

Web UI

Survey of Front End Technologies

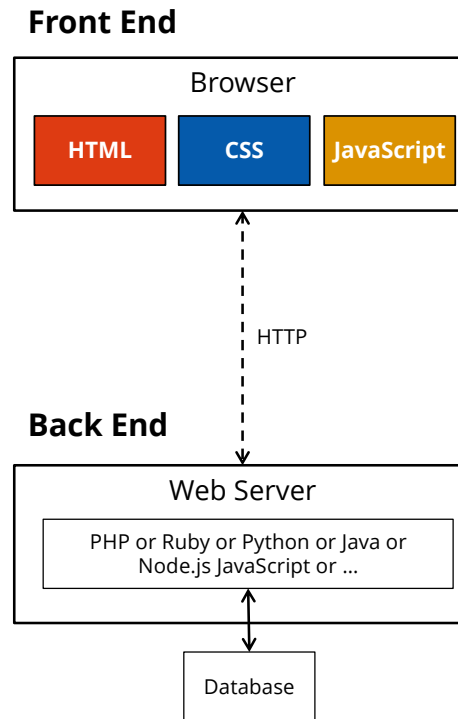
Web Challenges and Constraints

- Desktop and mobile devices
 - mouse vs. touch input, big vs. small screen
- Highly variable runtime environment
 - different OS, different browsers and version, user settings
- Distributed system
 - managing state, handling latency and lost connections, ...
- Evolving standards
 - best practices, specifications, frameworks, tools
- Legal and marketing
 - search engine optimization, copyright, mandated accessibility



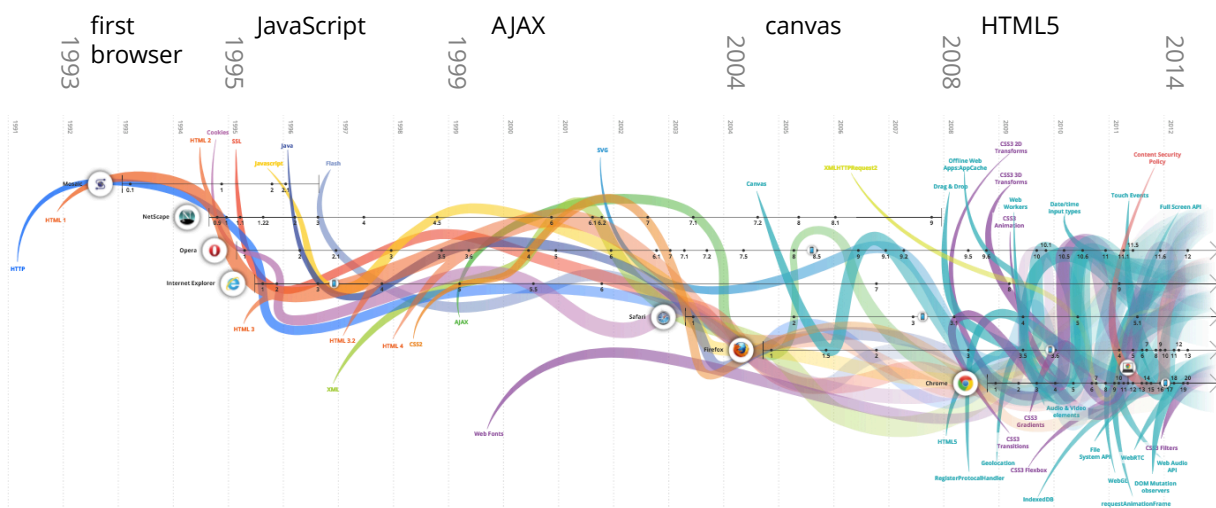
Web Architecture

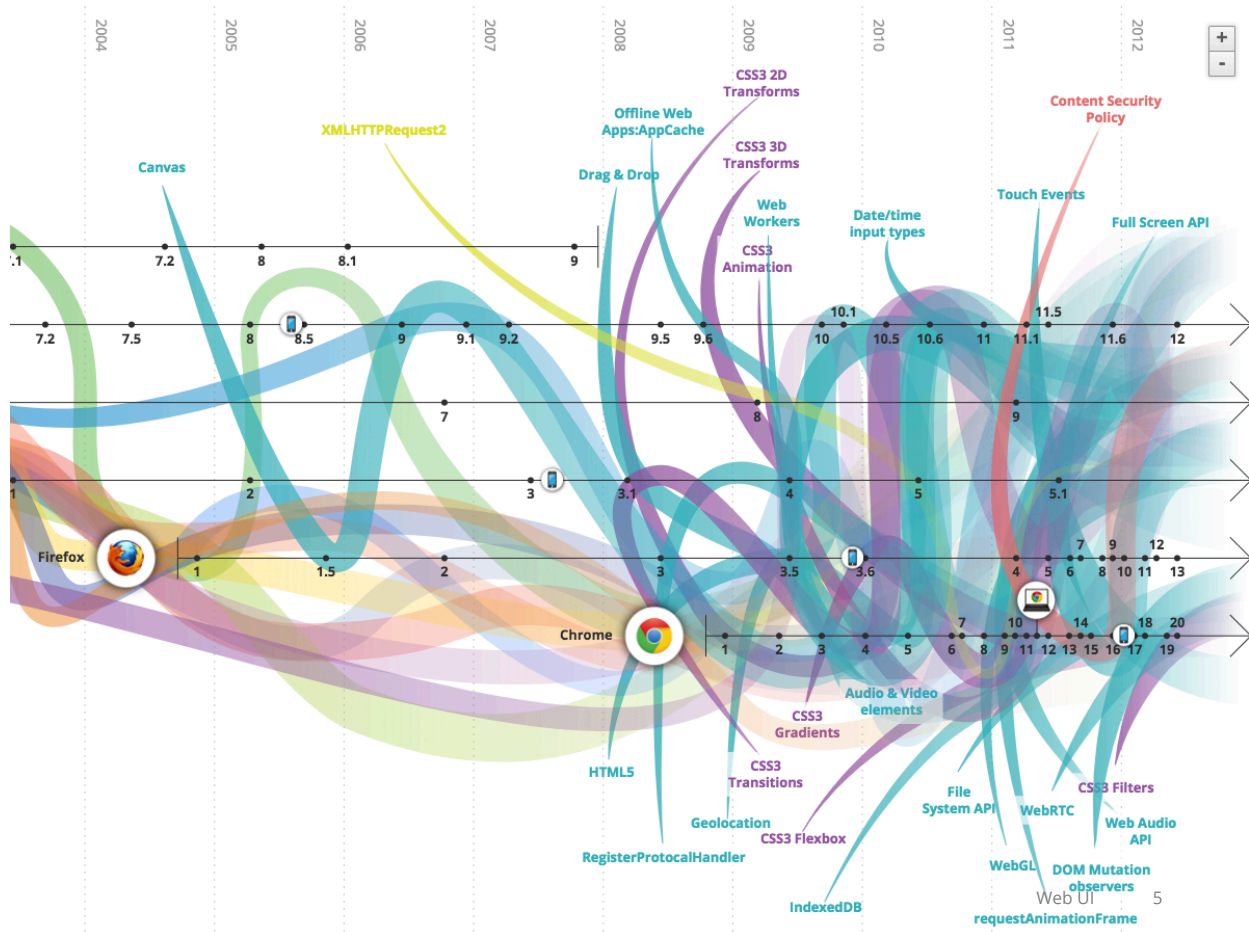
- HTML
 - DOM
 - canvas
- CSS
 - media queries
 - frameworks
 - pre-processors
- JavaScript
 - language
 - frameworks
 - transcompilers



Web Development Technology Timeline (up to 2014)

- <http://www.evolutionoftheweb.com/#/evolution/day>

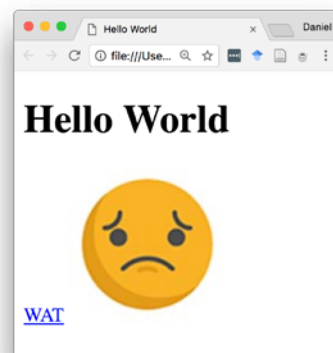




Hypertext Mark-up Language (HTML) (~1993)

- Language for describing the structure of Web pages

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>Hello World</title>
  </head>
  <body>
    <h1>Hello World</h1>
    <p>
      <a href="http://www.wat.com">WAT</a>
      
    </p>
  </body>
</html>
```



HTML is a W3C Standard

- World Wide Web Consortium (W3C)
 - HTML 5.1 2nd Edition (W3C Recommendation 3 October 2017)
 - <https://www.w3.org/TR/2017/REC-html51-20171003/>
- Elements
 - metadata tags: <head>, <title>, <style>, ...
 - section tags: <body>, <nav>, <header>, <h1>, <h2>, ...
 - grouping tags: <p>, , , , <figure>, <div>, ...
 - text-level semantic tags: <a>, , , <code>, ...
 - embedded content tags: , <video>, ...
 - tabular data tags: <table>, <tbody>, <tr>, <td>, ...
 - form tags: <button>, <input>, <textarea>, ...

Tag Attributes

- Examples
 - Defining a hyperlink URL:
`WAT`
 - Defining an Image:
``

- Universal Attributes

Attribute	Meaning
id	Assigns a unique identifier to the element.
class	Assigns one or more classifications to the element.
style	Apply in-line CSS styles to the element.
title	Provide a title or advisory information about the element.

Document Object Model (DOM)

- defines the structure of the HTML document as well as the behavior of the objects it contains
- resembles an interactor tree

```
DOCTYPE: html
HTML lang="en"
  HEAD
    #text:
    META charset="utf-8"
    #text:
    TITLE
      #text: Hello World
    #text:
  #text:
  BODY
    #text:
    H1
      #text: Hello World
    #text:
    P
      #text:
      A href="http://www.wat.com"
        #text: WAT
      #text:
      IMG src="emoji.jpg" alt="Wor..."
      #text:
    #text:
```

Web UI 9

Cascading Style Sheets (CSS) (~1998)

- Separate presentation using CSS from content defined by HTML
- create “rules” by setting style properties on “selectors”
 - simple selectors (specific element type, class, id)
 - attribute selectors (elements with attribute types and values)
 - pseudo selectors (elements in a certain state, e.g. :hover)

CSS rule:

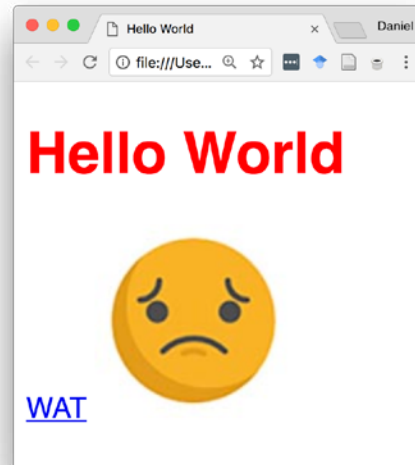
```
selector
h1 {
  color: red;
  font-size: 2rem;
}
```

property value

Web UI 10

CSS Example

```
body {  
  font-family: Helvetica, Arial, sans-serif;  
  font-size: 14pt;  
}  
  
h1 {  
  color: red;  
  font-size: 2rem;  
}  
  
p {  
  font-size: 1rem;  
}
```

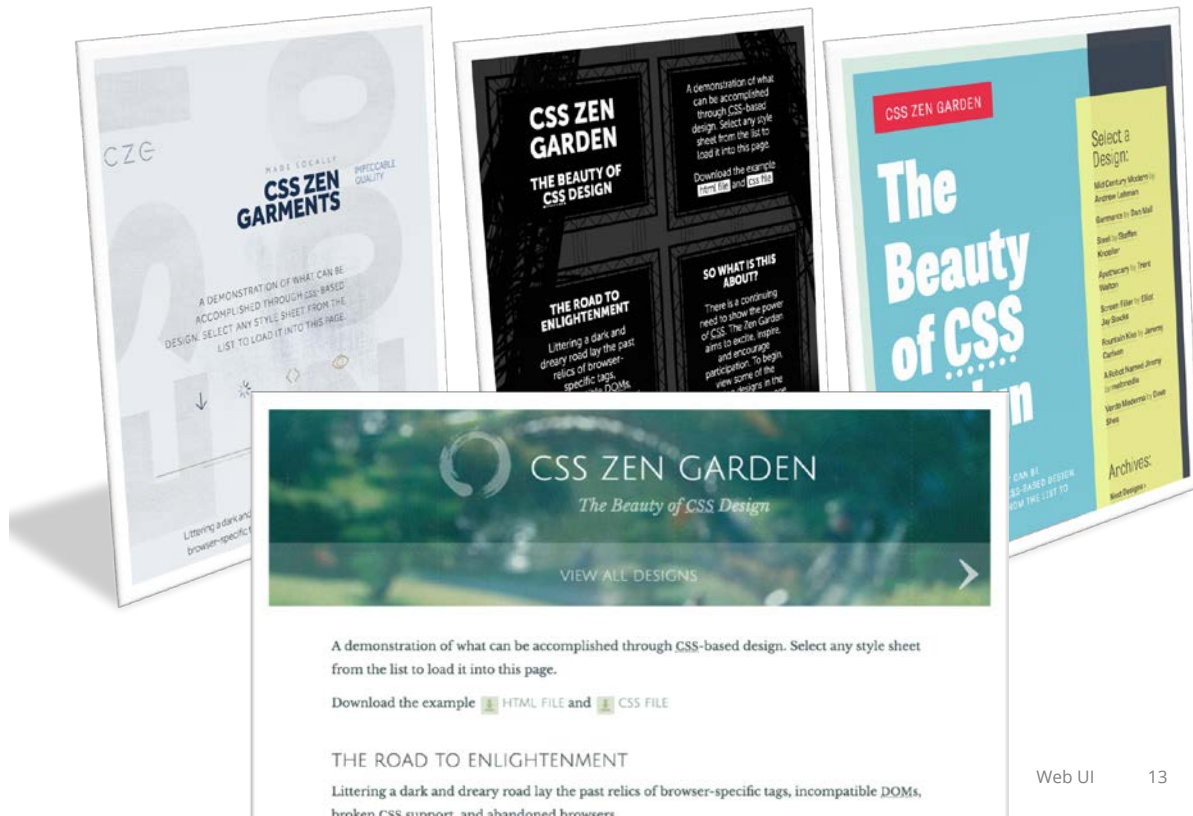


Web UI 11

CSS Inheritance and Cascade

- **Inheritance** in CSS is the mechanism for passing certain properties from a parent element down to its children
 - e.g. font property for <body> is inherited for <p>
- **Cascade** is mechanism that controls the end result when multiple, conflicting CSS declarations apply to the same element
 - e.g. font property for <p> can be overridden

Web UI 12



Web UI 13

CSS frameworks (~1997)

- A CSS file of pre-configured styling
 - (and related JavaScript, assets, ...)
- Popular frameworks
 - Meteor, Bulma, Pure, Materialize
 - **Bootstrap** (~2011)



```
<div class="container">
  <div class="row">
    <div class="col-sm">
      One of three columns
    </div>
    <div class="col-sm">
      One of three columns
    </div>
    <div class="col-sm">
      One of three columns
    </div>
  </div>
</div>
```

One of three columns
One of three columns
One of three columns

One of three columns	One of three columns	One of three columns
----------------------	----------------------	----------------------

Web UI 14

CSS Preprocessors (~2006)

- A CSS superset language that generates CSS
- Less and **SASS** (syntactically awesome stylesheets) most common
 - variables, operators, loops, nesting, mixins (functions), explicit inheritance, modular files (with partials)

SASS

```
@for $i from 1 through 8 {  
  $width: percentage(1 / $i)  
  
  .col-#{ $i } {  
    width: $width;  
  }  
}
```



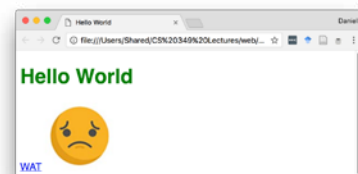
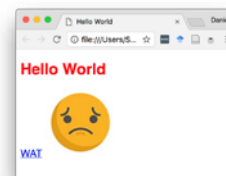
CSS

```
.col-1 {width: 100%;}  
.col-2 {width: 50%;}  
.col-3 {width: 33.333%;}  
.col-4 {width: 25%;}  
.col-5 {width: 20%;}  
.col-6 {width: 16.666%;}  
.col-7 {width: 14.285%;}  
.col-8 {width: 12.5%;}
```

Web UI 15

CSS3 Media Queries for Responsive Layout (~2012)

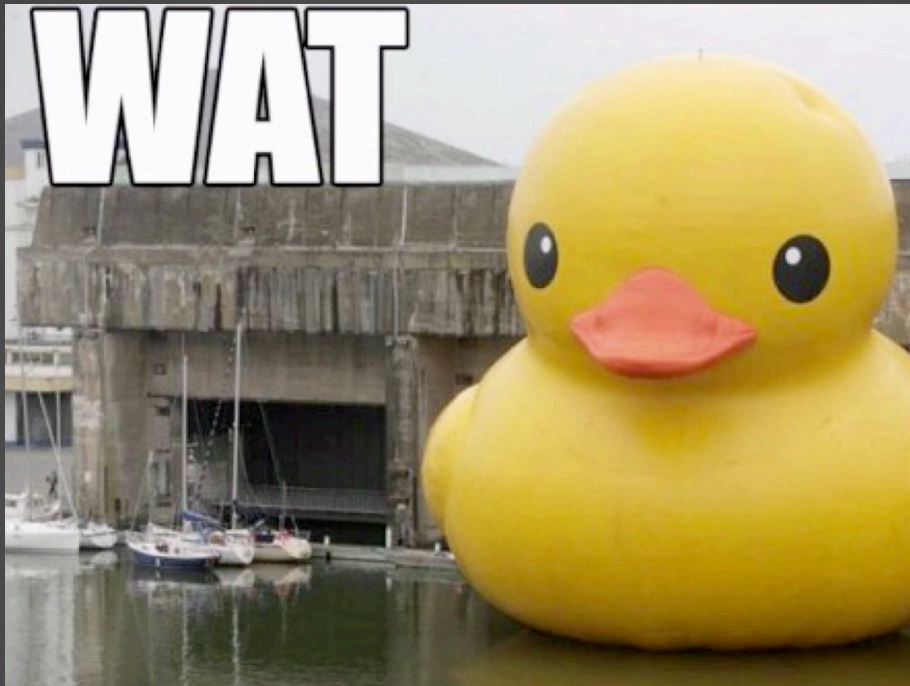
```
/* For mobile phones: */  
h1 {  
  color: red;  
  font-size: 1.5rem;  
}  
  
/* For tablets: */  
@media only screen and (min-width: 600px) {  
  h1 {  
    color: green;  
    font-size: 2rem;  
  }  
}  
  
/* For desktop: */  
@media only screen and (min-width: 768px) {  
  h1 {  
    color: blue;  
    font-size: 3rem;  
  }  
}
```



Web UI 16

JavaScript (1995)

- partly based on Scheme
 - Java-like syntax for business reasons
 - first version written in 10 days (and shipped!)
- originally for designers to “script” web pages
 - interpreted
 - inheritance by **prototypes**, not classes
 - functional programming traits
 - dynamically typed



Wat, A lightning talk by Gary Bernhardt from CodeMash 2012

- <https://www.destroyallsoftware.com/talks/wat>

JSON (JavaScript Object Notation) (early 2000s)

- lightweight data-interchange format
 - alternative to XML
- Hierarchy of two structures:
 - A collection of name/value pairs
 - An ordered list of values

```
{
  "title": "Person",
  "type": "object",
  "properties": {
    "firstName": {
      "type": "string"
    },
    "lastName": {
      "type": "string"
    },
    "age": {
      "description": "Age in years",
      "type": "integer",
      "minimum": 0
    }
  },
  "required": ["firstName", "lastName"]
}
```

Web UI 19

AJAX (asynchronous JavaScript and XML) (1995)

- JavaScript and XMLHttpRequest object enable exchanging data between browser and server to avoid full page reloads
- Oddly, XML is rarely used, almost always JSON data format
- Lead to the single-page app paradigm

(1995 Article by Jesse James Garret)



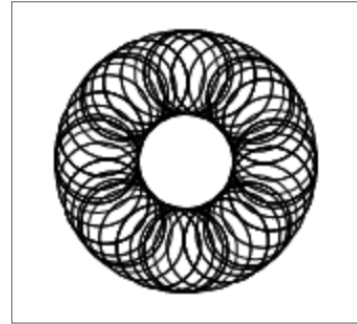
Web UI 20

HTML Canvas (~2005)

```
<canvas id="myCanvas" width="200" height="200"/>
<script>
var c = document.getElementById("myCanvas");
var ctx = c.getContext("2d");

ctx.translate(100, 100);

for (a = 0; a < 360; a += 3) {
  ctx.rotate(a * (Math.PI / 180) );
  ctx.beginPath();
  ctx.arc(30, 30, 20, 0, 2 * Math.PI);
  ctx.stroke();
}
```



Jquery (2006)

- “Vanilla” JavaScript is verbose, error-prone, browser-specific
- jQuery is a JavaScript library that simplifies:
 - Traversing the DOM
 - Animating elements in the DOM
 - Handling events on the DOM, **example:**

JavaScript

```
var el = document.getElementById("myButton")
el.addEventListener("click", function() {
  alert("Hello");
}, false);
```



```
$("#myButton").click(function() {
  alert('Hello');
});
```

JavaScript Transcompilers (~2009)


- A transcompiler is **source-to-source** compiler

JavaScript

```
function greet(name) {  
    return "Hi " + name;  
}  
greet("Bob");
```

TypeScript

```
function greet(name: string): string {  
    return `Hi ${name}`  
}  
greet("Bob");
```

 Kotlin

```
fun greet(name: String): String {  
    return "Hi $name."  
}  
greet("Bob")
```

Web UI 23

Web MV* Frameworks (~2010)



```
import { Component, Input } from '@angular/core';  
  
@Component({  
    selector: 'app-hello-world',  
    templateUrl: './hello-world.component.html',  
    styleUrls: ['./hello-world.component.css']  
})  
export class HelloWorldComponent {  
    @Input() name = 'test';  
}
```

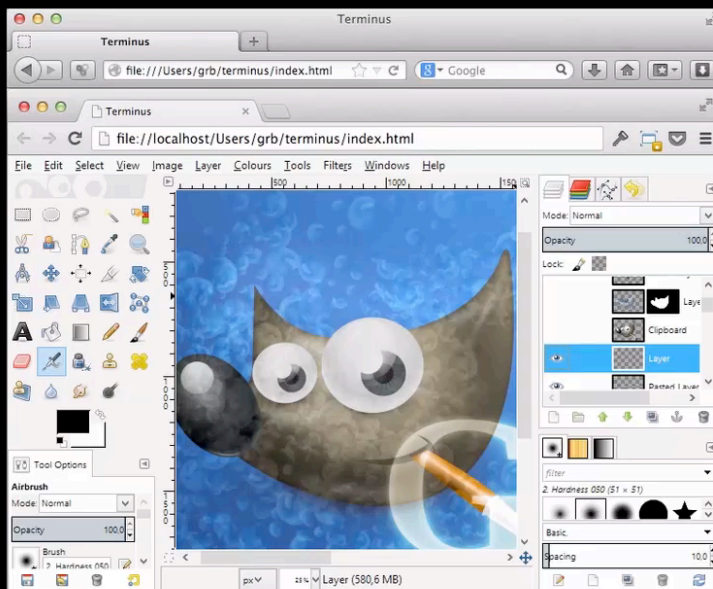
```
<div>Hello, </div>
```



```
import React from 'react';  
import './HelloWorld.css'; //import component style  
  
class HelloWorld extends React.Component {  
    render () {  
        return (  
            <div>Hello, { this.props.name }</div>  
        )  
    }  
}  
  
export default HelloWorld;
```

ECMAScript

- JavaScript is based on ECMAScript, a ECMA-262 standard
- 4th edition – ECMAScript (2007)
 - abandoned
- 5th edition - ECMAScript 3.1 (2009)
 - current JavaScript in most browsers
- 6th edition - ECMAScript 2015 (incomplete browser support)
 - huge update to language
- 8th edition - ECMAScript 2017
 - await/async using generators and promises
- **Babel** (2014) is a transpiler to convert code written using new JavaScript standards down to older JavaScript standards

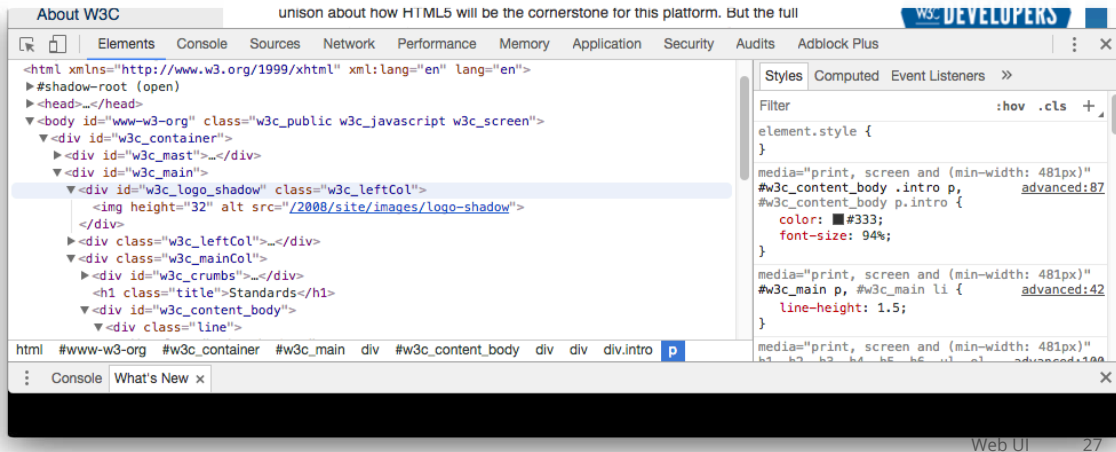


asm.js (*The Birth & Death of JavaScript*, Gary Bernhardt)

- <https://www.destroyallsoftware.com/talks/the-birth-and-death-of-javascript>

Other Tools

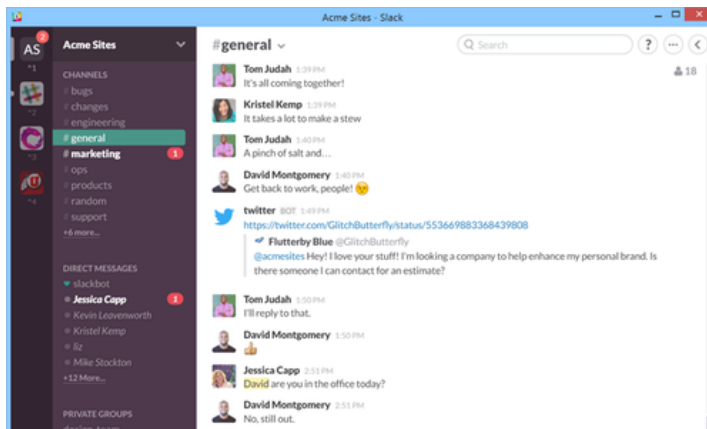
- package manager
 - Bower, npm
- automation (“building”, “task running”)
 - Gulp, Gradle
- Debugging and inspecting



Electron (2013)

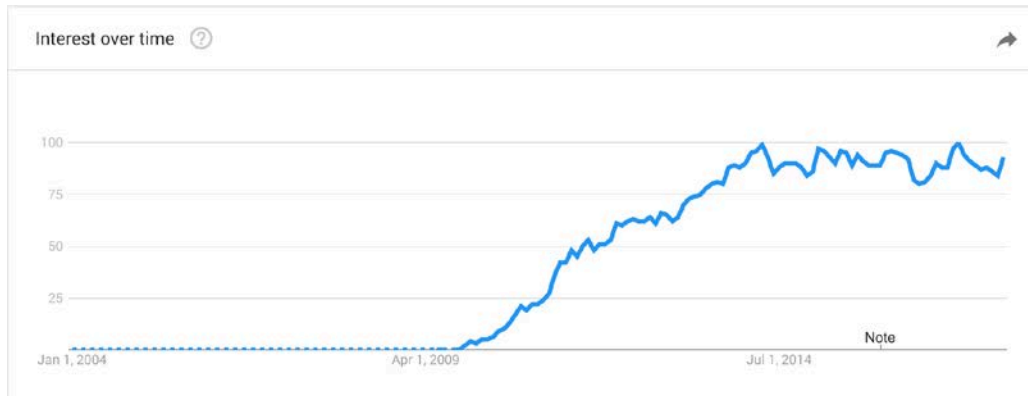


- Cross-platform desktop apps built with HTML, CSS, JavaScript
- Electron provides:
 - a minimal web browser to run your app interface
 - JavaScript libraries to handle cross platform file access
 - a package to bundle everything together to install and run
- Slack is an Electron app



Node.js (2009)

- A JavaScript runtime built on Chrome's V8 JavaScript engine
 - JavaScript on the **back end**



Web UI 29

There's much, much more ...

- Semantic Web
 - Resource Desc. Framework (RDF), Web Ontology Language (OWL)
- RESTful web services
- Managing State
 - cookies, query strings, HTTP state, etc.
- Cloud Computing
 - Amazon Web Services, Docker
- Server Frameworks and Common Platforms
 - Ruby on Rails, Django, WordPress,
- Other Key Formats and Technologies
 - XML, JSON, PNG, SVG, web fonts, markdown, URI, SSL
- Search Engine Optimization (SEO), Privacy and Security, Internationalization, Testing, Staging and Deployment, ...

Web UI 30