

Real-world Zoneless Optimizations



Brian Treece
Chief of User Experience

www.youtube.com/@briantreece | www.socreate.it

Real-world Situations



Key Concepts

Timers don't trigger change detection
Manual change detection is a quick fix
Signals are better!





Knowing how to connect state to updates



When Events Don't Trigger Change Detection



Events and Change Detection



Button clicks



Scroll events



Key Concepts

Imperative DOM listeners, no change detection

Manual change detection is a quick fix

Signals are better!





Working with Third-party Library Events



Third-party Libraries



Can't be changed



Different approach
needed!



Key Concepts

Third-party libraries may need help
Manual change detection is a quick fix
Async pipe is better
Signals are best!





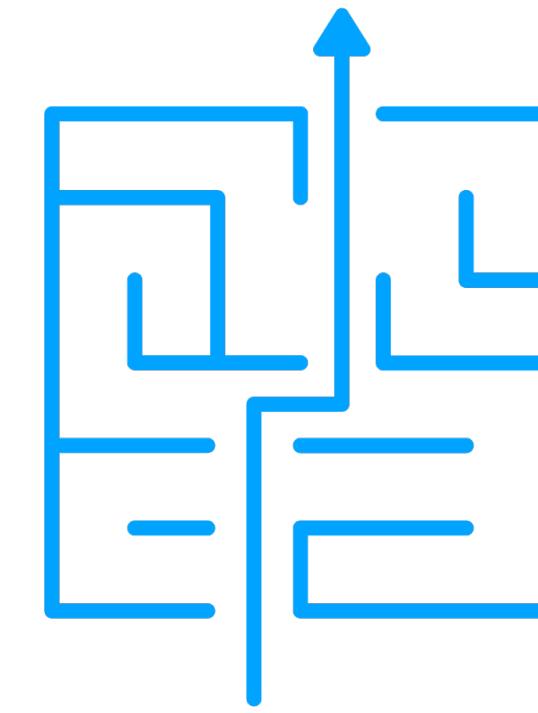
Reliably Updating Components Created at Runtime



Dynamic Components



Mutating properties
doesn't work!



We need
another way!



Key Concepts

Direct property mutation doesn't work!

Signals do!

Use `input()` and `setInput()`

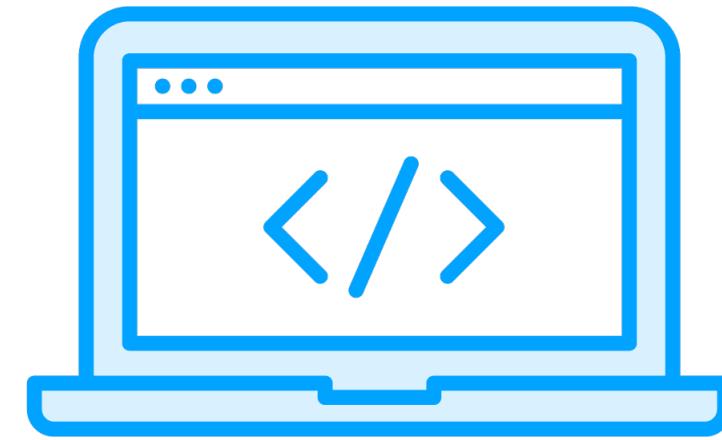




Catching Missed Updates in Development



provideCheckNoChangesConfig()



**Development specific
troubleshooting feature**



**Helps catch subtle
change detection issues**



Key Concepts

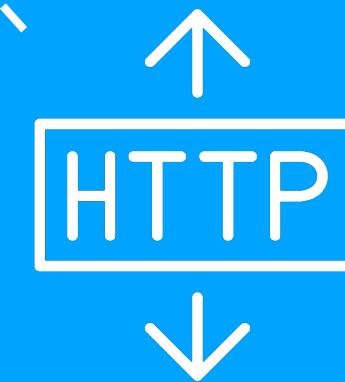
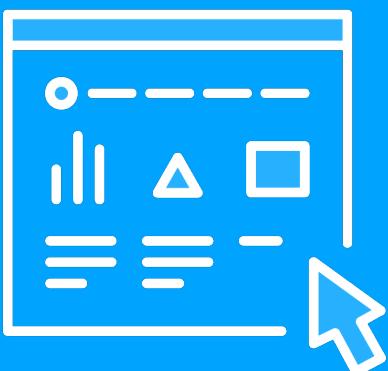
Don't bind from the DOM

Do bind from state

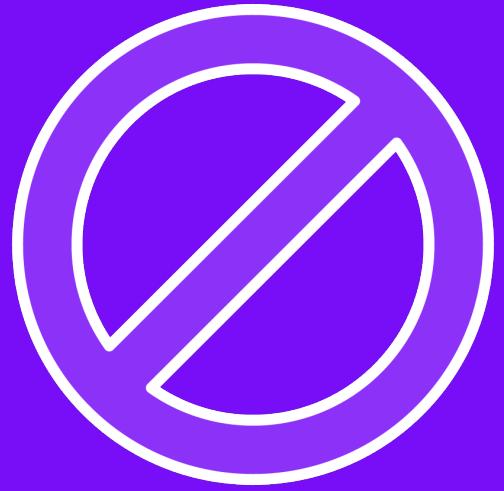
Use `provideCheckNoChangesConfig()`



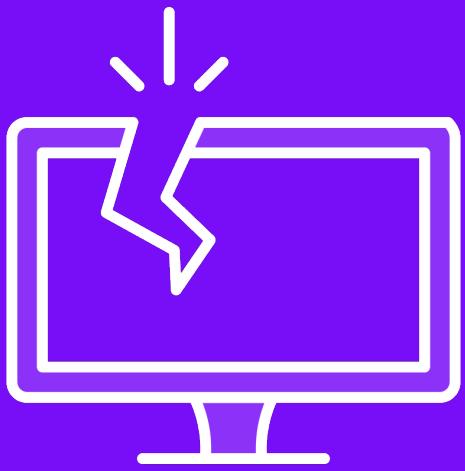
Zone.js



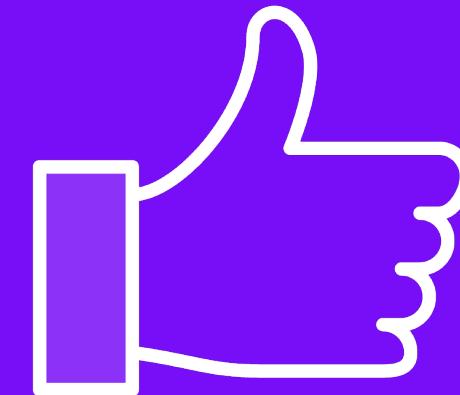
Zoneless Angular



No Zone.js



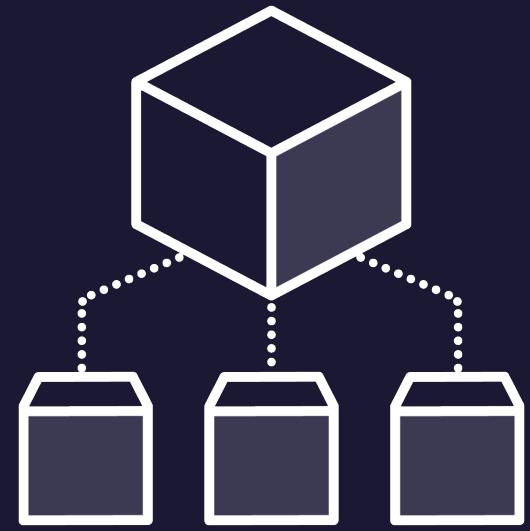
What breaks?



You're in charge



Fine-tuned Change Detection



OnPush



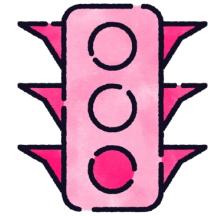
Manual change
detection



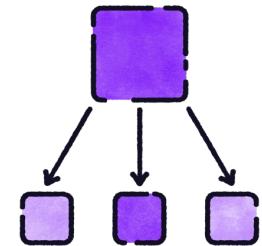
Signals!



What Now?



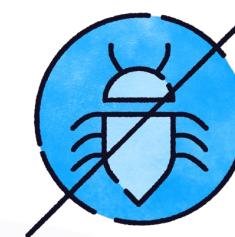
Use signals as much as possible!



Use OnPush!



Use manual change detection sparingly



Use provideCheckNoChangesConfig()



You Got This!



Thank you!

www.youtube.com/@briantreese | www.socreate.it

