FRAMEWORKS ARE DEAD, LONG LIVE THE COMPILER

Mike Hartington | @mhartington

OR

OUR BEST PRACTICES ARE FAILING AND OUR USERS ARE PAYING FOR IT.

I LOVE JAVASCRIPT

I kind of have to say this...

I LOVE THE WEB

Obviously

THE WEB AS A PLATFORM IS BLOATED

KIND OF OUR FAULT 😬

MBs of JS to create...A DIV?

Devs throw too much at the browser

HOW DID WE GET HERE?

The history of the web platform

How have framework evolved

What does the future bring us?

COMPILERS MAY OR MAY NOT SAVE THE WEB

CHAPTER 1

The rise of the frontend

IN THE BEGINNING...

HTML ('93)

JavaScript ('95)

CSS ('96)

Same building blocks we have today

A HOSTILE BROWSE

Internet Explorer was a thing

Difference in implementations

Varying degrees of support for features

INTERNET EXPLORER

THE WEB WAS A BAD BET

Java applets

Flash & Actionscript

Silverlight

2006-2010: THE WEB-RENAISSANCE

Ajax and Google Maps

Jquery became a thing

ES5 shipped (First since '99)

Backbone, AngularJS, Knockout, etc

THE WEB STARTED TO GET BETTER

Libs eased over browser differences

Frameworks built better apps

JavaScript at the center

CHAPTER 2

JavaScript meets ES6 and mobile

CAN'T STOP, WON'T STOP

Mobile-first became common place

Frameworks kept pushing forward

ES2015 Shipped

More JavaScript for all the things

FRAMEWORKS KEPT BUILDING

Custom syntax for "components"

Templating languages

Alternatives to native DOM

CSS-in-JS

WOAH...JAVASCRIPT IS EVERYWHERE

THE WEB GOT BIG

MEANWHILE, IN MOBILE

Devices haven't changed much

Less memory/resources than desktop

Slower parse/compile process for JS

TOO MUCH RUNTIME JS

Runtime templating

Runtime styling

Runtime diffing

Scientists were so preoccupied with whether or not they could, they didn’t stop to think if they should.

Dr. Ian Malcolm

Should JavaScript do everything?

Is this a good experience?

Will anyone think about mobile users?

CHAPTER 3

Enter the Compiler

THE STATUS QUO

Frameworks and JS are everywhere

Average JS bundle keeps going up

Up 47% from 2019

httparchive.org/reports/state-of-javascript

All this runtime logic is hurting actual users

ENTER A COMPILER

COMPILERS ARE NOT NEW

Most low level languages have them

Move runtime framework APIs to build time

Reduce runtime JavaScript sent down the wire

LET'S TAKE AN EXAMPLE

```
class Component {
  name = "world"
  shouldShow = true
}
```

What should the output be?

ANGULAR

Pre 9.0

```
@Component({
  selector: 'app-root',
  template: `
    <div> Hello, {{ title }}</div>
    <div *ngIf="shouldShow"> Should I show? </div>
  `

})
export class AppComponent {
  title = 'world';
  shouldShow = true
}
```

```
}
function View_AppComponent_0(_l) {
  return _angular_core__WEBPACK_IMPORTED_MODULE_0__['evid']( 0,
    [
      (_l()(), _angular_core__WEBPACK_IMPORTED_MODULE_0__['eeld']( 0, 0, null, null, 1, 'div', [], null, null, null, null, null)),
      (_l()(), _angular_core__WEBPACK_IMPORTED_MODULE_0__['eted'](1, null, [ ' Hello, ', '\n', ])),
      (_l()(), _angular_core__WEBPACK_IMPORTED_MODULE_0__['eand']( 16777216, null, null, 1, null, View_AppComponent_1)),
      _angular_core__WEBPACK_IMPORTED_MODULE_0__['edid']( 3, 16384, null, 0,
      _angular_common__WEBPACK_IMPORTED_MODULE_1__['NgIf'],

        [
          _angular_core__WEBPACK_IMPORTED_MODULE_0__[ 'ViewContainerRef' ],
          _angular_core__WEBPACK_IMPORTED_MODULE_0__['TemplateRef'],
        ],
        { ngIf: [0, 'ngIf'] }, null),
    ],
    function (_ck, _v) {
      var _co = _v.component;
      var currVal_1 = _co.shouldShow;

      _ck(_v, 3, 0, currVal_1);
    },
    function (_ck, _v) {
      var _co = _v.component;
      var currVal_0 = _co.title;
      _ck(_v, 1, 0, currVal_0);
    }
  );
}
```

Difficult to understand

Difficult to debug

Difficult to TREESHAKE

Treeshaking: The removal of code not used by a program at build time.

ANGULAR 8.X

Not easy to treeshake

Tightly coupled-rendering

Passes the cost down to users

ANGULAR

Post 9.0

New compiler: Ivy

Decouples internals

More easily treeshaken

```
@Component({
  selector: 'app-root',
  template: `
    <div> Hello, {{ title }} </div>
    <div *ngIf="shouldShow"> Should I show? </div>
  `

})
export class AppComponent {
  title = 'world';
  shouldShow = true
}
```

```
    _angular_core__WEBPACK_IMPORTED_MODULE_0__['eeelementEnd']();
  }
}
class AppComponent {
  constructor() {
    this.title = 'world';
    this.shouldShow = true;
  }
}
function AppComponent_Template(rf, ctx) {
  if (rf & 1) {
    _angular_core__WEBPACK_IMPORTED_MODULE_0__['eeelementStart']( 0, 'div');
    _angular_core__WEBPACK_IMPORTED_MODULE_0__['eetext'](1);
    _angular_core__WEBPACK_IMPORTED_MODULE_0__['eeelementEnd']();
    _angular_core__WEBPACK_IMPORTED_MODULE_0__['eetemplate']( 2, AppComponent_div_2_Template, 2, 0, 'div', 0);
  }
  if (rf & 2) {
    _angular_core__WEBPACK_IMPORTED_MODULE_0__['eeadvance'](1);
    _angular_core__WEBPACK_IMPORTED_MODULE_0__['eetextInterpolate1']( ' Hello, ', ctx.title, '\n');
    _angular_core__WEBPACK_IMPORTED_MODULE_0__['eeadvance'](1);
    _angular_core__WEBPACK_IMPORTED_MODULE_0__['eeproperty']( 'ngIf', ctx.shouldShow);
  }
},
```

WITH IVY

✓ Easier to understand

✓ Compiler simplifies renderer

✓ Easier to treeshake

An Idea so great, others had it too

GLIMMERJS: EMBER

Underlying API for ember components

Templates ship as bytecode

Linear sequence of instructions

STENCILJS: IONIC

Compiler for web components

Treeshake-able feature set

SVELTEJS

"Write less code"

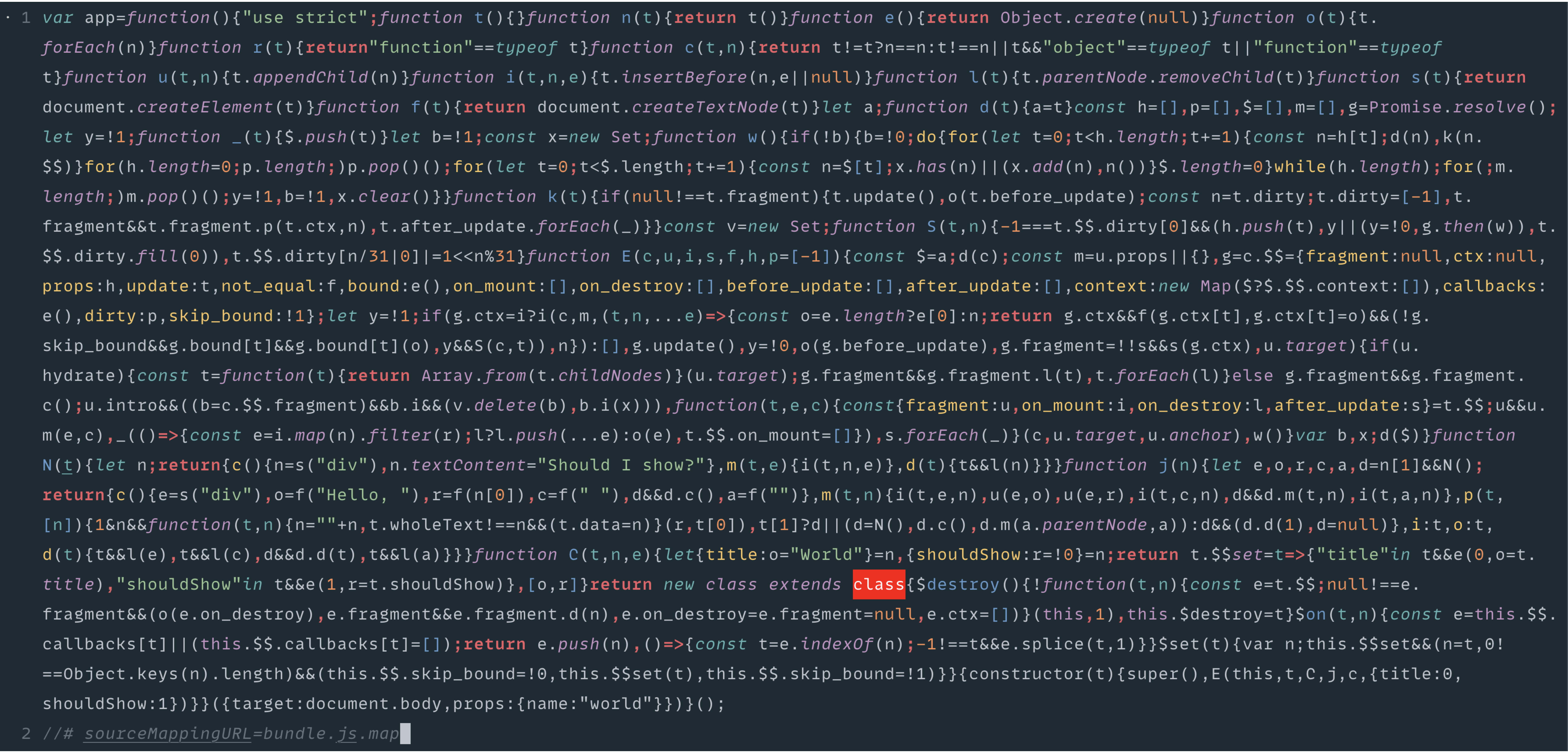
Every aspect of Svelte is treeshake-able

Super compelling

```
<script>
  export let title = 'World';
  export let shouldShow = true
</script>

<div>

  Hello, {title}
</div>
{#if shouldShow}
  <div>Should I show?</div>
{/if}
```



```
var app=function(){"use strict";function t(){}function n(t){return t()}function e(){return Object.create(null)}function o(t){t.
forEach(n)}function r(t){return"function"==typeof t}function c(t,n){return t!=t?n==n:t!==n||t&&"object"==typeof t||"function"==typeof
t}function u(t,n){t.appendChild(n)}function i(t,n,e){t.insertBefore(n,e||null)}function l(t){t.parentNode.removeChild(t)}function s(t){return
document.createElement(t)}function f(t){return document.createTextNode(t)}let a;function d(t){a=t}const h=[],p=[],$=[],m=[],g=Promise.resolve();
let y=!1;function _(t){$.push(t)}let b=!1;const x=new Set;function w(){if(!b){b=!0;do{for(let t=0;t<h.length;t+=1){const n=h[t];d(n),k(n.
$$)}for(h.length=0;p.length;)p.pop()();for(let t=0;t<$.length;t+=1){const n=$[t];x.has(n)||(x.add(n),n())}$.length=0}while(h.length);for(;m.
length;)m.pop()();y=!1,b=!1,x.clear()}}function k(t){if(null!==t.fragment){t.update(),o(t.before_update);const n=t.dirty;t.dirty=[-1],t.
fragment&&t.fragment.p(t.ctx,n),t.after_update.forEach(_)}}const v=new Set;function S(t,n){-1===t.$$.dirty[0]&&(h.push(t),y||(y=!0,g.then(w)),t.
$$.dirty.fill(0)),t.$$.dirty[n/31|0]|=1<<n%31}function E(c,u,i,s,f,h,p=[-1]){const $=a;d(c);const m=u.props||{},g=c.$$={fragment:null,ctx:null,
props:h,update:t,not_equal:f,bound:e(),on_mount:[],on_destroy:[],before_update:[],after_update:[],context:new Map($?$.$$.context:[]),callbacks:
e(),dirty:p,skip_bound:!1};let y=!1;if(g.ctx=i?i(c,m,(t,n,...e)=>{const o=e.length?e[0]:n;return g.ctx&&f(g.ctx[t],g.ctx[t]=o)&&(!g.
skip_bound&&g.bound[t]&&g.bound[t](o),y&&S(c,t)),n}):[],g.update(),y=!0,o(g.before_update),g.fragment=!!s&&s(g.ctx),u.target){if(u.
hydrate){const t=function(t){return Array.from(t.childNodes)}(u.target);g.fragment&&g.fragment.l(t),t.forEach(l)}else g.fragment&&g.fragment.
c();u.intro&&((b=c.$$.fragment)&&b.i&&(v.delete(b),b.i(x))),function(t,e,c){const{fragment:u,on_mount:i,on_destroy:l,after_update:s}=t.$$;u&&u.
m(e,c),_(()=>{const e=i.map(n).filter(r);l?l.push(...e):o(e),t.$$.on_mount=[]}),s.forEach(_)}(c,u.target,u.anchor),w()}var b,x;d($)}function
N(t){let n;return{c(){n=s("div"),n.textContent="Should I show?"},m(t,e){i(t,n,e)},d(t){t&&l(n)}}}function j(n){let e,o,r,c,a,d=n[1]&&N();
return{c(){e=s("div"),o=f("Hello, "),r=f(n[0]),c=f(" "),d&&d.c(),a=f("")},m(t,n){i(t,e,n),u(e,o),u(e,r),i(t,c,n),d&&d.m(t,n),i(t,a,n)},p(t,
[n]){1&n&&function(t,n){n=""+n,t.wholeText!==n&&(t.data=n)}(r,t[0]),t[1]?d||(d=N(),d.c(),d.m(a.parentNode,a)):d&&(d.d(1),d=null)},i:t,o:t,
d(t){t&&l(e),t&&l(c),d&&d.d(t),t&&l(a)}}}function C(t,n,e){let{title:o="World"}=n,{shouldShow:r=!0}=n;return t.$$set=t=>{"title"in t&&e(0,o=t.
title),"shouldShow"in t&&e(1,r=t.shouldShow)},[o,r]}return new class extends class{$destroy(){!function(t,n){const e=t.$$;null!==e.
fragment&&(o(e.on_destroy),e.fragment&&e.fragment.d(n),e.on_destroy=e.fragment=null,e.ctx=[])}(this,1),this.$destroy=t}$on(t,n){const e=this.$$.
callbacks[t]||(this.$$.callbacks[t]=[]);return e.push(n),()=>{const t=e.indexOf(n);-1!==t&&e.splice(t,1)}}$set(t){var n;this.$$set&&(n=t,0!
==Object.keys(n).length)&&(this.$$.skip_bound=!0,this.$$set(t),this.$$.skip_bound=!1)}}{constructor(t){super(),E(this,t,C,j,c,{title:0,
shouldShow:1})}}({target:document.body,props:{name:"world"}})}();
//# sourceMappingURL=bundle.js.map
```

245 LOC

3KB

No instructions, just JS

The most extreme case

intentionally left blank

EPLIOUGE

There and back again

COMPILERS ARE THE FUTURE?

Offer smarter apps

Reduce overall JS sent to user

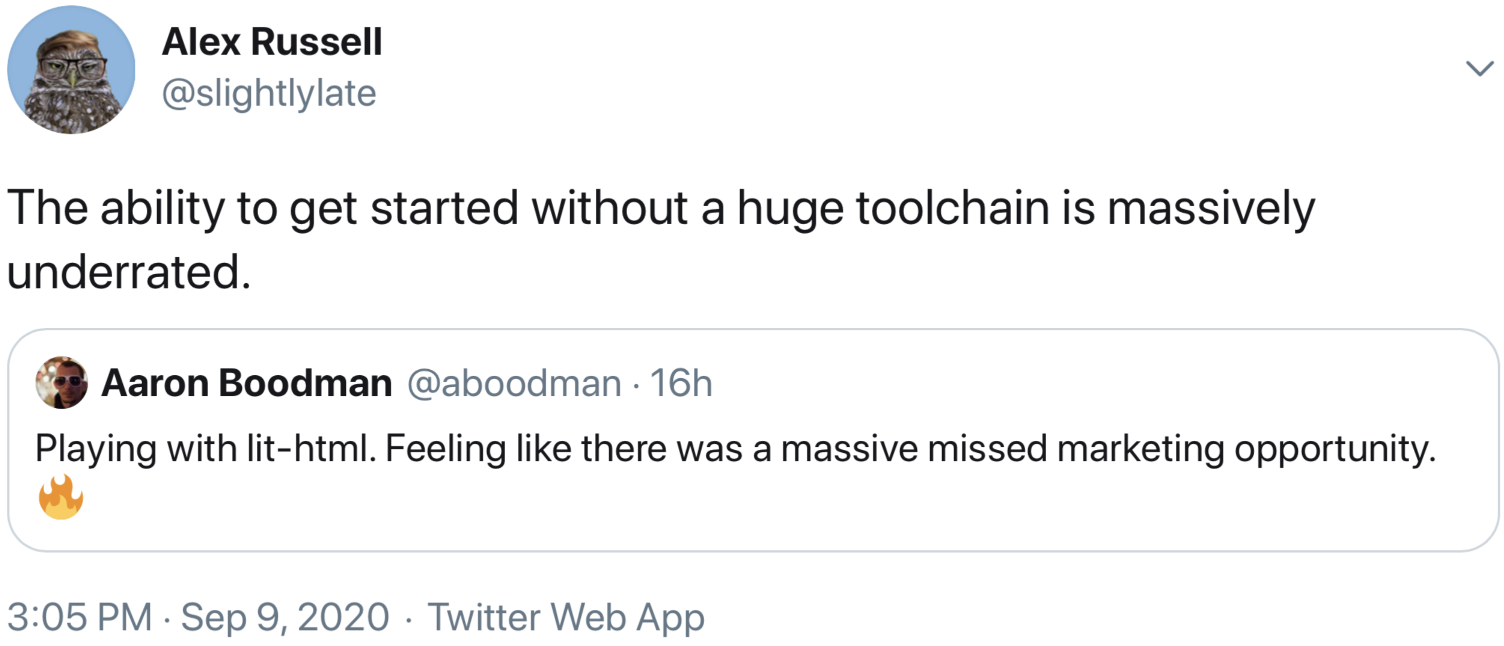
Simplifies framework architecture

BUT…

There are trade-offs

Not common place, yet

Offsets complexity to build tools



Alex Russell
@slightlylate

The ability to get started without a huge toolchain is massively underrated.

> **Aaron Boodman** @aboodman · 16h
>
> Playing with lit-html. Feeling like there was a massive missed marketing opportunity. 🔥

3:05 PM · Sep 9, 2020 · Twitter Web App

He's not wrong

But smarter tools can help build better apps

With time, there is hope

Computer, build me an app

https://youtu.be/qqt6YxAZoOc

THANK YOU

Mike Hartington | @mhartington

`</html>`