elem scans
- We cascade “in software” instead of in DOM
- Cleaner, faster, easier to debug
- Better UX – less styling “stutter”

# What we gain with JS-CSS

- Allows us to load the entire 3<sup>rd</sup> party application and data in a single HTTP request
  - Application logic, API
  - Content including images
  - Styling, including dynamic styles
- Much less CSS bloat
- Faster (Cascade in “software”)
- Same model controls content ***and*** styling
- “Do the impossible” - millions of styling combinations

# What we miss from {less} or Sass

- Less safety and built-in wisdom
  - {less} and Sass have been around a very long time
  - JS-CSS as we employed does have *some* guardrails
- Browser bugs
  - Weird stuff *can* and *does* happen
  - Older browsers can't cope
  - We support IE9+, Firefox 28+, Chrome 28+, Opera, Safari 6+

# Questions?

About Dynamic CSS Manipulation  
*Powered by JavaScript*

