

It's a Wrap

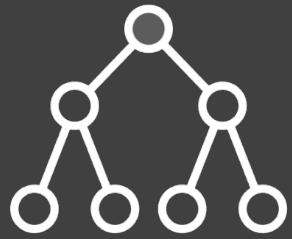


Brian Treece

CHIEF OF USER EXPERIENCE AT SOCREATE.IT

@brianmtreece





Modular & Componentized
Scoped & Isolated







Changing the Way We Style
A Component Structure
Shadow DOM & Style Encapsulation
Local Not Global
View Encapsulation: On, Off, or
ShadowDom



CSS Scoping

`:host`

`:host-context`

`::ng-deep`



Manipulate Styles Within the
Shadow DOM



Think Differently



Organized, Structured,
& Maintainable





We Don't Have to Change
If We Don't Want To



Instead...

Leverage the
Framework

Think Differently

Scalable &
Maintainable





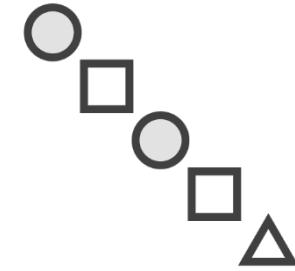
Global Styles Are Still Ok...
But They're Different Now



Naming Conventions Are Still Beneficial!



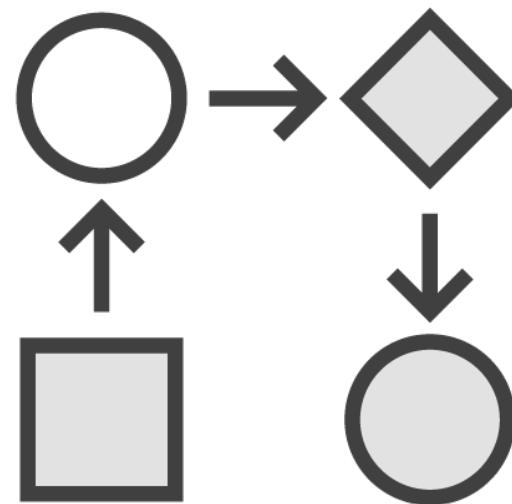
**Traditionally Applied
on a Global Level**



**Can Be Applied
Locally in Angular**



We Can Create a System for Relative Units!



And We Can Make It Predictable!

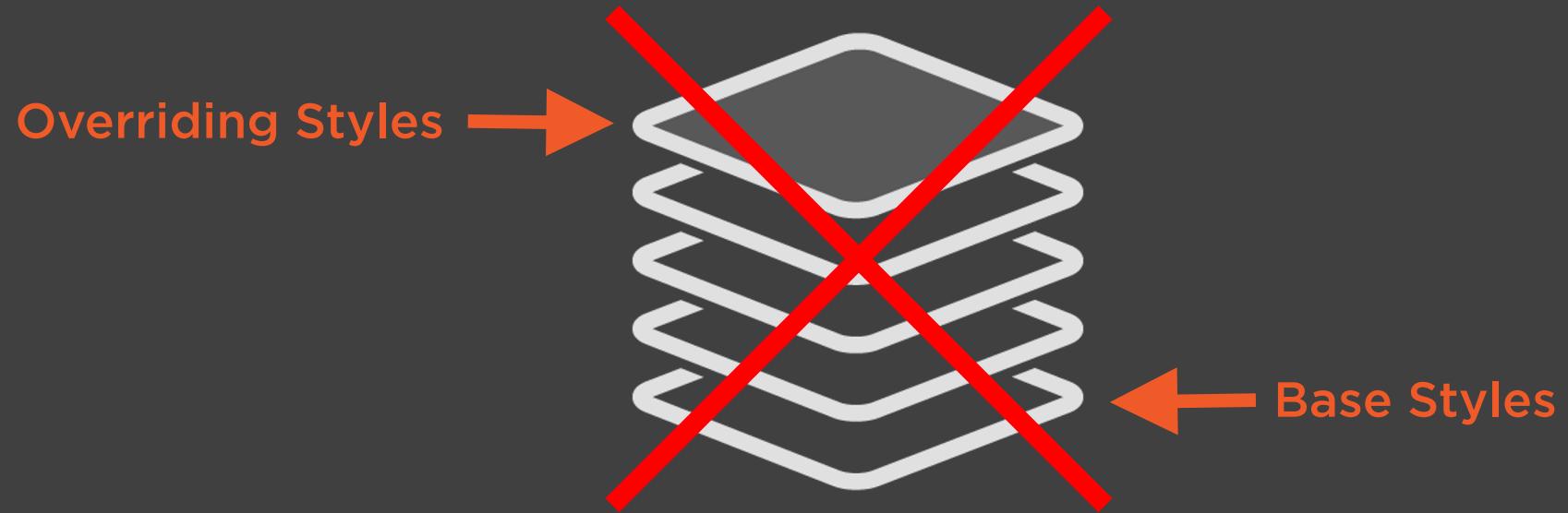




some.component.scss

```
├── _partial.scss  
├── _partial.scss  
├── _partial.scss  
└── _partial.scss
```

Overriding Styles



Don't Do It!





Keep Specificity Low



Sass

{less}

stylus



“Local First”



Sass

{less}

stylus

File ✓

Component ✓

Grouping of Components ✓

Global ✓





Components Need Themes





Layouts



Colors





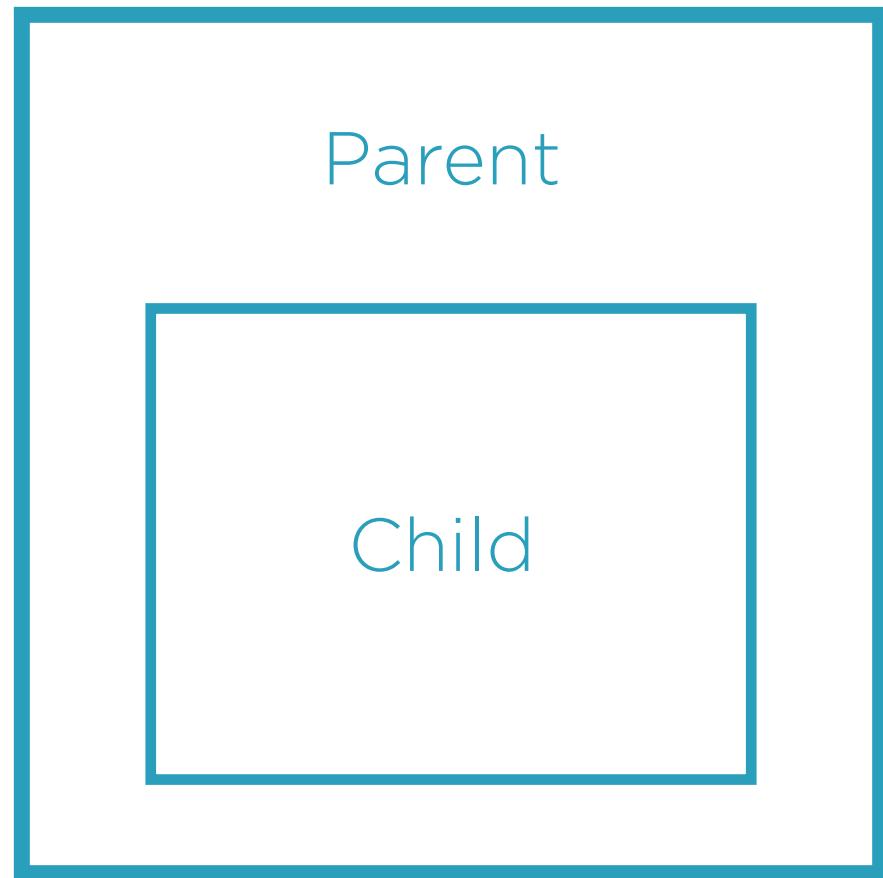
Using :host

Using :host-context

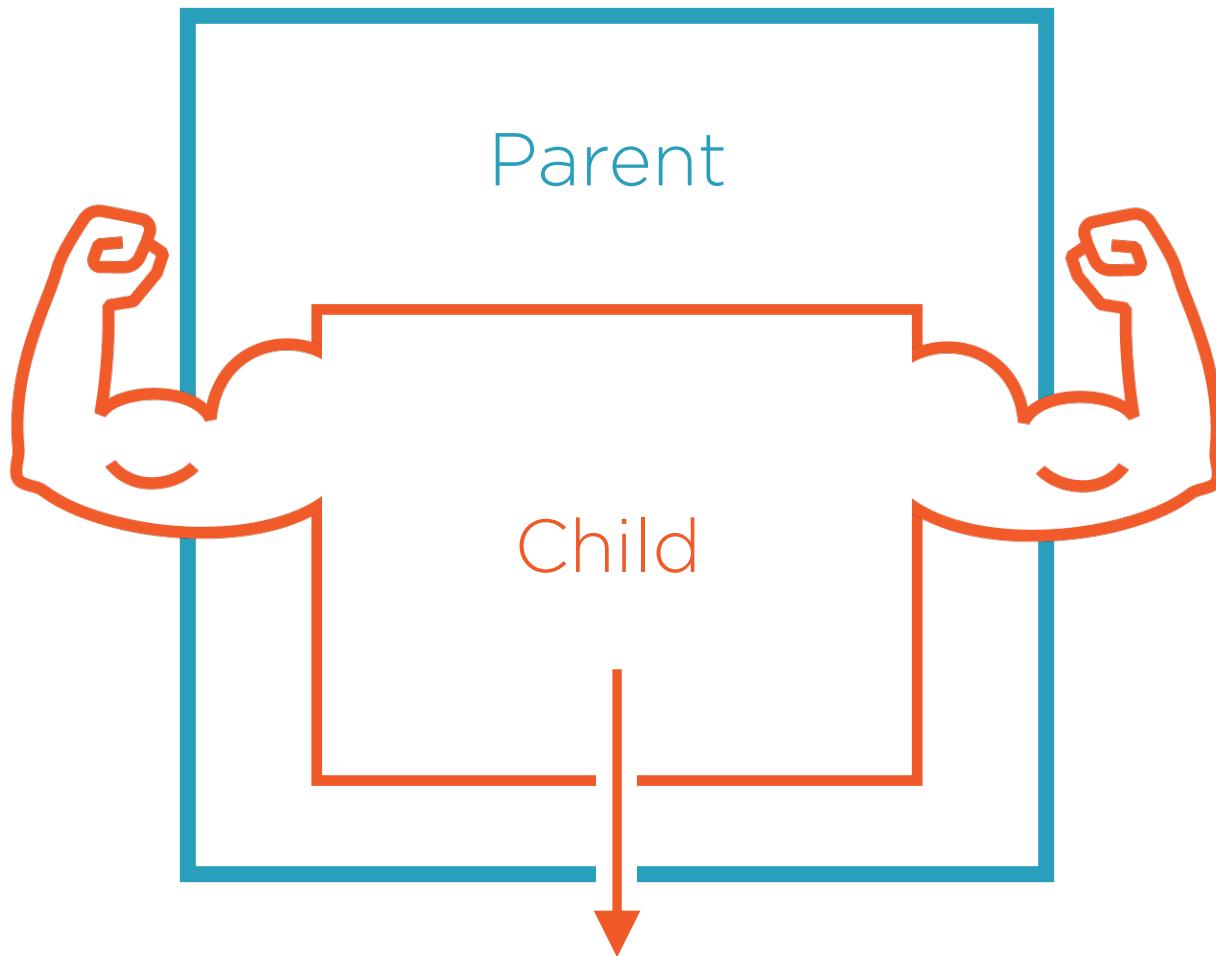
Using Logic in Templates



CSS Custom Properties



CSS Custom Properties



In Charge



Summary



How Styles Work in Angular
Organized, Scalable, & Maintainable CSS
Component Themes



That's All We Need!

Look Amazing

Work Well Today

Easy to Bring Forward

