

VOXXED DAYS

SINGAPORE

BIG THANKS
TO OUR SPONSORS

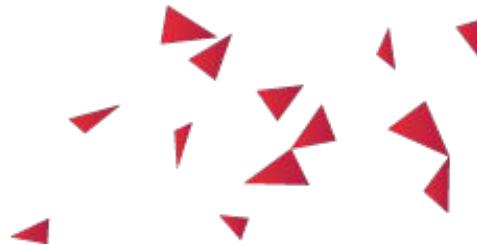
FOLLOW US ON



#voxxeddayssingapore

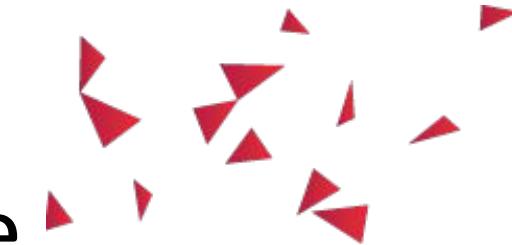


zenika
<led by passion>



Web Western

The Good, The Bad and The Web Components



DesperaDOM

<- Warning: this presentation is not bad jokes free...



Adrien Nortain
CTO @ Zenika





Welcome to the Wild West!



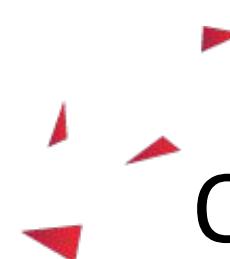
The Good: Web development has never been this **interesting with flourishing standards**

The Bad: People tend to learn frameworks and not the platform - **is the browser not enough?**

The Web Components: Since 2016, they propose a **way to unleash the power of the DOM**

- API powerful enough to compete with the shiny frameworks?
- Any gains from dropping favorite frameworks?
- Any trade-off? AND WHY **WEB COMPONENTS === UGLY??**





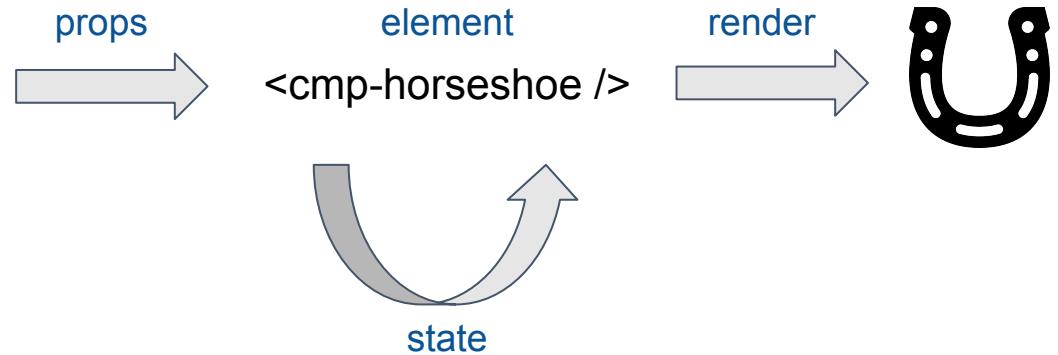
Quick Draws Planning

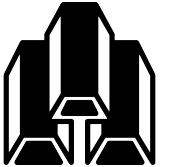
1. History VS Future
2. Pros VS Cons
3. Good VS Bad VS WC



My Name Is Component

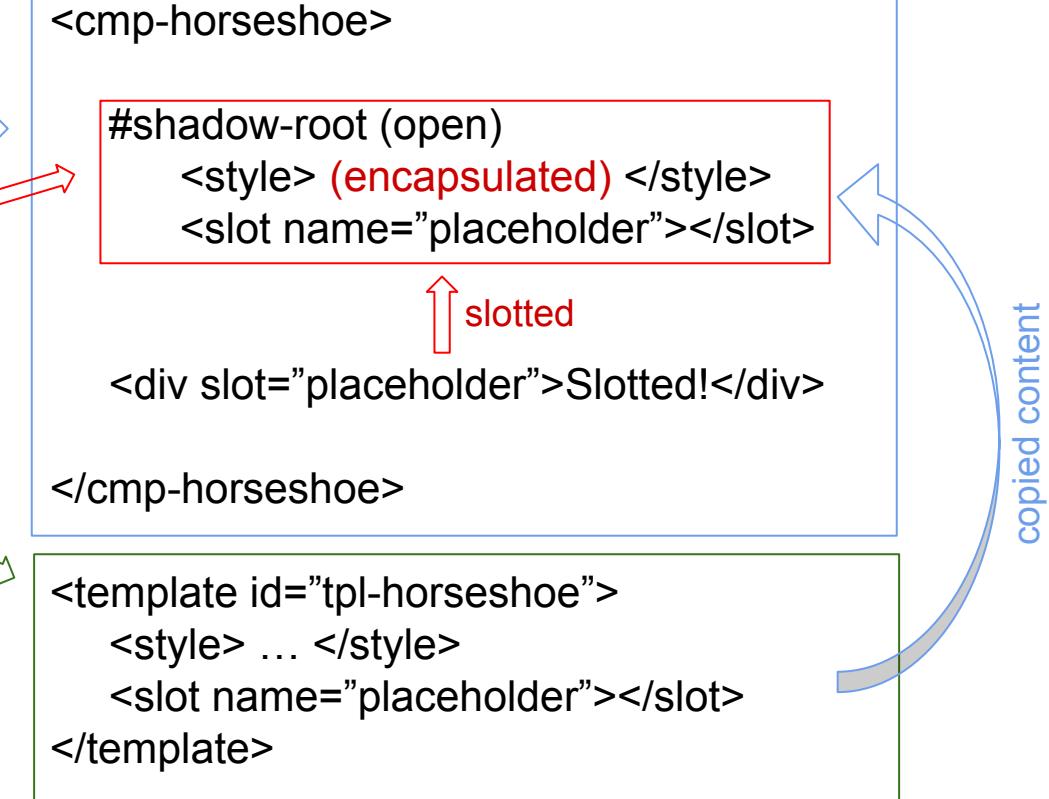
- **Reusable / shareable**
 - Paired with composition
 - Application state agnostic
- **Configurable**
 - Initialization parameters
 - Customizable rendering
- **Extensible**
 - Using lifecycle hooks
 - Using mixins





For a Few Features More

- **Custom Elements**: Component tags + definition
- **Shadow DOM**: Encapsulation + slots
- **HTML Template**: Reusability + lazy-loading
- ~~HTML Imports~~: Code splitting => how then???





Pale VO (2014 - 2019)

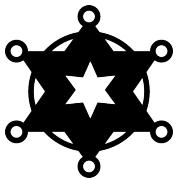
- Adoption: **Only natively supported by Chrome** (Blink engine) since M33 (2014)
- Future: **Removed with M74** - can only be run with polyfills now
- Legacy: A **nice spec** / one additional lifecycle hook (createdCallback)

```
var MyCmpProto = Object.create(HTMLElement.prototype);

MyCmpProto.createdCallback = function() {
    this.innerHTML = "<span>Some component content</span>";
};

var MyCmp = document.registerElement('my-cmp', {prototype: MyCmpProto});
```



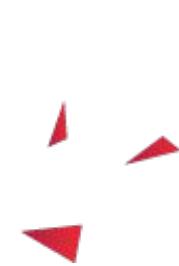


V1 Unchained (2016 - ?)

- Adoption: **Fully adopted** by Chrome, Opera, Safari and Firefox
- Future: **Soon supported by Edge** (Chromium-based version)
- Legacy: A pretty neat **framework kind of feeling** (ES6 modules...)

```
class MyCmp extends HTMLElement {  
  constructor() {  
    super();  
    this.attachShadow({mode: 'open'}).innerHTML = "<span>Some component  
content</span>";  
  }  
};  
  
customElements.define('my-cmp', MyCmp);
```

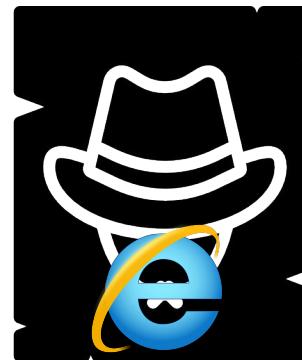




Browse 'Em High



- Problem: **Older browsers** support
- Solution: **Polyfills**
 - webcomponentsjs
 - WebReflection
- Yet: **Not optimized** and sometimes buggy (especially on IE)



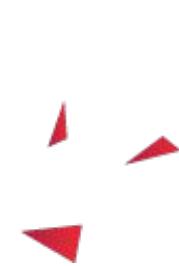


The Wild Tools



- **Babel** is optional
- **Webpack** is still a must-have, sadly (even if it rocks):
 - Code splitting
 - HTML / CSS imports in JS
 - Helps with templating, PWA, chunks bundling, dynamic imports...
- **lit-html**, StencilJS and other components libraries or frameworks can help:
 - Kickstart projects
 - Organize the code
 - Support legacy browsers





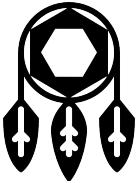
Billy The Libs



- **Enterprise grade libs:** SGX, DBS and other companies have their own
 - <https://www.predix-ui.com> from GE Digital
 - <https://github.com/elix/elix>
 - <https://github.com/vaadin/vaadin>
 - <https://github.com/SAP/ui5-webcomponents>
- **Visual libs:** for a nicer look and feel
 - <https://github.com/material-components/material-components-web-components> w/ Polymer
 - <https://github.com/wiredjs/wired-elements>
- **Your own lib:** always fits your needs



True Git



Personal demo project: <https://wonton-plate.herokuapp.com>

- **Pure** web development
- **Lightweight**, accessible and performant
- **Free** software design decisions

The screenshot shows a web application for managing a movie database. At the top, there are navigation links: HOME (highlighted in red), HELLOJOHN, FORM, and MOVIES. A user session is indicated with 'Hi! Log in to rate your favorite movies with wontons!' and a login link. The main content area displays a grid of movie cards. Each card includes the movie title, release date, directors, producers, cast, and a rating section with five yellow stars and a 'Rate' button. The movies listed are:

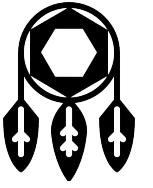
- A Chinese Ghost Story (18/07/1987)
- In The Line Of Duty 4 (01/01/1970)
- Once Upon A Time in China (15/08/1991)
- The Blade (21/12/1993)
- The Chinese Feast (28/09/1993)
- The Killer (06/07/1989)

At the bottom left, there's a language selection dropdown set to English, and at the bottom right, a copyright notice: © LamSonFei 2019.

<https://github.com/LamSonFei/wonton-plate>



True Git



```
import './styles.css';
import template from './template.html';

import log from 'services/log';
import { WontonMixin } from 'components/mixins/wonton';
import { I18nMixin } from 'components/mixins/i18n';
import { mix } from 'utils/mixins';

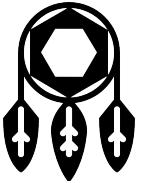
/**
 * Hello World demo component.
 */
export class HelloWorld extends mix(HTMLElement).with(WontonMixin, I18nMixin) {
    // Wonton config
    static componentName() { return 'hello-world' }
    template() { return template }
    references() { return { 'helloWorldButton': '.cmp-hello-world_button' } }
    listeners() {
        return {
            'this': { 'click': () => log.info('HelloWorld got clicked as a component!') },
            'helloWorldButton': { 'click': () => this.sayHello() }
        }
    }
    propertiesAttributes() { return ['name'] }

    // I18n config
    i18nFilePath() { return 'components/hello-world/i18n' }
    localeChangedCallback() { this.refreshInnerTexts() }

    // Custom properties
    get helloWorldMessage() { return this.i18n('cmp.hello-world.hello', {name: this.name || this.i18n('cmp.hello-world.world')}) }
    // Methods
    sayHello() {
        this.dispatchEvent(new CustomEvent('hello', {
            detail: this.name
        }))
    }
}
```



True Git



Hi! Log in to rate your favorite movies with wontons!

[+ Add Movie](#)

adrien.nortain@gmail.com [Logout](#)

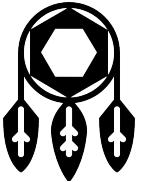
A Chinese Ghost Story 18/07/1987 Directors <ul style="list-style-type: none">• Ching Siu-Tung Producers <ul style="list-style-type: none">• Tsui Hark Cast <ul style="list-style-type: none">• Leslie Cheung• Joey Wong• Wu Ma ☆ Rate	In The Line Of Duty 4 01/01/1970 Directors Producers Cast <ul style="list-style-type: none">• Donnie Yen ☆ Rate	Once Upon A Time in China 15/08/1991 Directors <ul style="list-style-type: none">• Tsui Hark Producers <ul style="list-style-type: none">• Tsui Hark Cast ☆ Rate
The Blade 21/12/1995 	The Chinese Feast 28/01/1995 	The Killer 06/07/1989

Lang: [English](#)

@LamSonFei



The Lone Render



SGX [Securities](#) [Fixed Income](#) [Derivatives](#) [Indices](#) [Research & Education](#) [Data & Connect](#)

Home > Securities

■ Securities Prices

< [Favorites](#) [All](#) [Stocks](#) [REITs](#) [Business Trusts](#) [ETFs](#) [Structured Warrants](#) [Daily Leverage Certificates](#) >

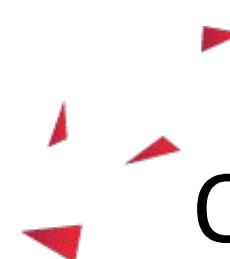
Sort by [Top Value](#) [Top Volume](#) [Advancers](#) [% Advancers](#) [Decliners](#) [% Decliners](#)

Search by name [🔍](#)

Show more filters ▾

	Trading Name	Trading...	Last	Chng	Chng %	Vol (000s)	Val (\$M)	Bid Vol	Bid	Ask	Ask Vol
★	DBS	D05	25.060	-0.140	-0.56	3,111.2	77.94	4.2	25.060	25.070	5
★	SingTel	Z74	3.170	+0.020	+0.64	12,711.9	40.14	1,295.6	3.160	3.170	1,094.9
★	UOB	U11	24.700	+0.060	+0.24	1,530.5	37.75	4.6	24.700	24.710	1.5
★	OCBC Bank	O39	10.990	-0.020	-0.18	3,007.1	32.96	18.5	10.980	10.990	51.1
★	YZJ Shipbldg SGD	BS6	1.370	-	-	20,627.5	28.42	2,708	1.370	1.380	2,587.9
★	SGX	S68	7.620	-	-	2,783.3	21.21	8.3	7.620	7.630	98.1
★	CapitaLand	C31	3.290	+0.010	+0.31	4,586.4	15.08	175.7	3.290	3.300	708.3
★	ST Engineering	S63	3.920	+0.020	+0.51	3,157.5	12.35	7.8	3.920	3.930	340
★	CityDev	C09	8.310	-0.020	-0.24	1,446.3	12.02	1.5	8.310	8.320	62.7





Quick Draws Planning

1. ~~History~~ VS Future
2. Pros VS Cons
3. Good VS Bad VS WC



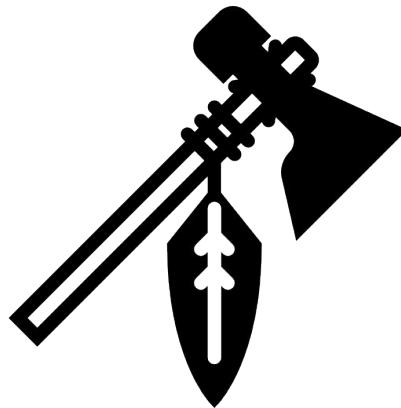
Once Upon a Time in the Web

Problem #1: How many frameworks are currently in use worldwide?

We developed fast.

We developed with what we had.

We did not develop with durability in mind.



And now we have all of this **mess to maintain or rebuild**.



Tombstones

Problem #2: How many times do we have to learn a new technology?

Javascript fatigue is real.

But fundamentals are always necessary.
How do you choose your next framework?

JQuery UI, Dojo, AngularJS, GWT, first versions of ExtJS, VueJS, Polymer, ReactJS... **RIP, legends!**



The Quick and the Virtual DOM

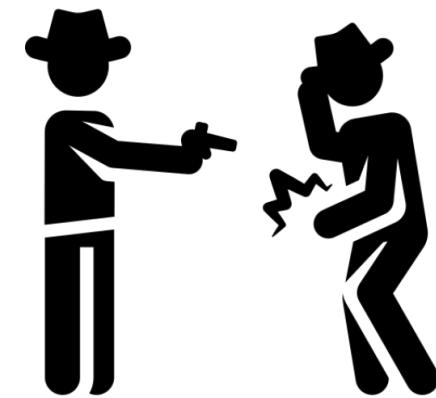
Problem #3: Why does virtually nobody care about runtime efficiency?

Frameworks are **more or less optimized**.

Performance is always the first sacrificed aspect.

Refactoring boilerplate code is difficult and often avoided.

Could we focus more on RAM consumption and data usage?



The Hateful Cons

- No Server-Side Rendering
- 3 languages: HTML / CSS / Javascript
- **Only one possible runtime**: a browser

Where it may never improve:

- No cross-platform (outside PWA)
- No server-side support
- Will always be a **browser low-level API**



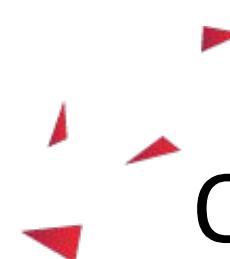
The Magnificent Pros

- **Pure** web development
- **Durable** standard
- **Fastest** implementations

Future improvements:

- Growing number of libraries, frameworks and dev tools
- Third-party web frameworks awareness
- Further versions propositions (Tim Branyen...)





Quick Draws Planning

1. ~~History~~ VS Future
2. ~~Pros~~ VS ~~Cons~~
3. Good VS Bad VS WC





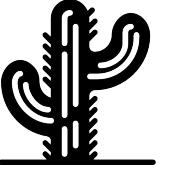
Dance With Framewolves

Some popular frameworks, such as Angular and Vue.js, offer the ability to
integrate web components



Other awesome frameworks, such as Polymer and StencilJS, help you **write pure Web Components faster**





Western Spaghetti

People seem to miss Portlets and came up with the concept of **micro-frontend architecture**

- Widgets developed w/ a framework and wrapped in web components
- A “runtime” managing the libs and defining the communication pattern (welcome back JSR-286)
- A private server for each widget to handle SSR (optional)

But where is the simplicity in this?

Will the underlying legacy ever be touched again?

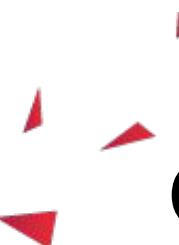




Mexican Standoff

- **VS GOOD:** Coding **productivity**
 - Community help and documentation
 - Turnkey features (routing, iterations, data binding...)
- **VS BAD:** Runtime **performance**
 - Virtual DOM VS Native DOM only
 - Tailored resources management
- **VS BAD:** Solution **standardization**
 - No SSR or new language to learn (JSX)
 - Backward compatibility





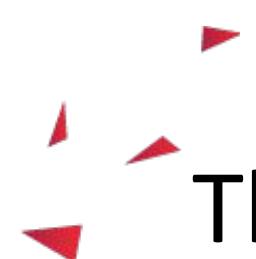
Green Guns



In terms of **Green Coding**, Web Components are:

- **Performant** 
- **Optimized** 
- **Discreet** 
- **Durable** 
- **Inclusive** 





The Good, The Bad And The Web Components

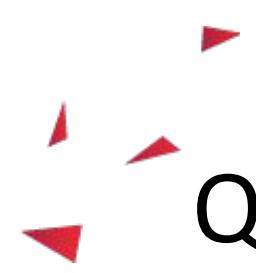
- **The Good Frameworks** and their **PRODUCTIVITY** and **ACCESSIBILITY**
- **The Bad Mess** left by **FORSAKEN** frameworks and their **GROWING LEGACY**
- **The Ugly Web Components** and their 1st good attempt at **STANDARDIZATION**

As in any Western movie, there is **no real right or wrong...**

But one may wonder: **Will there ever be a good time to stop this Western Era?**

Are Dan Abramov and Evan You going to be in **Web Dead Redemption?**

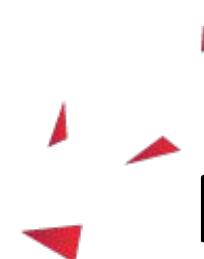




Quick Draws Planning

1. ~~History~~ VS **Future**
2. **Pros** VS ~~Cons~~
3. **Good** VS Bad VS **WC**





Riding into the sunset with Web Components



THANK YOU!



Stop at the saloon



Any questions?





Getting Started



< CODING THE WORLD />
<with> zenika LAH

wonton-plate demo



Web Western slides



Contact



< CODING THE WORLD />
<with> zenika LAH



adrien.nortain@zenika.com



<https://www.linkedin.com/in/adriennortain>



+65 9035 1852



@AdrienNortain



@LamSonFei

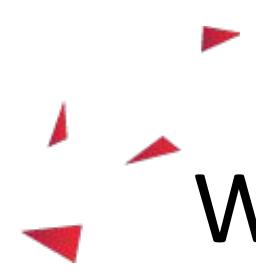


Review this talk



< CODING THE WORLD />
<with> zenika LAH



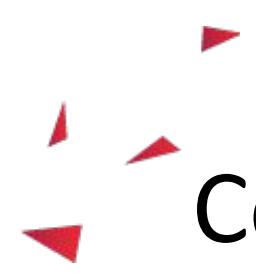


Wild Wild Web

- <https://developers.google.com/web/fundamentals/web-components/>
- <https://tbranyen.com/post/web-components-v2-wish-list>
- <https://www.carlrippon.com/are-web-components-finally-a-thing/>
- <https://dev.to/bennypowers/lets-build-web-components-part-8-mythbusters-edition-3la>
- <https://blog.bitsrc.io/9-web-component-ui-libraries-you-should-know-in-2019-9d4476c3f103>

Icons made by Smashicons, Freepik from www.flaticon.com





Coming Soon

- Crouching DOM, Hidden WebAssembly
- Caching: A Browser's Odyssey
- A Vue.js Endgame
- ...

