

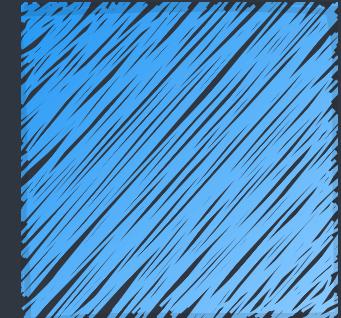
Web Components:



Natif



StencilJS



LitElement





Julien Renaux

GDE Web, Freelancer

@julienrenaux <<https://twitter.com/julienrenaux>>

<https://julienrenaux.fr/>

<<https://julienrenaux.fr/>>



Igor Laborie

Expert Web & Java

@ilaborie <<https://twitter.com/ilaborie>>

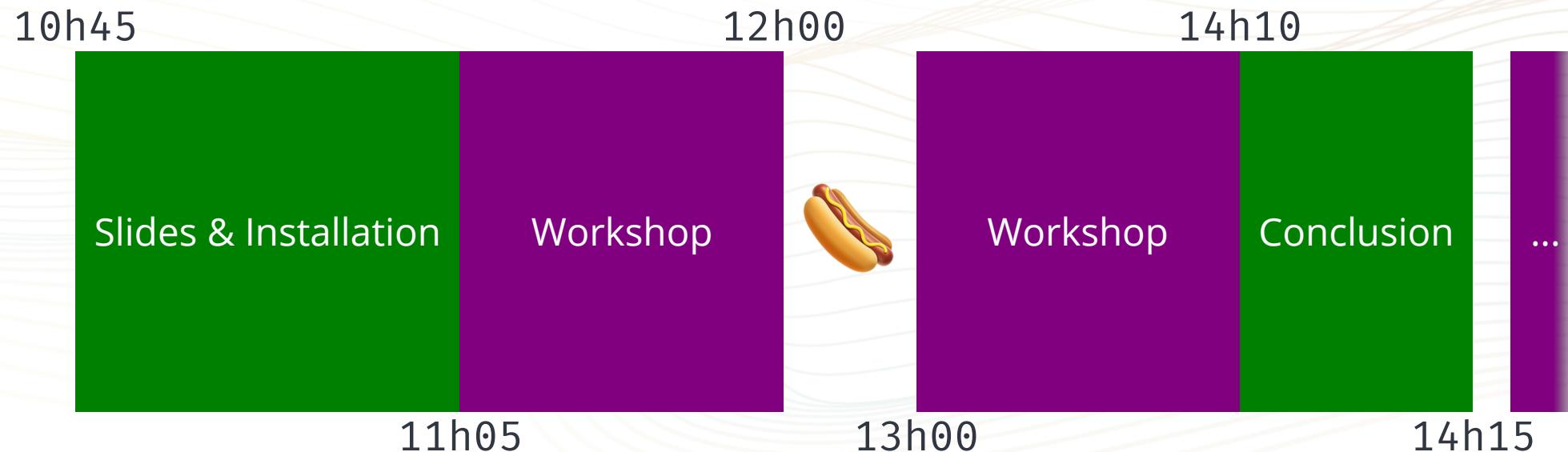
igor@monkeypatch.io

<<mailto:igor@monkeypatch.io>>





Roadmap



Instructions

<https://github.com/ilaborie/webcomponents-devoxx-19> <<https://github.com/ilaborie/webcomponents-devoxx-19>>

Puis dans les répertoires native, stencil,
lit-element, faire

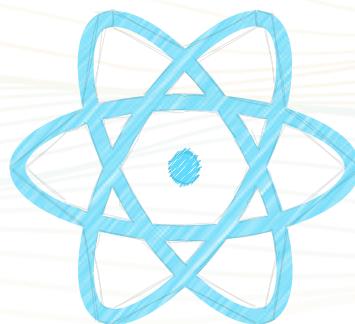


```
$ npm ci
```

<http://bit.ly/dvx-wc-slides> <<http://bit.ly/dvx-wc-slides>>

Développons une application Web en 2019

Commençons par choisir un Framework



React



Vue.js



Angular

Puis comment allons-nous écrire le style

- CSS
- Sass/Scss
- Less
- Stylus
- CSS-in-JS
- PostCSS
- NextCSS
- ...

Puis construisons notre application

- Webpack
- ParcelJs
- RollupJs
- Bazel

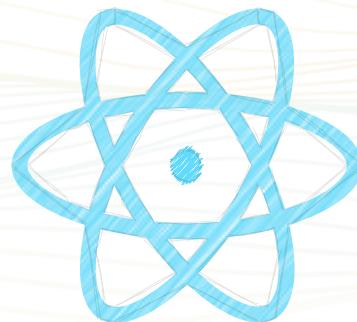


**Développer une application en JS n'est
plus simple...**

Ça va trop vite ...

A screenshot of a Twitter post. The profile picture is a small circular image of a person with short hair. The username is "I Am Devloper" and the handle is "@jamdevloper". The tweet text is: "I think I've had milk last longer than some JavaScript frameworks." Below the tweet are engagement metrics: 1,123 likes and 2:22 PM - Dec 4, 2014. To the right of the tweet is a blue Twitter icon. At the bottom of the card, there is a blue link that says "1,748 people are talking about this" followed by a right-pointing arrow.

...Interopérabilité ne vient plus gratuitement...



React

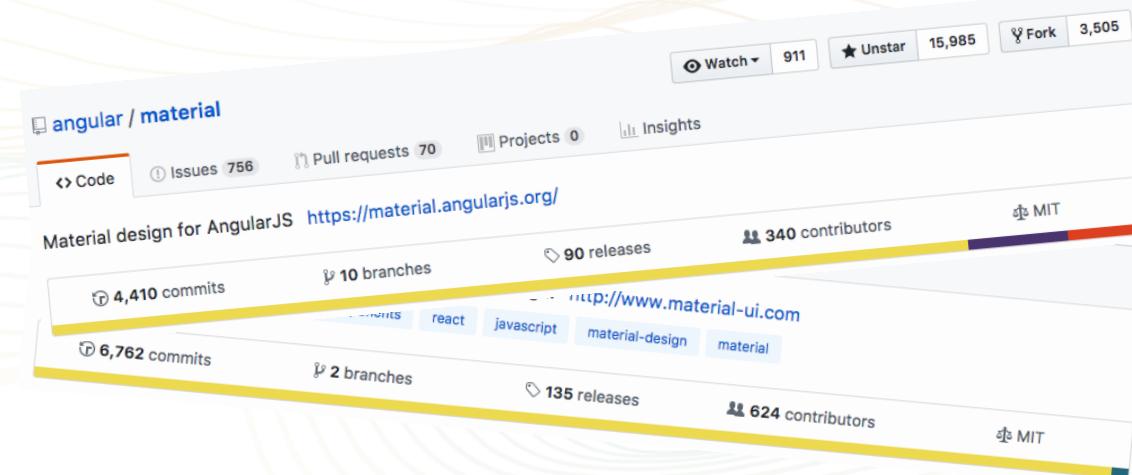


Vue.js



Angular

...et on réinvente sans arrêt la roue !





[material-components / material-components-web-components](#)

Watch 101 Star 786 Fork 137

Code Issues 75 Pull requests 28 Projects 1 Insights

Material Web Components - Material Design implemented as Web Components <https://material-components.github.io...>

290 commits 8 branches 13 releases 16 contributors Apache-2.0

Branch: master New pull request Create new file Upload files Find File Clone or download

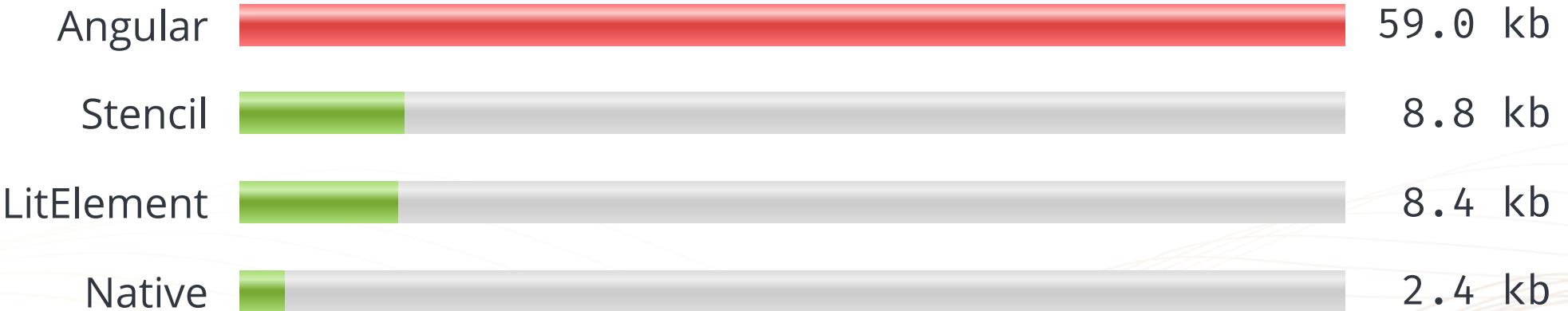
Material Web Components

► <https://github.com/material-components/material-components-web-components> <<https://github.com/material-components/material-components-web-components>>

DO IT! DO IT! DO IT!

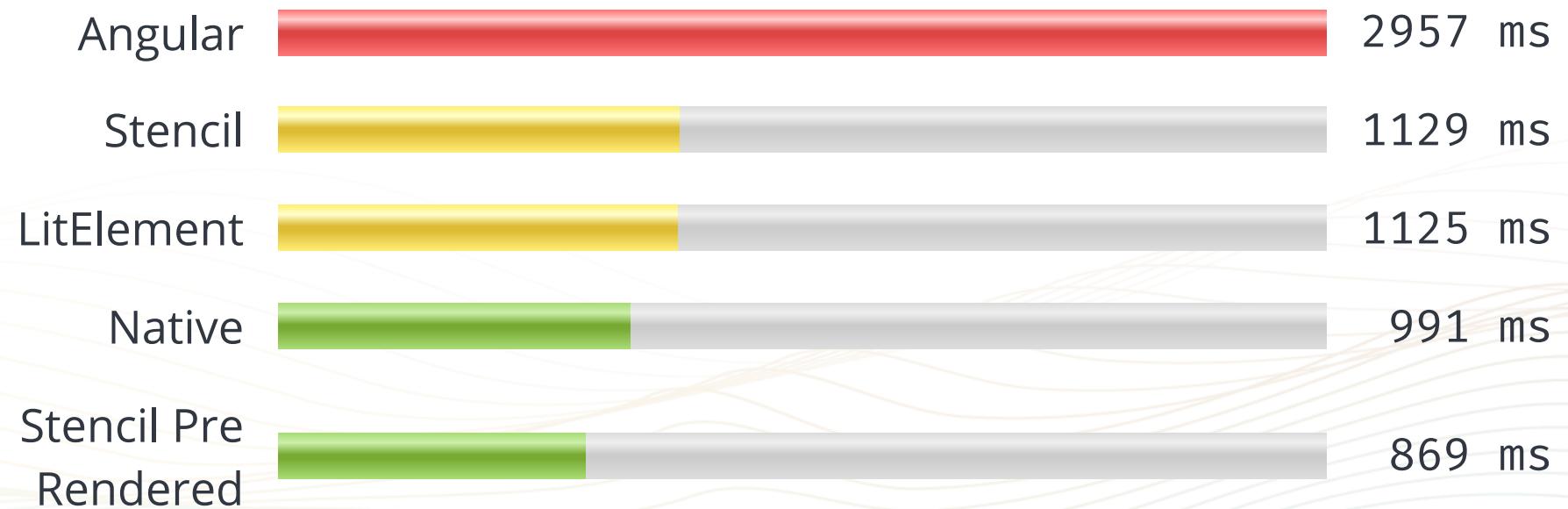


**Toute cette complexité doit
s'arrêter avec les Web
Components**



Size matters (Gzipped)

- 😢 Stencil et litElement sont 5 fois plus petits qu'Angular
- 😱 Native est 23 fois plus petit qu'Angular



Le temps ça compte (FMP 3G 📱 en ms)

😢 Angular est 3 fois plus lent

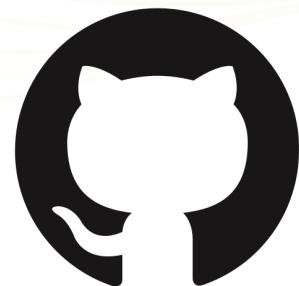
Web Components



Spécifier par le **World Wide Web Consortium** (W3C)
Débuté en **2012**

~10%

of page view in Chrome use Web Components





Custom
Elements



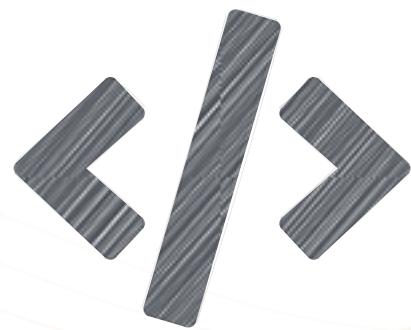
Shadow DOM



HTML templates



HTML Imports



Custom Elements

“Les *Custom Elements* sont la capacité de créer un balise HTML avec ses propres attributs et méthodes



Shadow DOM

“Le Shadow DOM fournit l'encapsulation du DOM et du CSS”



HTML templates

“Définit un bloc d'HTML réutilisable au moment de l'exécution”

```
JS
class PopUpInfo extends HTMLElement {
  constructor() {
    super();
    // ...
  }
  // ...
}

customElements.define('popup-info', PopUpInfo);
```



```
<twitter-widget class="twitter-tweet twitter-tweet-rendered" id="twitter-widget-0" style="position: static; visibility: visible; display: block; transform: rotate(0deg); max-width: 100%; width: 500px; min-width: 220px; margin-top: 10px; margin-bottom: 10px;" data-tweet-id="540481335362875392">
...
  <#shadow-root (open) == $0>
    <style type="text/css">.SandboxRoot { display: none; max-height: 10000px; }</style>
    <div data-twitter-event-id="0" class="SandboxRoot env-bp-350" style="position: relative;">...
      <style type="text/css">...</style>
    </div>
</twitter-widget>
```

Support des navigateurs



11



18



65



72



12

	IE	Edge	Firefox	Chrome	Safari
Custom Elements (V1)	😢	😢	😊	😊	😊
Shadow DOM (V1)	😢	😢	😊	😊	😊
HTML templates	😢	😊	😊	😊	😊

Supported

Partial Support

Not Supported

<https://caniuse.com> <<https://caniuse.com>>

Support avec le polyfill



11+



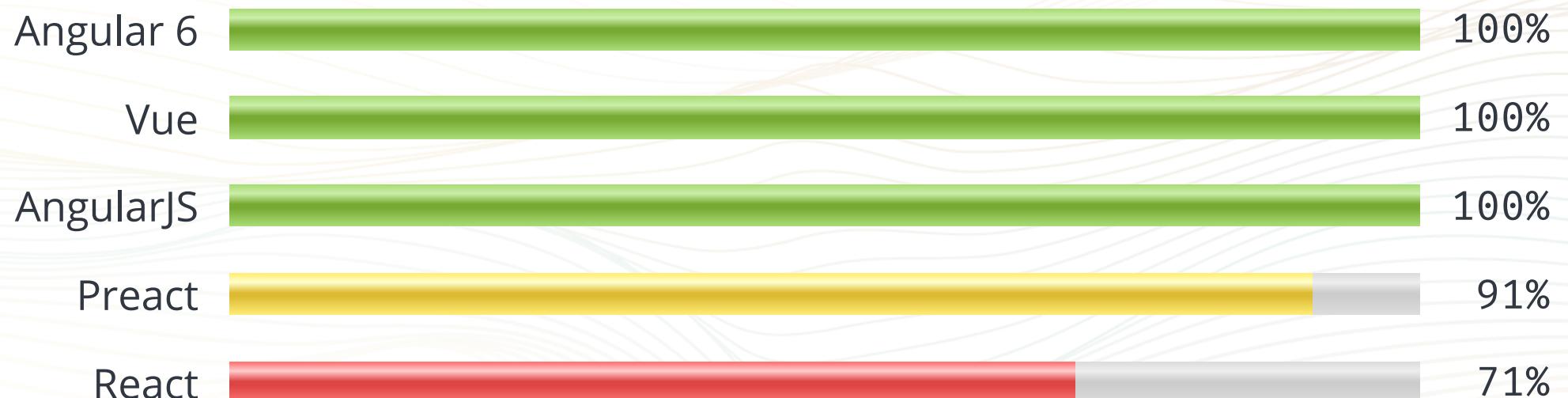
9+

	IE	Edge	Firefox	Chrome	Safari
Custom Elements (V1)	✓	✓	✓	✓	✓
Shadow DOM (V1)	✓	✓	✓	✓	✓
HTML templates	✓	✓	✓	✓	✓

<https://github.com/webcomponents/webcomponentsjs>

<<https://github.com/webcomponents/webcomponentsjs>>

Interopérabilité des Web Component



<https://custom-elements-everywhere.com/> <<https://custom-elements-everywhere.com/>>

StencilJS



<https://stenciljs.com/> <<https://stenciljs.com/>>



Projet **Open Source**, ➡ MIT License <[https://github.com/ionic-](https://github.com/ionic-team/stencil/blob/master/LICENSE)

team/stencil/blob/master/LICENSE>

Créé par l'équipe d'**Ionic** en 2017

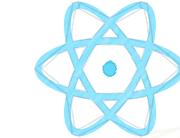
5.2k ⭐ sur github



**StencilJS n'est pas un autre
framework**

**StencilJS c'est un compilateur qui
génère des web components**

StencilJS c'est un ensemble de bons outils

				
JSX / Virtual DOM	✗	✗	✓	✓
TypeScript	✗	✓	✗	✓
Decorators	✗	✓	✗	✓
Prerendering SSR	✗	✗	✗	✓

StencilJS marche partout

Il charge les polyfills à la demande

La syntaxe de Stencil est concise

Web component

```
1 class TodoItem extends HTMLElement {
2   constructor() {
3     super();
4     this._root = this.attachShadow({ 'mode': 'open' });
5     this._checked = false;
6     this._text = '';
7   }
8
9   connectedCallback() {
10   this._root.innerHTML =
11     `<style>` +
12     `</style>` +
13     `<li class="item">` +
14       `<input type="checkbox">` +
15       `<label>${this._text}</label>` +
16       `<button class="destroy">x</button>` +
17     `</li>` +
18
19     this._site = this._root.querySelector('.item');
20     this._removeButton = this._root.querySelector('.de');
21     this._text = this._root.querySelector('label');
22     this._checkbox = this._root.querySelector('input');
23     this._removeButton.addEventListener('click', (e) => {
24       e.preventDefault();
25       this.dispatchEvent(new CustomEvent('onRemove', { index: this._index }));
26     });
27     this._checkbox.addEventListener('click', (e) => {
28       e.preventDefault();
29       this.dispatchEvent(new CustomEvent('onToggle', { index: this._index }));
30     });
31     this._render();
32   }
33
34   disconnectedCallback() []
35   static get observedAttributes() {
36     return ['text'];
37   }
38
39   attributeChangedCallback(name, oldValue, newValue) {
40     if (name === 'text') {
41       this._text = newValue;
42     }
43   }
44
45   set index(value) {
46     this._index = value;
47   }
48
49   get index() {
50     return this._index;
51   }
52
53   set checked(value) {
54     this._checked = Boolean(value);
55   }
56
57   get checked() {
58     return this.hasAttribute('checked');
59   }
60
61   _render() {
62     if (!this._item) return;
63     this._text.textContent = this._text;
64     if (!this._checked) {
65       this._item.classList.add('completed');
66       this._checkbox.setAttribute('checked', '');
67     } else {
68       this._item.classList.remove('completed');
69       this._checkbox.removeAttribute('checked');
70     }
71   }
72
73   _site.addEventListener('click', (e) => {
74     if (e.target === this._checkbox) {
75       this._checkbox.click();
76     }
77   });
78
79   _removeButton.addEventListener('click', (e) => {
80     e.stopPropagation();
81     this._remove();
82   });
83
84   _checkbox.addEventListener('change', (e) => {
85     this._checked = e.target.checked;
86     this._render();
87   });
88
89   _text.addEventListener('input', (e) => {
90     this._text = e.target.value;
91     this._render();
92   });
93
94   _root.addEventListener('click', (e) => {
95     if (e.target === this._root) {
96       this._remove();
97     }
98   });
99
100   _remove() {
101     this._root.remove();
102   }
103
104   _remove() {
105     this._root.remove();
106   }
107
108   _remove() {
109     this._root.remove();
110   }
111
112   _remove() {
113     this._root.remove();
114   }
115
116   _remove() {
117     this._root.remove();
118   }
119
120   _remove() {
121     this._root.remove();
122   }
123
124   _remove() {
125     this._root.remove();
126   }
127
128   _remove() {
129     this._root.remove();
130   }
131
132   _remove() {
133     this._root.remove();
134   }
135
136   _remove() {
137     this._root.remove();
138   }
139
140   _remove() {
141     this._root.remove();
142   }
143
144   _remove() {
145     this._root.remove();
146   }
147
148   _remove() {
149     this._root.remove();
150   }
151
152   _remove() {
153     this._root.remove();
154   }
155
156   _remove() {
157     this._root.remove();
158   }
159
160   _remove() {
161     this._root.remove();
162   }
163
164   _remove() {
165     this._root.remove();
166   }
167
168   _remove() {
169     this._root.remove();
170   }
171
172   _remove() {
173     this._root.remove();
174   }
175
176   _remove() {
177     this._root.remove();
178   }
179
180   _remove() {
181     this._root.remove();
182   }
183
184   _remove() {
185     this._root.remove();
186   }
187
188   _remove() {
189     this._root.remove();
190   }
191
192   _remove() {
193     this._root.remove();
194   }
195
196   _remove() {
197     this._root.remove();
198   }
199
200   _remove() {
201     this._root.remove();
202   }
203
204   _remove() {
205     this._root.remove();
206   }
207
208   _remove() {
209     this._root.remove();
210   }
211
212   _remove() {
213     this._root.remove();
214   }
215
216   _remove() {
217     this._root.remove();
218   }
219
220   _remove() {
221     this._root.remove();
222   }
223
224   _remove() {
225     this._root.remove();
226   }
227
228   _remove() {
229     this._root.remove();
230   }
231
232   _remove() {
233     this._root.remove();
234   }
235
236   _remove() {
237     this._root.remove();
238   }
239
240   _remove() {
241     this._root.remove();
242   }
243
244   _remove() {
245     this._root.remove();
246   }
247
248   _remove() {
249     this._root.remove();
250   }
251
252   _remove() {
253     this._root.remove();
254   }
255
256   _remove() {
257     this._root.remove();
258   }
259
260   _remove() {
261     this._root.remove();
262   }
263
264   _remove() {
265     this._root.remove();
266   }
267
268   _remove() {
269     this._root.remove();
270   }
271
272   _remove() {
273     this._root.remove();
274   }
275
276   _remove() {
277     this._root.remove();
278   }
279
280   _remove() {
281     this._root.remove();
282   }
283
284   _remove() {
285     this._root.remove();
286   }
287
288   _remove() {
289     this._root.remove();
290   }
291
292   _remove() {
293     this._root.remove();
294   }
295
296   _remove() {
297     this._root.remove();
298   }
299
300   _remove() {
301     this._root.remove();
302   }
303
304   _remove() {
305     this._root.remove();
306   }
307
308   _remove() {
309     this._root.remove();
310   }
311
312   _remove() {
313     this._root.remove();
314   }
315
316   _remove() {
317     this._root.remove();
318   }
319
320   _remove() {
321     this._root.remove();
322   }
323
324   _remove() {
325     this._root.remove();
326   }
327
328   _remove() {
329     this._root.remove();
330   }
331
332   _remove() {
333     this._root.remove();
334   }
335
336   _remove() {
337     this._root.remove();
338   }
339
340   _remove() {
341     this._root.remove();
342   }
343
344   _remove() {
345     this._root.remove();
346   }
347
348   _remove() {
349     this._root.remove();
350   }
351
352   _remove() {
353     this._root.remove();
354   }
355
356   _remove() {
357     this._root.remove();
358   }
359
360   _remove() {
361     this._root.remove();
362   }
363
364   _remove() {
365     this._root.remove();
366   }
367
368   _remove() {
369     this._root.remove();
370   }
371
372   _remove() {
373     this._root.remove();
374   }
375
376   _remove() {
377     this._root.remove();
378   }
379
380   _remove() {
381     this._root.remove();
382   }
383
384   _remove() {
385     this._root.remove();
386   }
387
388   _remove() {
389     this._root.remove();
390   }
391
392   _remove() {
393     this._root.remove();
394   }
395
396   _remove() {
397     this._root.remove();
398   }
399
400   _remove() {
401     this._root.remove();
402   }
403
404   _remove() {
405     this._root.remove();
406   }
407
408   _remove() {
409     this._root.remove();
410   }
411
412   _remove() {
413     this._root.remove();
414   }
415
416   _remove() {
417     this._root.remove();
418   }
419
420   _remove() {
421     this._root.remove();
422   }
423
424   _remove() {
425     this._root.remove();
426   }
427
428   _remove() {
429     this._root.remove();
430   }
431
432   _remove() {
433     this._root.remove();
434   }
435
436   _remove() {
437     this._root.remove();
438   }
439
440   _remove() {
441     this._root.remove();
442   }
443
444   _remove() {
445     this._root.remove();
446   }
447
448   _remove() {
449     this._root.remove();
450   }
451
452   _remove() {
453     this._root.remove();
454   }
455
456   _remove() {
457     this._root.remove();
458   }
459
460   _remove() {
461     this._root.remove();
462   }
463
464   _remove() {
465     this._root.remove();
466   }
467
468   _remove() {
469     this._root.remove();
470   }
471
472   _remove() {
473     this._root.remove();
474   }
475
476   _remove() {
477     this._root.remove();
478   }
479
480   _remove() {
481     this._root.remove();
482   }
483
484   _remove() {
485     this._root.remove();
486   }
487
488   _remove() {
489     this._root.remove();
490   }
491
492   _remove() {
493     this._root.remove();
494   }
495
496   _remove() {
497     this._root.remove();
498   }
499
500   _remove() {
501     this._root.remove();
502   }
503
504   _remove() {
505     this._root.remove();
506   }
507
508   _remove() {
509     this._root.remove();
510   }
511
512   _remove() {
513     this._root.remove();
514   }
515
516   _remove() {
517     this._root.remove();
518   }
519
520   _remove() {
521     this._root.remove();
522   }
523
524   _remove() {
525     this._root.remove();
526   }
527
528   _remove() {
529     this._root.remove();
530   }
531
532   _remove() {
533     this._root.remove();
534   }
535
536   _remove() {
537     this._root.remove();
538   }
539
540   _remove() {
541     this._root.remove();
542   }
543
544   _remove() {
545     this._root.remove();
546   }
547
548   _remove() {
549     this._root.remove();
550   }
551
552   _remove() {
553     this._root.remove();
554   }
555
556   _remove() {
557     this._root.remove();
558   }
559
560   _remove() {
561     this._root.remove();
562   }
563
564   _remove() {
565     this._root.remove();
566   }
567
568   _remove() {
569     this._root.remove();
570   }
571
572   _remove() {
573     this._root.remove();
574   }
575
576   _remove() {
577     this._root.remove();
578   }
579
580   _remove() {
581     this._root.remove();
582   }
583
584   _remove() {
585     this._root.remove();
586   }
587
588   _remove() {
589     this._root.remove();
590   }
591
592   _remove() {
593     this._root.remove();
594   }
595
596   _remove() {
597     this._root.remove();
598   }
599
600   _remove() {
601     this._root.remove();
602   }
603
604   _remove() {
605     this._root.remove();
606   }
607
608   _remove() {
609     this._root.remove();
610   }
611
612   _remove() {
613     this._root.remove();
614   }
615
616   _remove() {
617     this._root.remove();
618   }
619
620   _remove() {
621     this._root.remove();
622   }
623
624   _remove() {
625     this._root.remove();
626   }
627
628   _remove() {
629     this._root.remove();
630   }
631
632   _remove() {
633     this._root.remove();
634   }
635
636   _remove() {
637     this._root.remove();
638   }
639
640   _remove() {
641     this._root.remove();
642   }
643
644   _remove() {
645     this._root.remove();
646   }
647
648   _remove() {
649     this._root.remove();
650   }
651
652   _remove() {
653     this._root.remove();
654   }
655
656   _remove() {
657     this._root.remove();
658   }
659
660   _remove() {
661     this._root.remove();
662   }
663
664   _remove() {
665     this._root.remove();
666   }
667
668   _remove() {
669     this._root.remove();
670   }
671
672   _remove() {
673     this._root.remove();
674   }
675
676   _remove() {
677     this._root.remove();
678   }
679
680   _remove() {
681     this._root.remove();
682   }
683
684   _remove() {
685     this._root.remove();
686   }
687
688   _remove() {
689     this._root.remove();
690   }
691
692   _remove() {
693     this._root.remove();
694   }
695
696   _remove() {
697     this._root.remove();
698   }
699
700   _remove() {
701     this._root.remove();
702   }
703
704   _remove() {
705     this._root.remove();
706   }
707
708   _remove() {
709     this._root.remove();
710   }
711
712   _remove() {
713     this._root.remove();
714   }
715
716   _remove() {
717     this._root.remove();
718   }
719
720   _remove() {
721     this._root.remove();
722   }
723
724   _remove() {
725     this._root.remove();
726   }
727
728   _remove() {
729     this._root.remove();
730   }
731
732   _remove() {
733     this._root.remove();
734   }
735
736   _remove() {
737     this._root.remove();
738   }
739
740   _remove() {
741     this._root.remove();
742   }
743
744   _remove() {
745     this._root.remove();
746   }
747
748   _remove() {
749     this._root.remove();
750   }
751
752   _remove() {
753     this._root.remove();
754   }
755
756   _remove() {
757     this._root.remove();
758   }
759
760   _remove() {
761     this._root.remove();
762   }
763
764   _remove() {
765     this._root.remove();
766   }
767
768   _remove() {
769     this._root.remove();
770   }
771
772   _remove() {
773     this._root.remove();
774   }
775
776   _remove() {
777     this._root.remove();
778   }
779
780   _remove() {
781     this._root.remove();
782   }
783
784   _remove() {
785     this._root.remove();
786   }
787
788   _remove() {
789     this._root.remove();
790   }
791
792   _remove() {
793     this._root.remove();
794   }
795
796   _remove() {
797     this._root.remove();
798   }
799
800   _remove() {
801     this._root.remove();
802   }
803
804   _remove() {
805     this._root.remove();
806   }
807
808   _remove() {
809     this._root.remove();
810   }
811
812   _remove() {
813     this._root.remove();
814   }
815
816   _remove() {
817     this._root.remove();
818   }
819
820   _remove() {
821     this._root.remove();
822   }
823
824   _remove() {
825     this._root.remove();
826   }
827
828   _remove() {
829     this._root.remove();
830   }
831
832   _remove() {
833     this._root.remove();
834   }
835
836   _remove() {
837     this._root.remove();
838   }
839
840   _remove() {
841     this._root.remove();
842   }
843
844   _remove() {
845     this._root.remove();
846   }
847
848   _remove() {
849     this._root.remove();
850   }
851
852   _remove() {
853     this._root.remove();
854   }
855
856   _remove() {
857     this._root.remove();
858   }
859
860   _remove() {
861     this._root.remove();
862   }
863
864   _remove() {
865     this._root.remove();
866   }
867
868   _remove() {
869     this._root.remove();
870   }
871
872   _remove() {
873     this._root.remove();
874   }
875
876   _remove() {
877     this._root.remove();
878   }
879
880   _remove() {
881     this._root.remove();
882   }
883
884   _remove() {
885     this._root.remove();
886   }
887
888   _remove() {
889     this._root.remove();
890   }
891
892   _remove() {
893     this._root.remove();
894   }
895
896   _remove() {
897     this._root.remove();
898   }
899
900   _remove() {
901     this._root.remove();
902   }
903
904   _remove() {
905     this._root.remove();
906   }
907
908   _remove() {
909     this._root.remove();
910   }
911
912   _remove() {
913     this._root.remove();
914   }
915
916   _remove() {
917     this._root.remove();
918   }
919
920   _remove() {
921     this._root.remove();
922   }
923
924   _remove() {
925     this._root.remove();
926   }
927
928   _remove() {
929     this._root.remove();
930   }
931
932   _remove() {
933     this._root.remove();
934   }
935
936   _remove() {
937     this._root.remove();
938   }
939
940   _remove() {
941     this._root.remove();
942   }
943
944   _remove() {
945     this._root.remove();
946   }
947
948   _remove() {
949     this._root.remove();
950   }
951
952   _remove() {
953     this._root.remove();
954   }
955
956   _remove() {
957     this._root.remove();
958   }
959
960   _remove() {
961     this._root.remove();
962   }
963
964   _remove() {
965     this._root.remove();
966   }
967
968   _remove() {
969     this._root.remove();
970   }
971
972   _remove() {
973     this._root.remove();
974   }
975
976   _remove() {
977     this._root.remove();
978   }
979
980   _remove() {
981     this._root.remove();
982   }
983
984   _remove() {
985     this._root.remove();
986   }
987
988   _remove() {
989     this._root.remove();
990   }
991
992   _remove() {
993     this._root.remove();
994   }
995
996   _remove() {
997     this._root.remove();
998   }
999
1000   _remove() {
1001     this._root.remove();
1002   }
1003
1004   _remove() {
1005     this._root.remove();
1006   }
1007
1008   _remove() {
1009     this._root.remove();
1010   }
1011
1012   _remove() {
1013     this._root.remove();
1014   }
1015
1016   _remove() {
1017     this._root.remove();
1018   }
1019
1020   _remove() {
1021     this._root.remove();
1022   }
1023
1024   _remove() {
1025     this._root.remove();
1026   }
1027
1028   _remove() {
1029     this._root.remove();
1030   }
1031
1032   _remove() {
1033     this._root.remove();
1034   }
1035
1036   _remove() {
1037     this._root.remove();
1038   }
1039
1040   _remove() {
1041     this._root.remove();
1042   }
1043
1044   _remove() {
1045     this._root.remove();
1046   }
1047
1048   _remove() {
1049     this._root.remove();
1050   }
1051
1052   _remove() {
1053     this._root.remove();
1054   }
1055
1056   _remove() {
1057     this._root.remove();
1058   }
1059
1060   _remove() {
1061     this._root.remove();
1062   }
1063
1064   _remove() {
1065     this._root.remove();
1066   }
1067
1068   _remove() {
1069     this._root.remove();
1070   }
1071
1072   _remove() {
1073     this._root.remove();
1074   }
1075
1076   _remove() {
1077     this._root.remove();
1078   }
1079
1080   _remove() {
1081     this._root.remove();
1082   }
1083
1084   _remove() {
1085     this._root.remove();
1086   }
1087
1088   _remove() {
1089     this._root.remove();
1090   }
1091
1092   _remove() {
1093     this._root.remove();
1094   }
1095
1096   _remove() {
1097     this._root.remove();
1098   }
1099
1100   _remove() {
1101     this._root.remove();
1102   }
1103
1104   _remove() {
1105     this._root.remove();
1106   }
1107
1108   _remove() {
1109     this._root.remove();
1110   }
1111
1112   _remove() {
1113     this._root.remove();
1114   }
1115
1116   _remove() {
1117     this._root.remove();
1118   }
1119
1120   _remove() {
1121     this._root.remove();
1122   }
1123
1124   _remove() {
1125     this._root.remove();
1126   }
1127
1128   _remove() {
1129     this._root.remove();
1130   }
1131
1132   _remove() {
1133     this._root.remove();
1134   }
1135
1136   _remove() {
1137     this._root.remove();
1138   }
1139
1140   _remove() {
1141     this._root.remove();
1142   }
1143
1144   _remove() {
1145     this._root.remove();
1146   }
1147
1148   _remove() {
1149     this._root.remove();
1150   }
1151
1152   _remove() {
1153     this._root.remove();
1154   }
1155
1156   _remove() {
1157     this._root.remove();
1158   }
1159
1160   _remove() {
1161     this._root.remove();
1162   }
1163
1164   _remove() {
1165     this._root.remove();
1166   }
1167
1168   _remove() {
1169     this._root.remove();
1170   }
1171
1172   _remove() {
1173     this._root.remove();
1174   }
1175
1176   _remove() {
1177     this._root.remove();
1178   }
1179
1180   _remove() {
1181     this._root.remove();
1182   }
1183
1184   _remove() {
1185     this._root.remove();
1186   }
1187
1188   _remove() {
1189     this._root.remove();
1190   }
1191
1192   _remove() {
1193     this._root.remove();
1194   }
1195
1196   _remove() {
1197     this._root.remove();
1198   }
1199
1200   _remove() {
1201     this._root.remove();
1202   }
1203
1204   _remove() {
1205     this._root.remove();
1206   }
1207
1208   _remove() {
1209     this._root.remove();
1210   }
1211
1212   _remove() {
1213     this._root.remove();
1214   }
1215
1216   _remove() {
1217     this._root.remove();
1218   }
1219
1220   _remove() {
1221     this._root.remove();
1222   }
1223
1224   _remove() {
1225     this._root.remove();
1226   }
1227
1228   _remove() {
1229     this._root.remove();
1230   }
1231
1232   _remove() {
1233     this._root.remove();
1234   }
1235
1236   _remove() {
1237     this._root.remove();
1238   }
1239
1240   _remove() {
1241     this._root.remove();
1242   }
1243
1244   _remove() {
1245     this._root.remove();
1246   }
1247
1248   _remove() {
1249     this._root.remove();
1250   }
1251
1252   _remove() {
1253     this._root.remove();
1254   }
1255
1256   _remove() {
1257     this._root.remove();
1258   }
1259
1260   _remove() {
1261     this._root.remove();
1262   }
1263
1264   _remove() {
1265     this._root.remove();
1266   }
1267
1268   _remove() {
1269     this._root.remove();
1270   }
1271
1272   _remove() {
1273     this._root.remove();
1274   }
1275
1276   _remove() {
1277     this._root.remove();
1278   }
1279
1280   _remove() {
1281     this._root.remove();
1282   }
1283
1284   _remove() {
1285     this._root.remove();
1286   }
1287
1288   _remove() {
1289     this._root.remove();
1290   }
1291
1292   _remove() {
1293     this._root.remove();
1294   }
1295
1296   _remove() {
1297     this._root.remove();
1298   }
1299
1300   _remove() {
1301     this._root.remove();
1302   }
1303
1304   _remove() {
1305     this._root.remove();
1306   }
1307
1308   _remove() {
1309     this._root.remove();
1310   }
1311
1312   _remove() {
1313     this._root.remove();
1314   }
1315
1316   _remove() {
1317     this._root.remove();
1318   }
1319
1320   _remove() {
1321     this._root.remove();
1322   }
1323
1324   _remove() {
1325     this._root.remove();
1326   }
1327
1328   _remove() {
1329     this._root.remove();
1330   }
1331
1332   _remove() {
1333     this._root.remove();
1334   }
1335
1336   _remove() {
1337     this._root.remove();
1338   }
1339
1340   _remove() {
1341     this._root.remove();
1342   }
1343
1344   _remove() {
1345     this._root.remove();
1346   }
1347
1348   _remove() {
1349     this._root.remove();
1350   }
1351
1352   _remove() {
1353     this._root.remove();
1354   }
1355
1356   _remove() {
1357     this._root.remove();
1358   }
1359
1360   _remove() {
1361     this._root.remove();
1362   }
1363
1364   _remove() {
1365     this._root.remove();
1366   }
1367
1368   _remove() {
1369     this._root.remove();
1370   }
1371
1372   _remove() {
1373     this._root.remove();
1374   }
1375
1376   _remove() {
1377     this._root.remove();
1378   }
1379
1380   _remove() {
1381     this._root.remove();
1382   }
1383
1384   _remove() {
1385     this._root.remove();
1386   }
1387
1388   _remove() {
1389     this._root.remove();
1390   }
1391
1392   _remove() {
1393     this._root.remove();
1394   }
1395
1396   _remove() {
1397     this._root.remove();
1398   }
1399
1400   _remove() {
1401     this._root.remove();
1402   }
1403
1404   _remove() {
1405     this._root.remove();
1406   }
1407
1408   _remove() {
1409     this._root.remove();
1410   }
1411
1412   _remove() {
1413     this._root.remove();
1414   }
1415
1416   _remove() {
1417     this._root.remove();
1418   }
1419
1420   _remove() {
1421     this._root.remove();
1422   }
1423
1424   _remove() {
1425     this._root.remove();
1426   }
1427
1428   _remove() {
1429     this._root.remove();
1430   }
1431
1432   _remove() {
1433     this._root.remove();
1434   }
1435
1436   _remove() {
1437     this._root.remove();
1438   }
1439
1440   _remove() {
1441     this._root.remove();
1442   }
1443
1444   _remove() {
1445     this._root.remove();
1446   }
1447
1448   _remove() {
1449     this._root.remove();
1450   }
1451
1452   _remove() {
1453     this._root.remove();
1454   }
1455
1456   _remove() {
1457     this._root.remove();
1458   }
1459
1460   _remove() {
1461     this._root.remove();
1462   }
1463
1464   _remove() {
1465     this._root.remove();
1466   }
1467
1468   _remove() {
1469     this._root.remove();
1470   }
1471
1472   _remove() {
1473     this._root.remove();
1474   }
1475
1476   _remove() {
1477     this._root.remove();
1478   }
1479
1480   _remove() {
1481     this._root.remove();
1482   }
1483
1484   _remove() {
1485     this._root.remove();
1486   }
1487
1488   _remove() {
1489     this._root.remove();
1490   }
1491
1492   _remove() {
1493     this._root.remove();
1494   }
1495
1496   _remove() {
1497     this._root.remove();
1498   }
1499
1500   _remove() {
1501     this._root.remove();
1502   }
1503
1504   _remove() {
1505     this._root.remove();
1506   }
1507
1508   _remove() {
1509     this._root.remove();
1510   }
1511
1512   _remove() {
1513     this._root.remove();
1514   }
1515
1516   _remove() {
1517     this._root.remove();
1518   }
1519
1520   _remove() {
1521     this._root.remove();
1522   }
1523
1524   _remove() {
1525     this._root.remove();
1526   }
1527
1528   _remove() {
1529     this._root.remove();
1530   }
1531
1532   _remove() {
1533     this._root.remove();
1534   }
1535
1536   _remove() {
1537     this._root.remove();
1538   }
1539
1540   _remove() {
1541     this._root.remove();
1542   }
1543
1544   _remove() {
1545     this._root.remove();
1546   }
1547
1548   _remove() {
1549     this._root.remove();
1550   }
1551
1552   _remove() {
1553     this._root.remove();
1554   }
1555
1556   _remove() {
1557     this._root.remove();
1558   }
1559
1560   _remove() {
1561     this._root.remove();
1562   }
1563
1564   _remove() {
1565     this._root.remove();
1566   }
1567
1568   _remove() {
1569     this._root.remove();
1570   }
1571
1572   _remove() {
1573     this._root.remove();
1574   }
1575
1576   _remove() {
1577     this._root.remove();
1578   }
1579
1580   _remove() {
1581     this._root.remove();
1582   }
1583
1584   _remove() {
1585     this._root.remove();
1586   }
1587
1588   _remove() {
1589     this._root.remove();
1590   }
1591
1592   _remove() {
1593     this._root.remove();
1594   }
1595
1596   _remove() {
1597     this._root.remove();
1598   }
1599
1600   _remove() {
1601     this._root.remove();
1602   }
1603
1604   _remove() {
1605     this._root.remove();
1606   }
1607
1608   _remove() {
1609     this._root.remove();
1610   }
1611
1612   _remove() {
1613     this._root.remove();
1614   }
1615
1616   _remove() {
1617     this._root.remove();
1618   }
1619
1620   _remove() {
1621     this._root.remove();
1622   }
1623
1624   _remove() {
1625     this._root.remove();
1626   }
1627
1628   _remove() {
1629     this._root.remove();
1630   }
1631
1632   _remove() {
1633     this._root.remove();
1634   }
1635
1636   _remove() {
1637     this._root.remove();
1638   }
1639
1640   _remove() {
1641     this._root.remove();
1642   }
1643
1644   _remove() {
1645     this._root.remove();
1646   }
1647
1648   _remove() {
1649     this._root.remove();
1650   }
1651
1652   _remove() {
1653     this._root.remove();
1654   }
1655
1656   _remove() {
1657     this._root.remove();
1658   }
1659
1660   _remove() {
1661     this._root.remove();
1662   }
1663
1664   _remove() {
1665     this._root.remove();
1666   }
1667
1668   _remove() {
1669     this._root.remove();
1670   }
1671
1672   _remove() {
1673     this._root.remove();
1674   }
1675
1676   _remove() {
1677     this._root.remove();
1678   }
1679
1680   _remove() {
1681     this._root.remove();
1682   }
1683
1684   _remove() {
1685     this._root.remove();
1686   }
1687
1688   _remove() {
1689     this._root.remove();
1690   }
1691
1692   _remove() {
1693     this._root.remove();
1694   }
1695
1696   _remove() {
1697     this._root.remove();
1698   }
1699
1700   _remove() {
1701     this._root.remove();
1702   }
1703
1704   _remove() {
1705     this._root.remove();
1706   }
1707
1708   _remove() {
1709     this._root.remove();
1710   }
1711
1712   _remove() {
1713     this._root.remove();
1714   }
1715
1716   _remove() {
1717     this._root.remove();
1718   }
1719
1720   _remove() {
1721     this._root.remove();
1722   }
1723
1724   _remove() {
1725     this._root.remove();
1726   }
1727
1728   _remove() {
1729     this._root.remove();
1730   }
1731
1732   _remove() {
1733     this._root.remove();
1734   }
1735
1736   _remove() {
1737     this._root.remove();
1738   }
1739
1740   _remove() {
1741     this._root.remove();
1742   }
1743
1744   _remove() {
1745     this._root.remove();
1746   }
1747
1748   _remove() {
1749     this._root.remove();
1750   }
1751
1752   _remove() {
1753     this._root.remove();
1754   }
1755
1756   _remove()
```

Polymer 2

```
1 <link rel="import" href="../../bower_components/polymer/polymer.html">
2 <dom-module id="todo-item">
3   <template>
4     <style>
5       <!-->
6       <!-->
7       <li class="{$item [isCompleted (checked)]}">
8         <input type="checkbox" value="{$(checked)}" checked="{$(checked)}" />
9         <label>{$text}</label>
10        <button class="destroy" on-click="handleOnRemove">x</button>
11      </li>
12    </template>
13    <script>
14      class TodoItem extends Polymer.Element {
15        static get is() { return 'todo-item'; }
16        static get properties() {
17          return {
18            checked: { type: Boolean, value: false },
19            index: { type: Number, },
20            text: { type: String, value: '' }
21          };
22        }
23        handleOnRemove(e) {
24          this.dispatchEvent(new CustomEvent('remove', { detail: e }));
25        }
26        handleOnChecked(e) {
27          this.dispatchEvent(new CustomEvent('toggle', { detail: e }));
28        }
29        isCompleted(completed) {
30          return completed ? 'completed' : '';
31        }
32      }
33      window.customElements.define(TodoItem.is, TodoItem);
34    </script>
35  </dom-module>
```

Angular Elements

```
1 import { Component, EventEmitter, Input, Output, ViewEncapsulation } from '@angular/core';
2
3 @Component({
4   selector: 'todo-item',
5   template: `
6     <li class="item" [class.completed]="checked">
7       <input type="checkbox" [checked]="checked" (change)="handleOnChecked($event)">
8       <label>{{text}}</label>
9       <button class="destroy" (click)="handleOnRemove($event)">x</button>
10    </li>
11  `,
12  styles: [
13    ''
14  ],
15  encapsulation: ViewEncapsulation.Native
16 })
17 export class TodoItem {
18   @Input() checked: boolean;
19   @Input() text: string;
20   @Input() index: number;
21   @Output() onTodoItemChecked = new EventEmitter<number>();
22   @Output() onTodoItemRemove = new EventEmitter<number>();
23
24   handleOnRemove = () => this.onTodoItemRemove.emit(this.index);
25   handleOnChecked = () => this.onTodoItemChecked.emit(this.index);
26 }
27
28
29
30
31 // @jsx h
32 import { props } from "skatejs/dist/esnext";
33 import { h } from "preact";
34 import { Component } from "@util";
35
36 export default class extends Component {
37   static events = ["check", "remove"];
38   static props = {
39     checked: props.boolean,
40     index: props.number
41   };
42
43   handleCheck = e => {
44     this.onCheck({ index: this.index, value: e.target.checked });
45   };
46   handleRemove = () => {
47     this.onRemove({ index: this.index });
48   };
49
50   render({ checked, handleCheck, handleRemove }) {
51     return (
52       <div>
53         <style>`-`</style>
54         <li class="item" checked="checked" style="background-color: ${checked ? "#ccc" : "#fff"}">
55           <input type="checkbox" checked="checked" on-change={handleCheck}>
56           <label>${this.text}</label>
57           <button onClick={handleRemove}>x</button>
58         </li>
59       </div>
60     );
61   }
62 }
63
```

StencilJS

```
import { Component, Prop, Event, EventEmitter } from '@stencil/core';

@Component({
  tag: 'todo-item',
  styleUrl: 'todo-item.scss',
  shadow: true,
})
export class TodoItem {
  @Prop() checked: boolean;
  @Prop() text: string;
  @Prop() index: number;
  @Event() onTodoItemChecked: EventEmitter;
  @Event() onTodoItemRemove: EventEmitter;

  handleOnRemove = () => this.onTodoItemRemove.emit(this.index);
  handleOnChecked = () => this.onTodoItemChecked.emit(this.index);

  render() {
    return (
      <li class={this.checked ? 'completed' : ''}>
        <input type="checkbox" checked={this.checked} onChange={this.handleOnChecked}/>
        <label>{this.text}</label>
        <button onClick={this.handleOnRemove}>x</button>
      </li>
    );
  }
}

SkateJS + lit-html

import { props } from "skatejs/dist/esnext";
import { html } from "lit-html/lib/lit-extended";
import { Component } from "./util";

export default class extends Component {
  static events = ["check", "remove"];
  static props = {
    checked: props.boolean,
    index: props.number
  };
  handleCheck = e => {
    this.onCheck({ index: this.index, value: e.target.checked });
  };
  handleRemove = () => {
    this.onRemove({ index: this.index });
  };
  render({ checked, handleCheck, handleRemove }) {
    return html`<style>--</style>
    <li class="${checked ? "completed" : ""}">
      <input type="checkbox" checked="${checked}" on-change=${(h) => handleCheck(h)}>
      <label>
        <slot></slot>
      </label>
      <button on-click="${handleRemove}">x</button>
    </li>
  `;
}

```

SkateJS + Preact

SkateJS + lit-html

```
import {Component, Prop} from '@stencil/core';

@Component({
  tag: 'my-first-component',
  styleUrl: 'my-first-component.scss'
})
export class MyComponent {
  // Indicate that name should be
  // a public property on the component
  @Prop() name: string;

  render() {
    // JSX
    return (<p>My name is {this.name}</p>);
  }
}
```

Pour démarrer

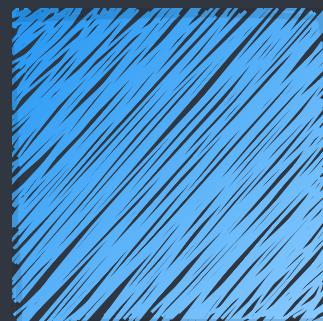


```
$ npm init stencil
```



```
? Pick a starter > - Use arrow-keys. Return to submit.  
> ionic-pwa      Everything you need to build fast, production ready PWAs  
    app           Minimal starter for building a Stencil app or website  
    component     Collection of web components that can be used anywhere
```

Lit-Elements



<https://lit-element.polymer-project.org/> <[https://lit-](https://lit-element.polymer-project.org/)
[element.polymer-project.org/](https://lit-element.polymer-project.org/)>



Projet **Open Source**, ➔ BSD 3-Clause License

<<https://github.com/Polymer/lit-element/blob/master/LICENSE>>

Créer par l'équipe **Polymer Team** en 2017

2.0k ⭐ sur github

Utilise la bibliothèque de template **lit-html**

<https://lit-html.polymer-project.org/> <<https://lit-html.polymer-project.org/>>

4.4k ⭐ sur github

Basé sur les **templates HTML**

Avec les ➔ *Template literals* <https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Template_literals> de ES2015

```
import {html, render} from 'lit-html';

// A lit-html template uses
// the `html` template tag:
const sayHello = (name) =>
  html`<h1>Hello ${name}</h1>`;

// It's rendered with the `render()` function:
render(sayHello('World'), document.body);

// And re-renders only update the
// data that changed, without VDOM diffing!
render(sayHello('Everyone'), document.body);
```



```
import { LitElement, html, css } from 'https://unpkg.com/lit-element/lit-element';

class MyElement extends LitElement {
    static get properties() {
        return {
            mood: { type: String }
        }
    }
    static get styles() {
        return css` .mood { color: green; } `;
    }
    render() {
        return html` Web Components are
            <span class="mood">${this.mood}</span>! `;
    }
}

customElements.define('my-element', MyElement);
```

```
import { /* ... */ } from 'lit-element';
import { html, css } from 'lit';

@customElement('my-element')
class MyElement extends LitElement {

    @property({ type: String }) mood;

    static get styles() {
        return css` .mood { color: green; } `;
    }

    render() {
        return html` Web Components are
            <span class="mood">${this.mood}</span>! `;
    }
}
```

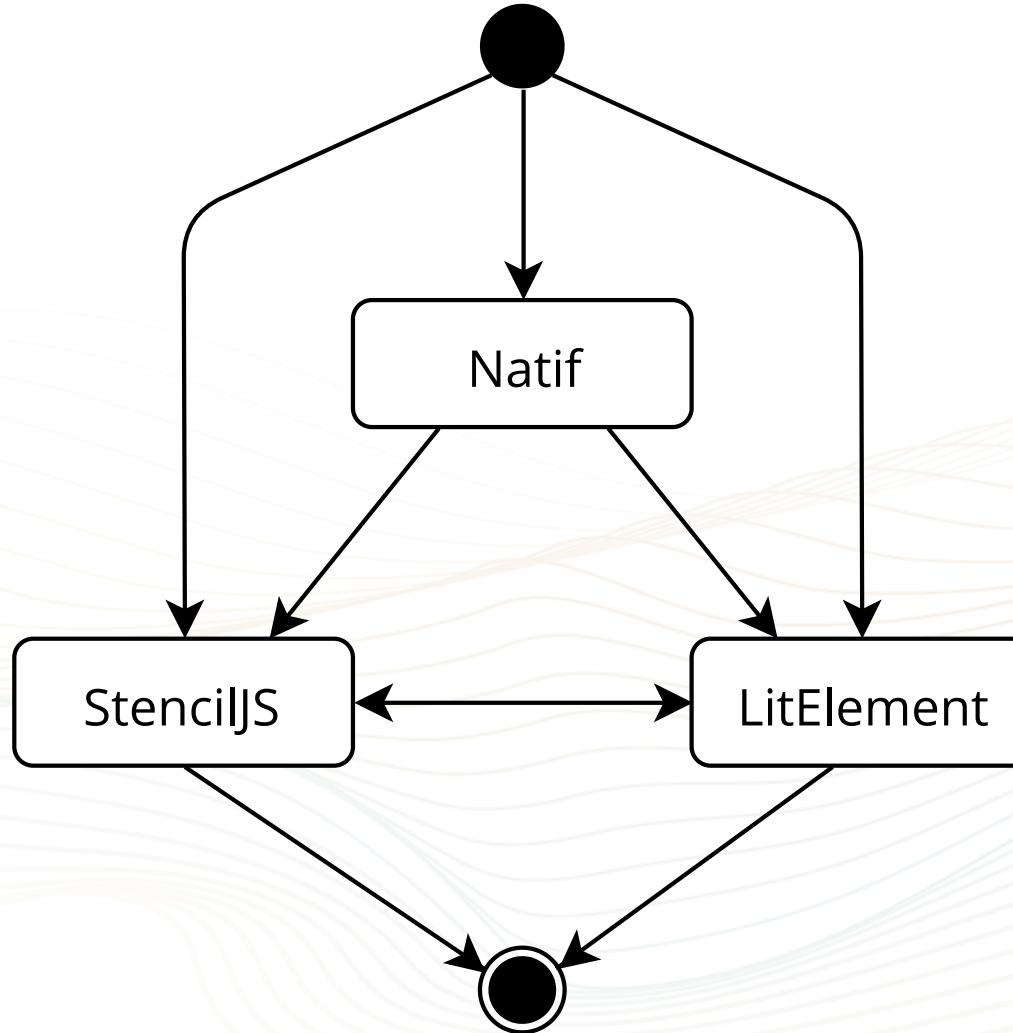
✓ Chrome, Safari, Opera, Firefox

 polyfills <<https://github.com/webcomponents/webcomponentsjs>> pour Edge et
IE 11

Workshop

@julienrenaux @ilaborie #DevoxxFR #webcomponents

DEVOXX
France



<http://bit.ly/devoxx-webc> <<http://bit.ly/devoxx-webc>>

Conclusion

@julienrenaux @ilaborie #DevoxxFR #webcomponents

DEVOXX
France

Les limites

Support des navigateurs

Ne remplacera pas vos frameworks

Accessibilité

Theming

Form validation / submission / autofocus

Les alternatives modernes

➡ SkateJS <<https://skatejs.netlify.com/>>

➡ Svelte <<https://svelte.technology/>>

➡ Slim.js <<http://slimjs.com>>

➡ Atomico <<https://github.com/atomicojs/core>>

...

Le future



➡ Pseudo elements :: part & :: theme

<<https://meowni.ca/posts/part-theme-explainer/>>

➡ Scoped Custom Element Registries

<<https://github.com/w3c/webcomponents/issues/716>>

SSR pour lit-html et litElement

...

Des liens

web-components/>

➡ **Web Components Todo** <<https://wc-todo.firebaseio.com/>>

➡ **Web Components specifications**

<<https://github.com/w3c/webcomponents>>

➡ **A curated list of awesome Web Components resources.** <<https://github.com/mateusortiz/webcomponents-the-right-way>>

➡ **A curated list of awesome lit-html resources.**

<<https://github.com/web-padawan/awesome-lit-html>>

➡ **Beyond the polyfills: how Web Components affect us today?** <<https://dev.to/webpadawan/beyond-the-polyfills-how-web-components-affect-us-today-3j0a>>

Fin

Merci

Pensez à nous faire des retours (votez !)