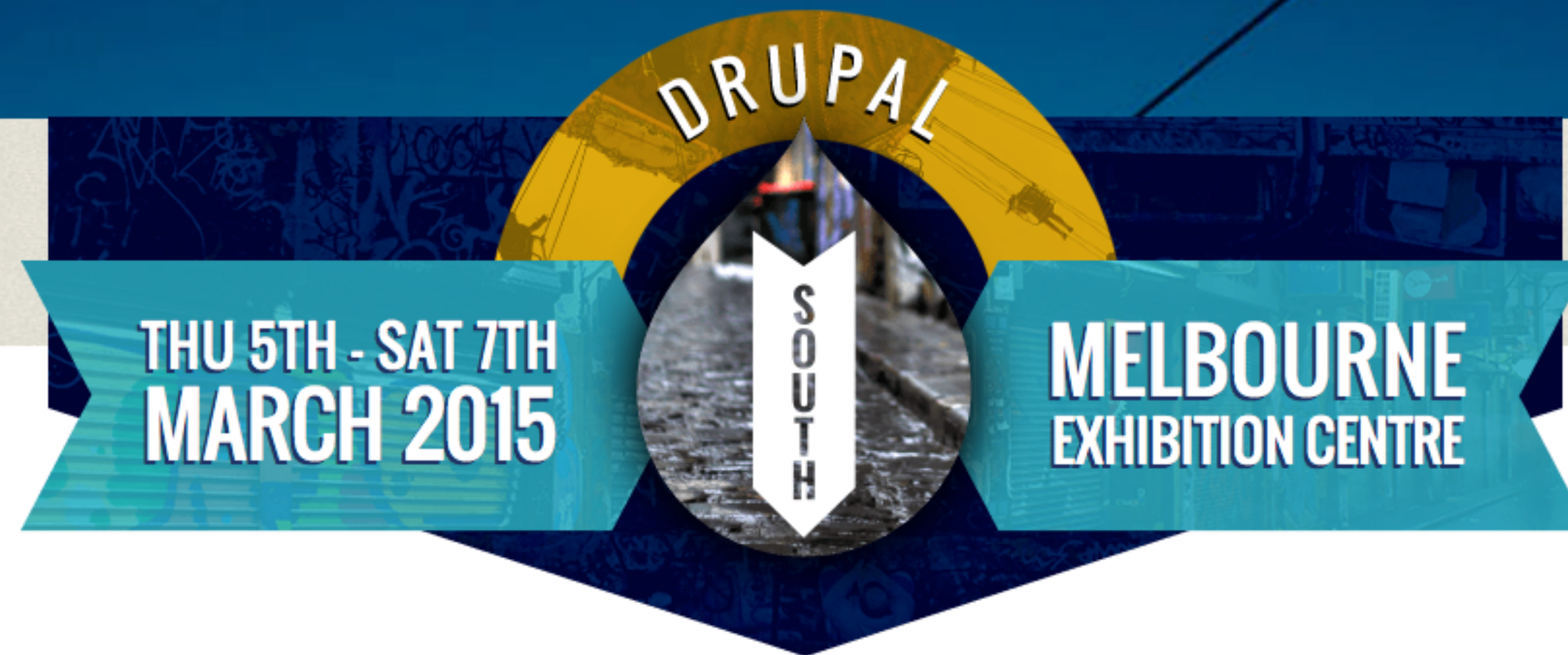




STYLE-GUIDE-DRIVEN DEVELOPMENT

JOHN ALBIN WILKINS

ALL THE TRACKS! – MARCH 5, 2015

PLATINUM SPONSOR

PREVIOUSNEXT®

@JOHNALBIN

Senior Front-end Developer

PREVIOUSNEXT®



FREE GIFTS FROM ME

Zen Grids
zengrids.com

Normalize.css for Sass/Compass
github.com/JohnAlbin/normalize-scss

KSS-node
github.com/kss-node/kss-node

git-svn-migrate
john.albin.net/git/git-svn-migrate

Zen theme for Drupal
drupal.org/project/zen



WHERE ARE WE HEADED?



**“ARE YOU NEW TO
FRONT-END WEB DEVELOPMENT?**

**HERE’S A SECRET:
NO ONE ELSE REALLY KNOWS
WHAT THEY’RE DOING EITHER.”**

– NICOLAS GALLAGHER

January, 2013



FRONT-END BLOG POSTS:

Vagrant
Twig / Handlerbars
npm shrinkwrap / Bundler
Travis
Component
Jenkins / CI
Grunt/Gulp
Sass / Less
task runner
build tools
KSS
Wraith
Behat / Selenium
Web Components
Node.js
regression testing
PageSpeed
YSlow
CSS/JS linting
Cucumber
CSS frameworks
Web Starter Kit
Bootstrap / Foundation
Yeoman / Bower
CSSLoad
scaffolding tools



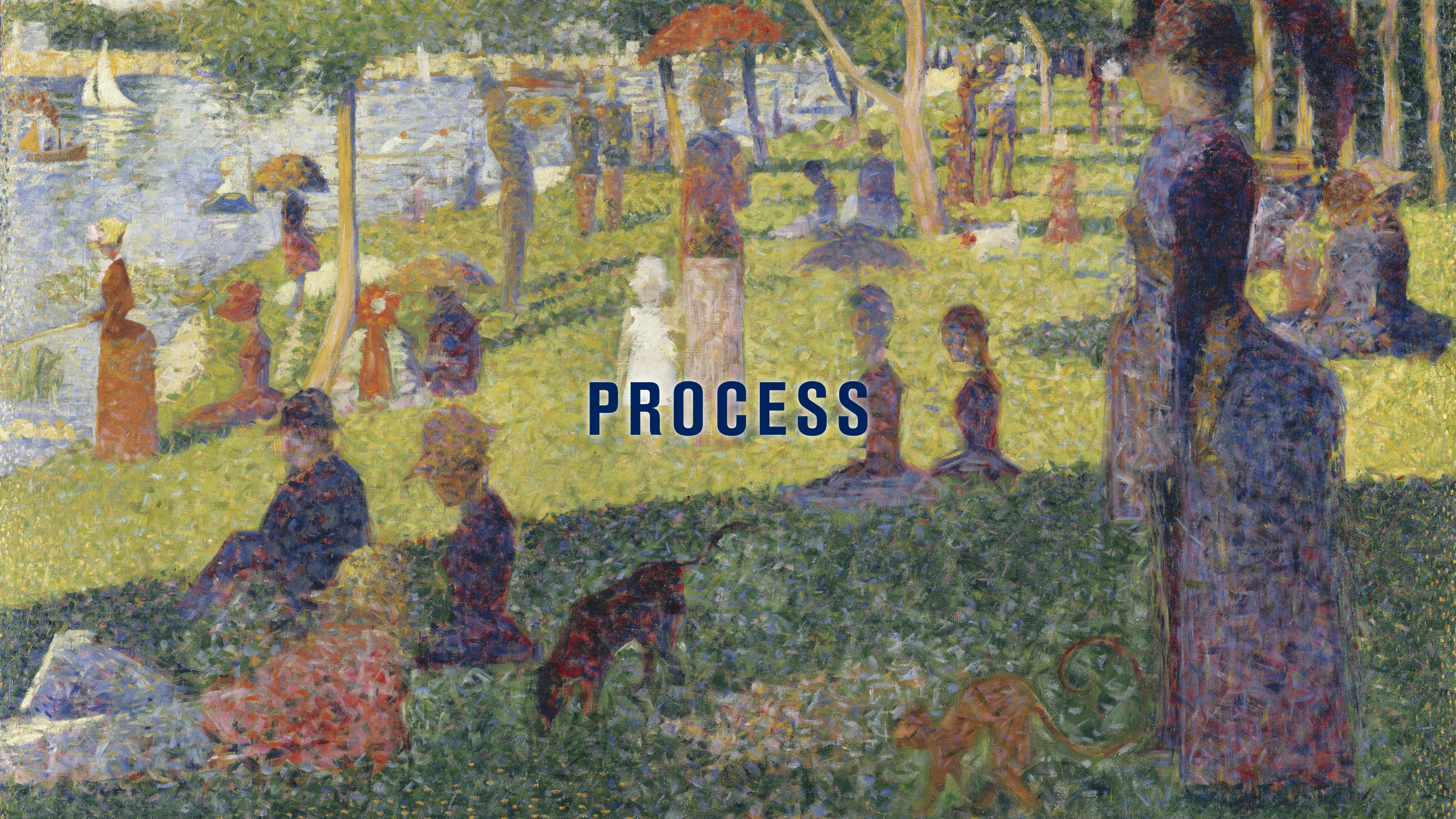
“Everyone is describing the one little piece they’ve created, but don’t explain (or even reference!) the larger concepts of how all of these elements link together.”

— Frank Chimero, July 2014 Designer News AMA



An abstract painting with a textured, impasto style. The background is a mix of light blue, yellow, and white strokes. On the right side, there are vertical strokes of dark blue and red. In the bottom right corner, there are more red and orange strokes. The overall composition is dynamic and colorful.

TECHNOLOGY



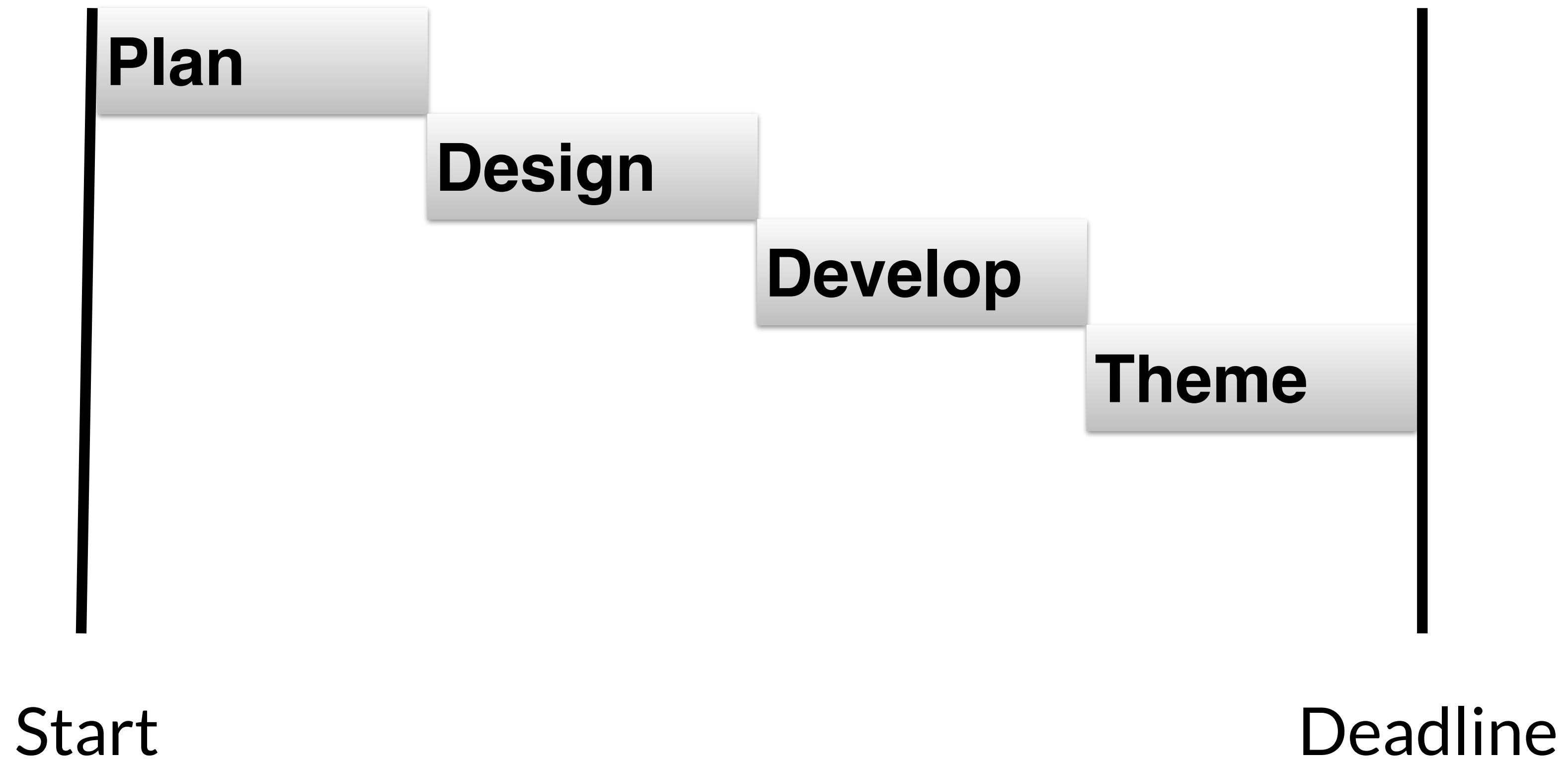
PROCESS

MAKING SENSE OF FRONT-END TECH

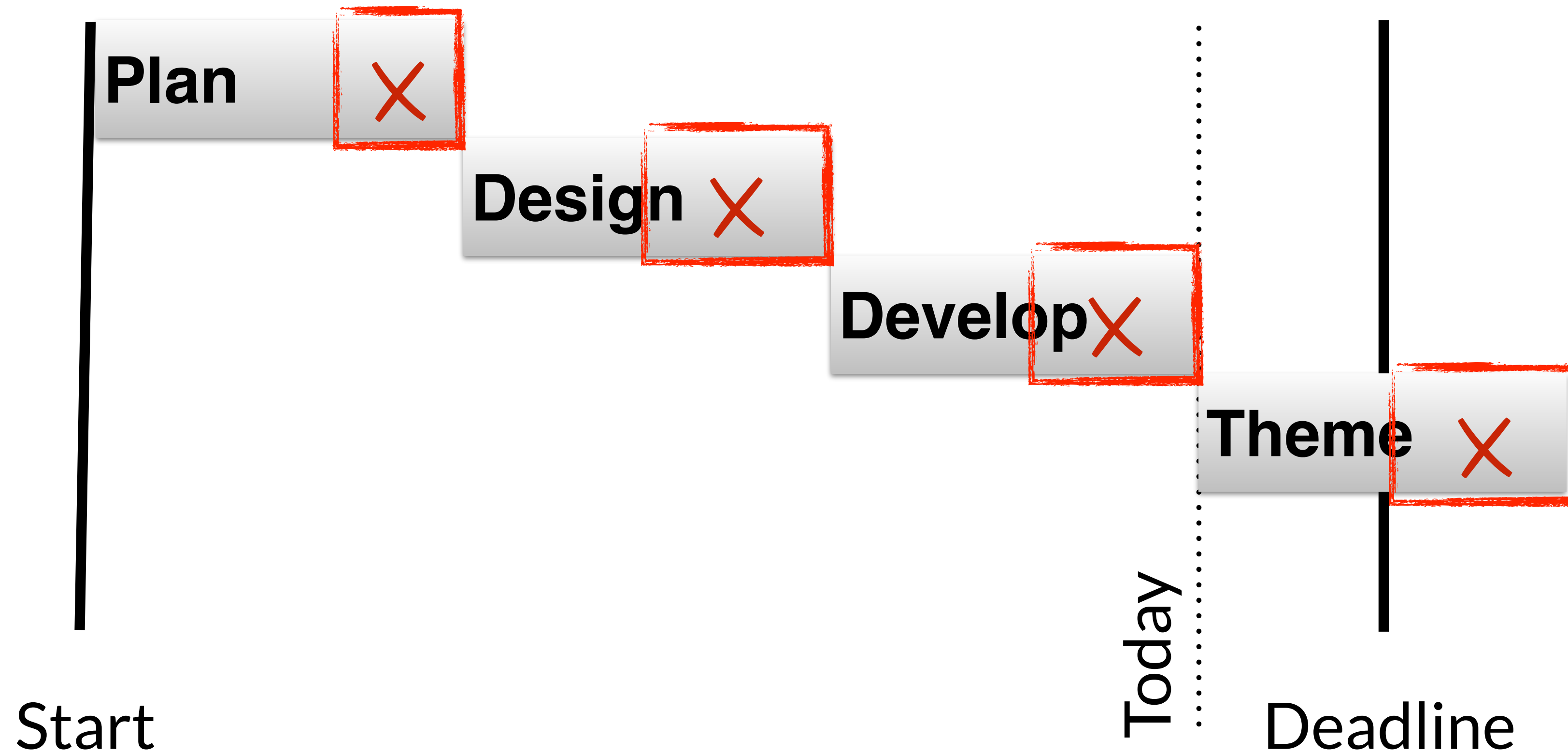
1. Front-end performance (make shit **faster**)
2. Components (make shit **modular**)
3. Continuous Integration (**automate** shit)

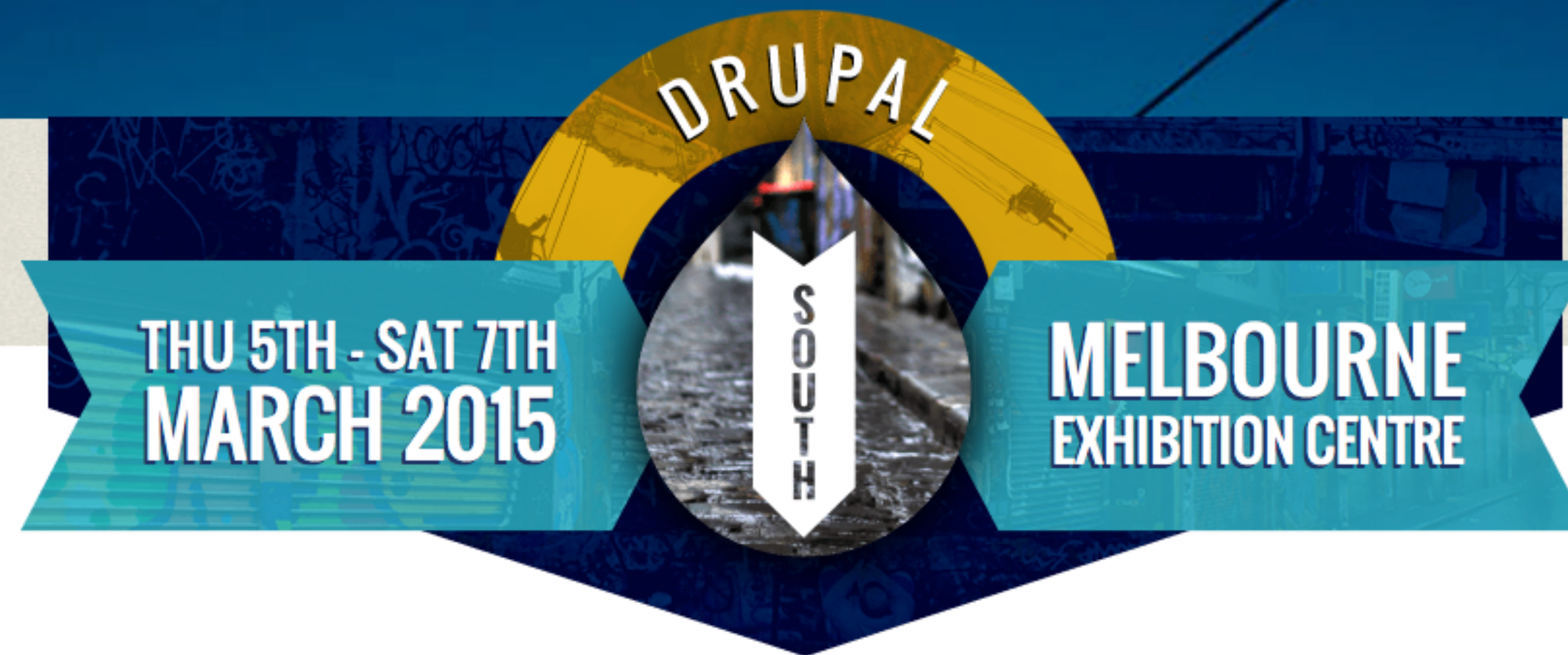


WATERFALL PROJECT



WATERFALL PROJECT

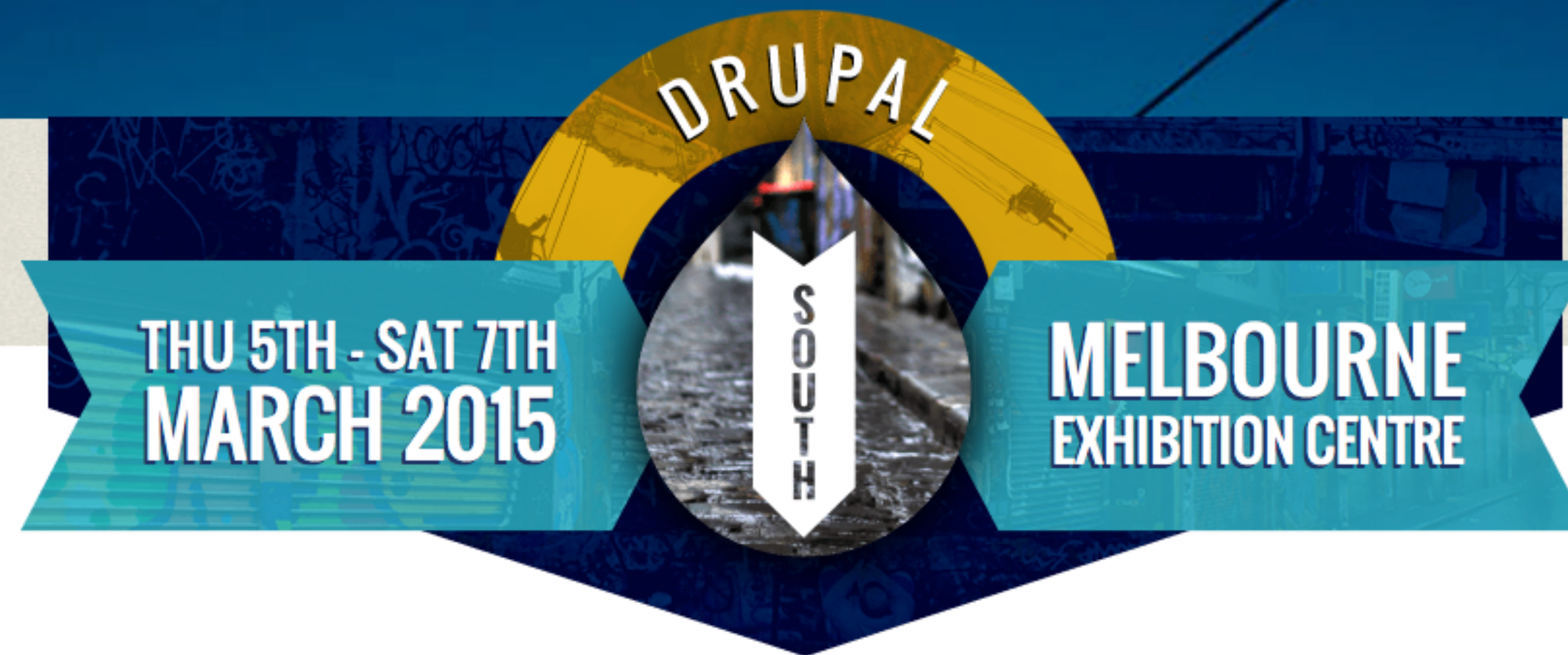




AGILE DEVELOPMENT

PLATINUM SPONSOR

PREVIOUSNEXT®



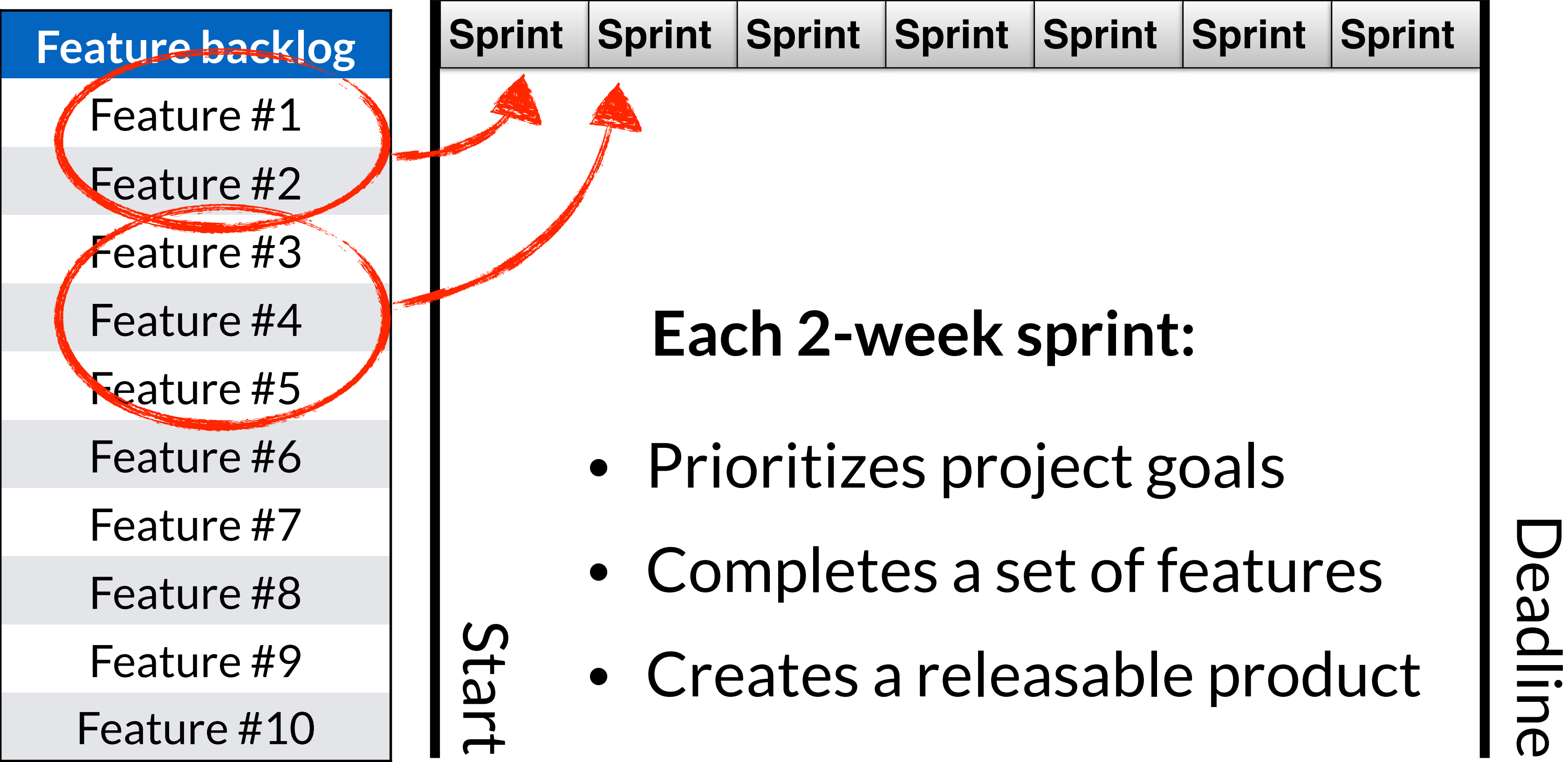
REDUCE YOUR RISK

by controlling/minimizing your risk

PLATINUM SPONSOR

PREVIOUSNEXT®

AGILE PROJECT



AGILE + WEB = ?



STYLE-GUIDE-DRIVEN DEVELOPMENT

Only requirements are:

Component-based design
and
Automated style guides



WHAT ARE WE DOING WRONG?

DRUPAL 7'S CSS



SEEMED LIKE A GOOD IDEA AT THE TIME



WHAT IS A DESIGN COMPONENT?

“Component” is the same as...

“Object” in OOCSS

“Module” in SMACSS

“Block” in BEM’s Block-Element-Modifier

“UI Pattern”



CSS DESIGN COMPONENTS ARE:

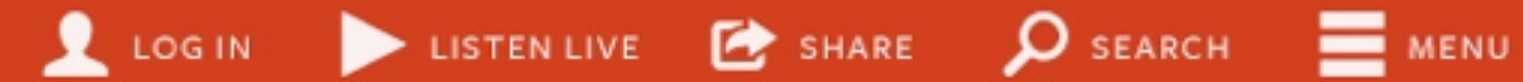
- **Applied to a loose collection of HTML elements**
- **Repeatable**
(even if never repeated)
- **Specific**
Replace CSS specificity with specific names
- **Self contained**
Styles do not bleed onto anything else
- **Nest-able**



Optional 970 x 90 Ad

Sharing Unique Perspectives on Global Issues

Powered by  PRI's The World



Politics & Society



84

How Leaks about NSA Surveillance Affect America's Role Around the Globe

 PRI's The World | June 10, 2013

The hot topic for President Obama's summit this Friday with the Chinese President Xi Jinping was supposed to be over China's efforts at cyber espionage.

Now there's another hot cyber-security topic to add to the agenda. Revelations this week about the US government's ongoing surveillance ...

ADVERTISEMENT

Artists you love. Music they love.

LEARN MORE



World News



969

NSA whistleblower reveals himself and hopes for safety in Hong Kong

The Takeaway | June 09, 2013

World News



969

Lebanon's Hezbollah Militia Celebrates Its Role in Syria's Civil War

 PRI's The World | June 07, 2013

SMACSS

1.**BASE**

2.**LAYOUT**

3.**MODULE**

4.**STATE**

5.**THEME**



SMACSS

1.**BASE**

2.**LAYOUT**

3.**COMPONENTS**

4.**STATE**

5.**SKIN**

1.**BASE**

2.**LAYOUT**

3.**MODULE**

4.**STATE**

5.**THEME**





SNACSS

1. **BASE COMPONENTS**
2. **LAYOUT COMPONENTS**
3. **COMPONENTS**
 - 3.1. **COMPONENT**
 - 3.2. **ELEMENT**
 - 3.3. **MODIFIER**
 - 3.4. **STATE**
 - 3.5. **SKIN**

} **BEM**

**“You see, radio is a kind
of a very, very long cat.**



...there is no cat.”
— Albert Einstein



.flower

3.1.COMPONENT

3.2.ELEMENT

3.3.MODIFIER

3.4.STATE

3.5.SKIN



`.flower__petals`

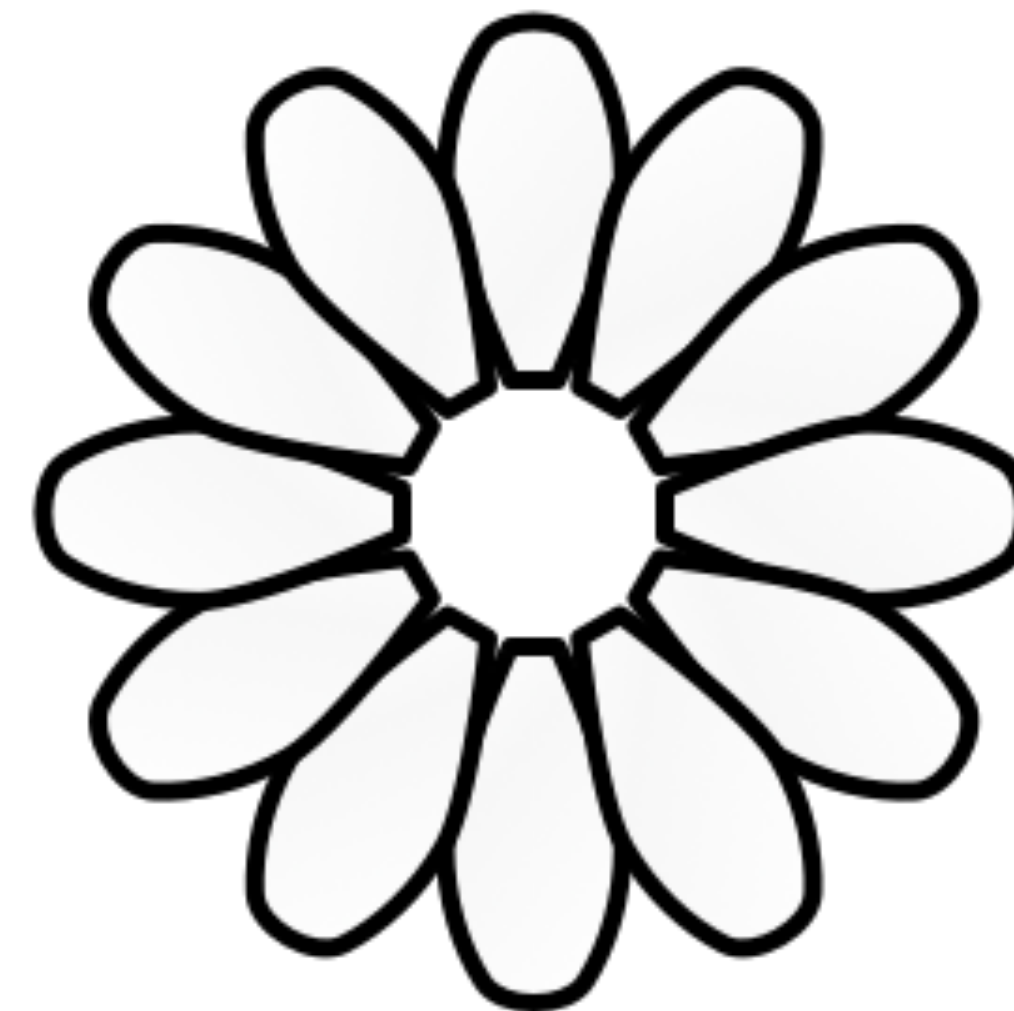
3.1.COMPONENT

3.2.**ELEMENT**

3.3.MODIFIER

3.4.STATE

3.5.SKIN



.flower__face



.flower__stem



.flower__leaves



`.flower__bed`

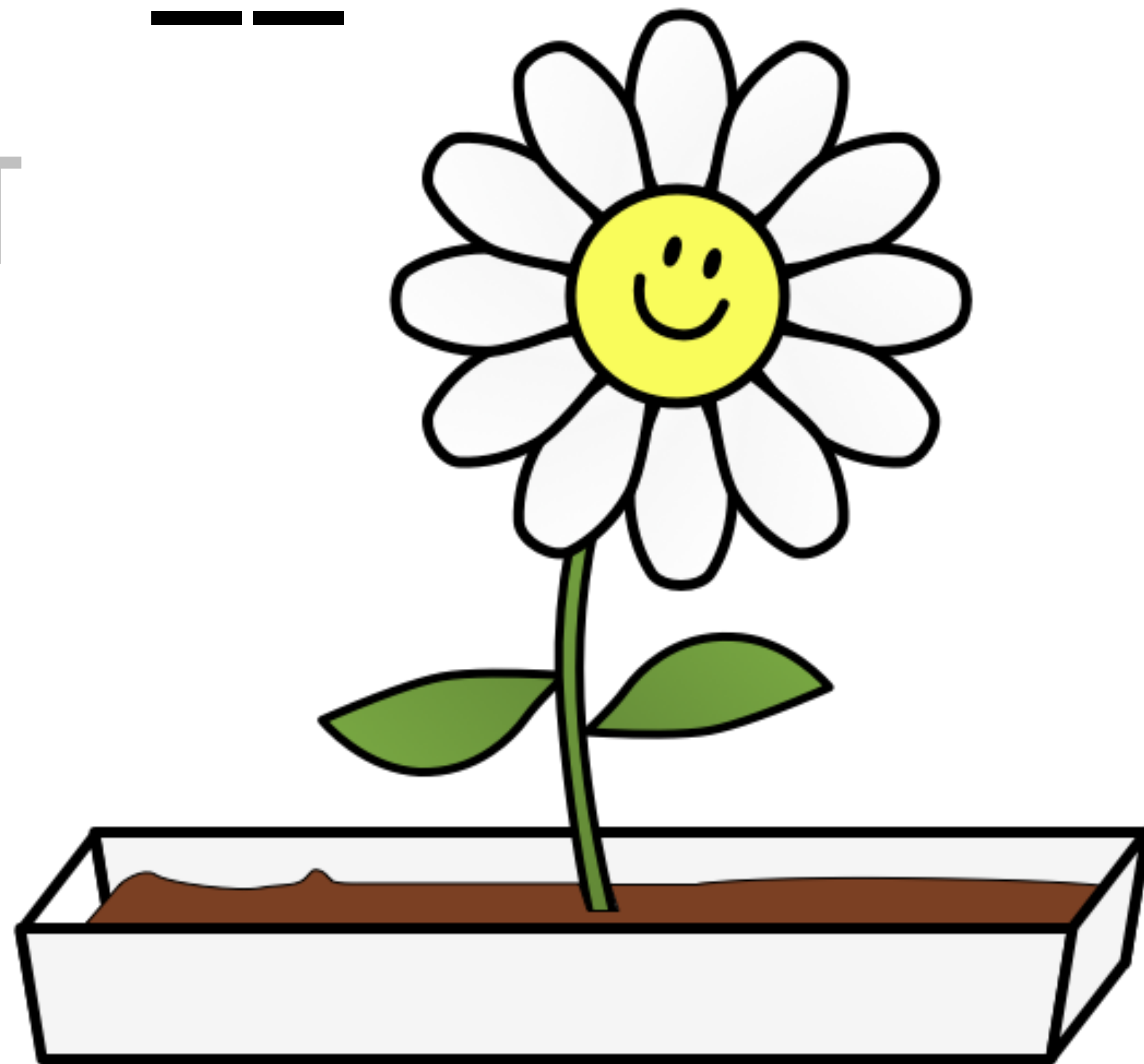
3.1.COMPONENT

3.2.ELEMENT

3.3.MODIFIER

3.4.STATE

3.5.SKIN



.flower--tulip

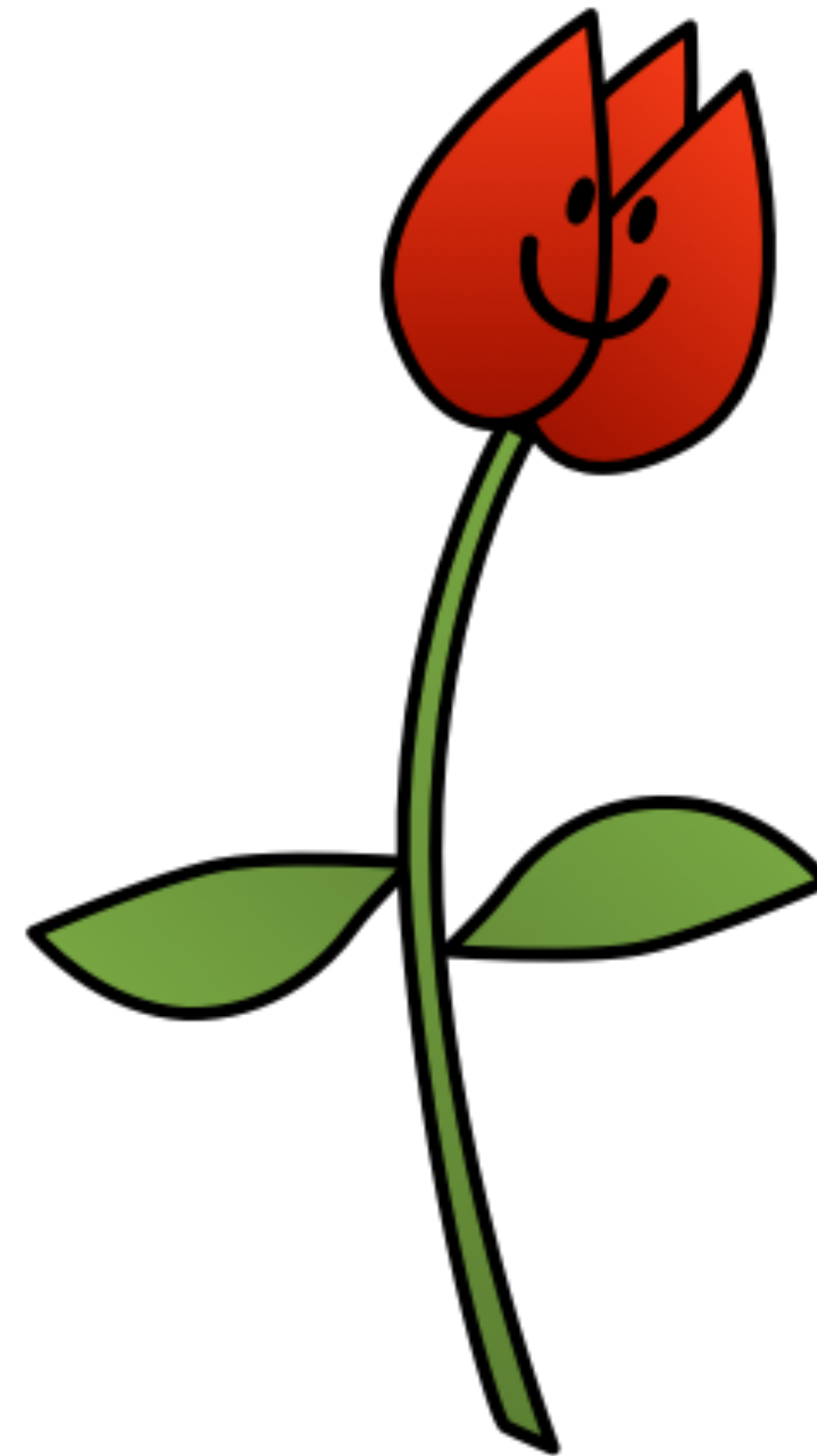
3.1.COMPONENT

3.2.ELEMENT

3.3.**MODIFIER**

3.4.STATE

3.5.SKIN



DON'T MAKE IT COMPLICATED

.channel-tab__guide__upcoming-video__info__time



SUCKING AT SOMETHING

**IS THE FIRST STEP TO BECOMING SORTA GOOD
AT SOMETHING**



ON THE MEANING OF SEMANTICS

- Content semantics
are handled by HTML5 elements
- Let's make our class names use
Design semantics
- Class names should be meaningful to
developers and designers



.flower: hover



3.1.COMPONENT

3.2.ELEMENT

3.3.MODIFIER

3.4.**STATE**

3.5.SKIN



.flower.is-pollinating

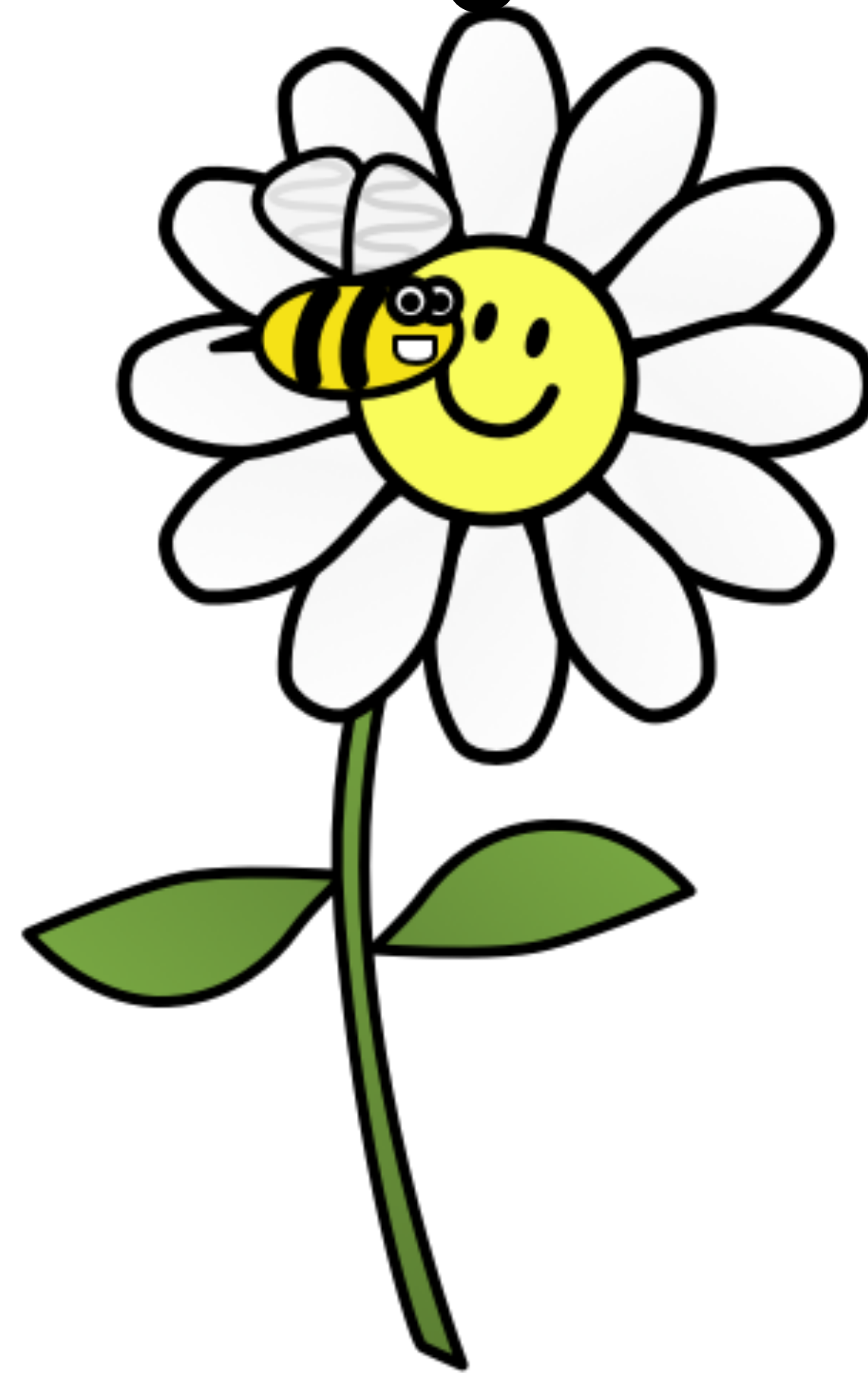
3.1.COMPONENT

3.2.ELEMENT

3.3.MODIFIER

3.4.**STATE**

3.5.SKIN



@media (min-width: 48em) { .flower }

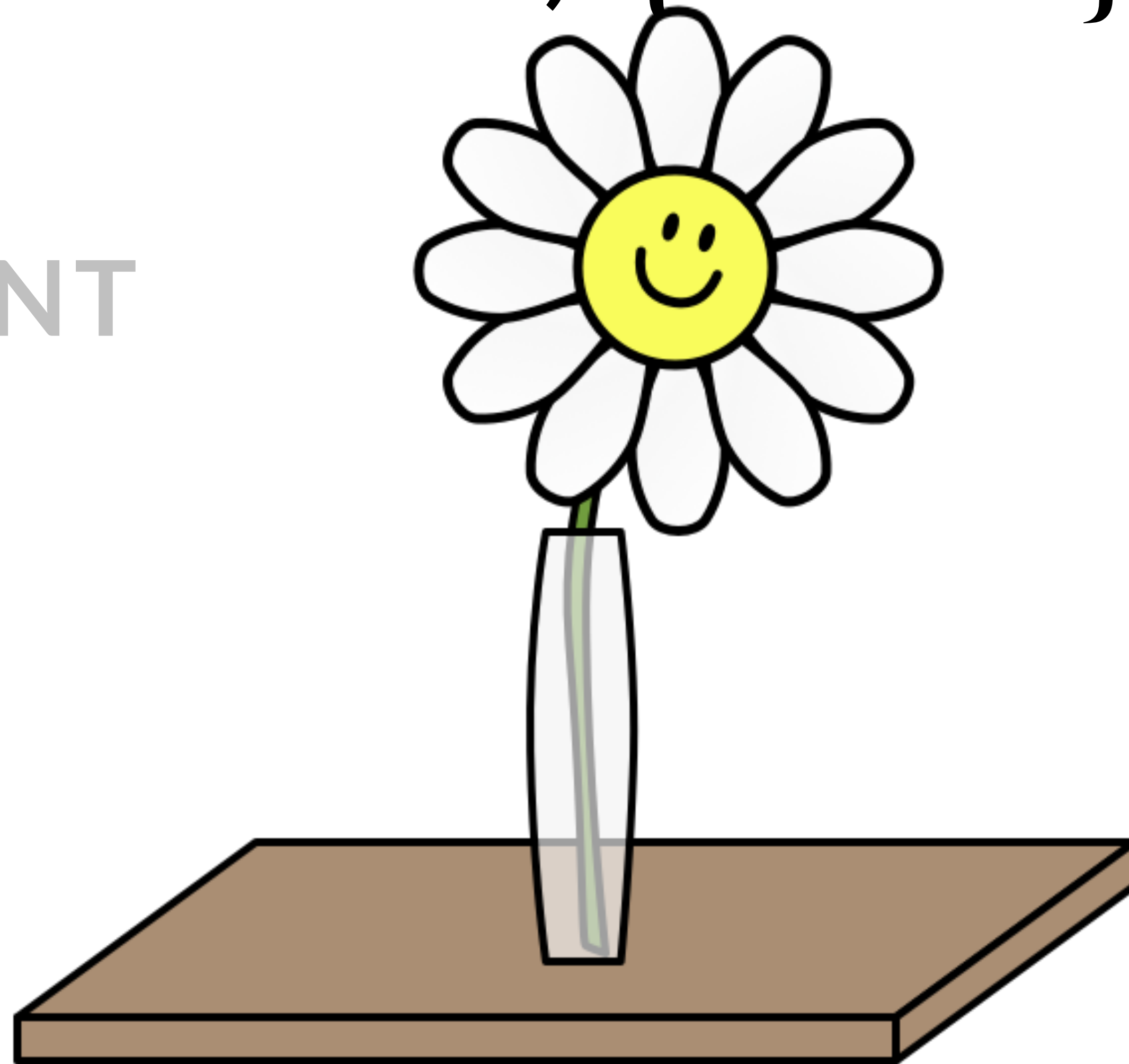
3.1.COMPONENT

3.2.ELEMENT

3.3.MODIFIER

3.4.**STATE**

3.5.SKIN



@media print { .flower }

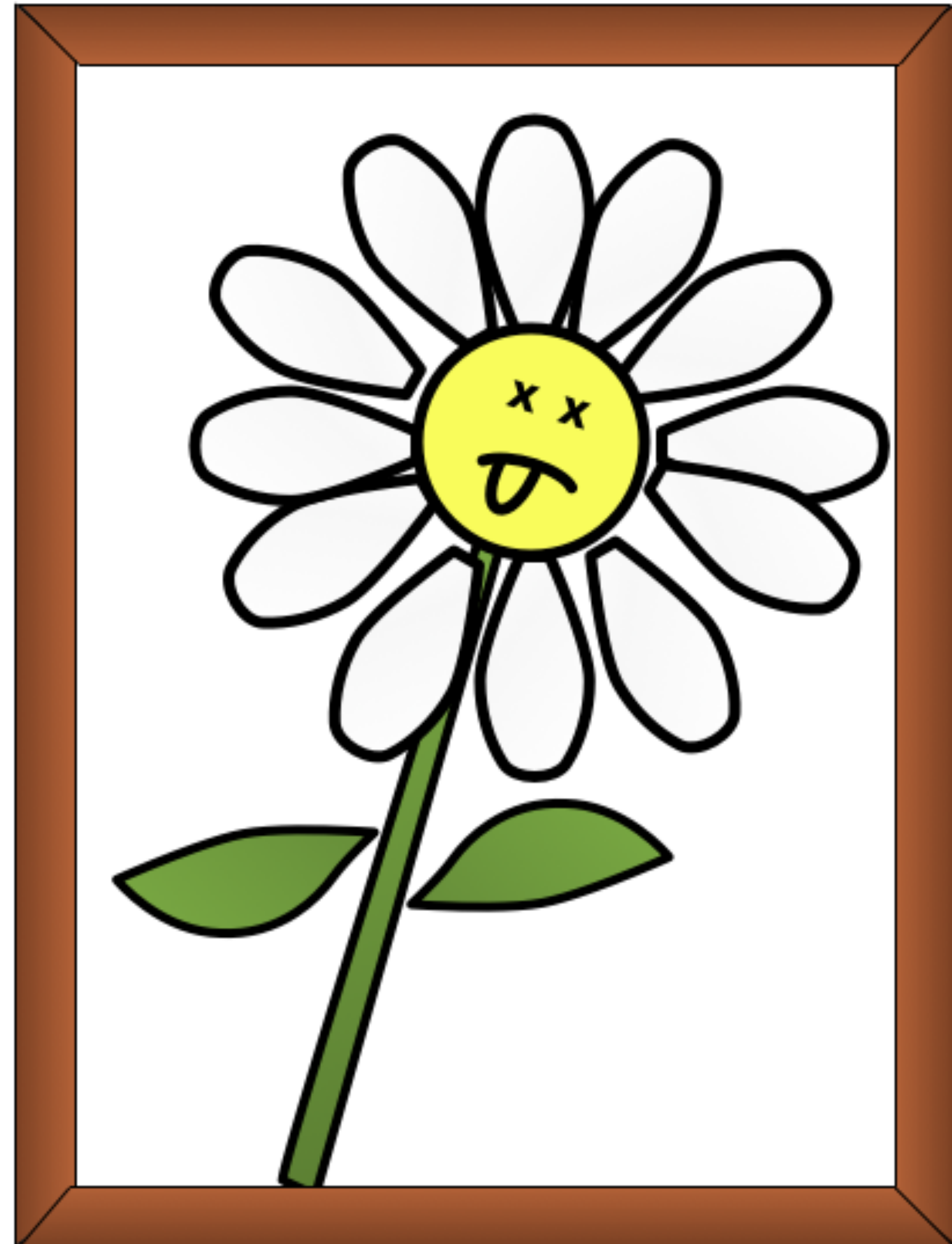
3.1.COMPONENT

3.2.ELEMENT

3.3.MODIFIER

3.4.**STATE**

3.5.SKIN



.is-night .flower

3.1.COMPONENT

3.2.ELEMENT

3.3.MODIFIER

3.4.STATE

3.5.**SKIN** (global modifier)



.flower

An automated
style guide of
.flower is available:

johnalbin.github.io/flower-power



All the selectors

`.the-component`

`.the-component--modifier`

`.the-component__an-element`

`.the-component--modifier__an-element`

`.the-component.is-state`

`.the-component: hover`

`@media all { .the-component {} }`

`.the-skin .the-component`



CSS DESIGN COMPONENTS ARE:

- **Applied to a loose collection of HTML elements**
- **Repeatable**
(even if never repeated)
- **Specific**
Replace CSS specificity with specific names
- **Self contained**
Styles do not bleed onto anything else
- **Nest-able**





SNACSS

1. **BASE COMPONENTS**
2. **LAYOUT COMPONENTS**
3. **COMPONENTS**
 - 3.1. **COMPONENT**
 - 3.2. **ELEMENT**
 - 3.3. **MODIFIER**
 - 3.4. **STATE**
 - 3.5. **SKIN**

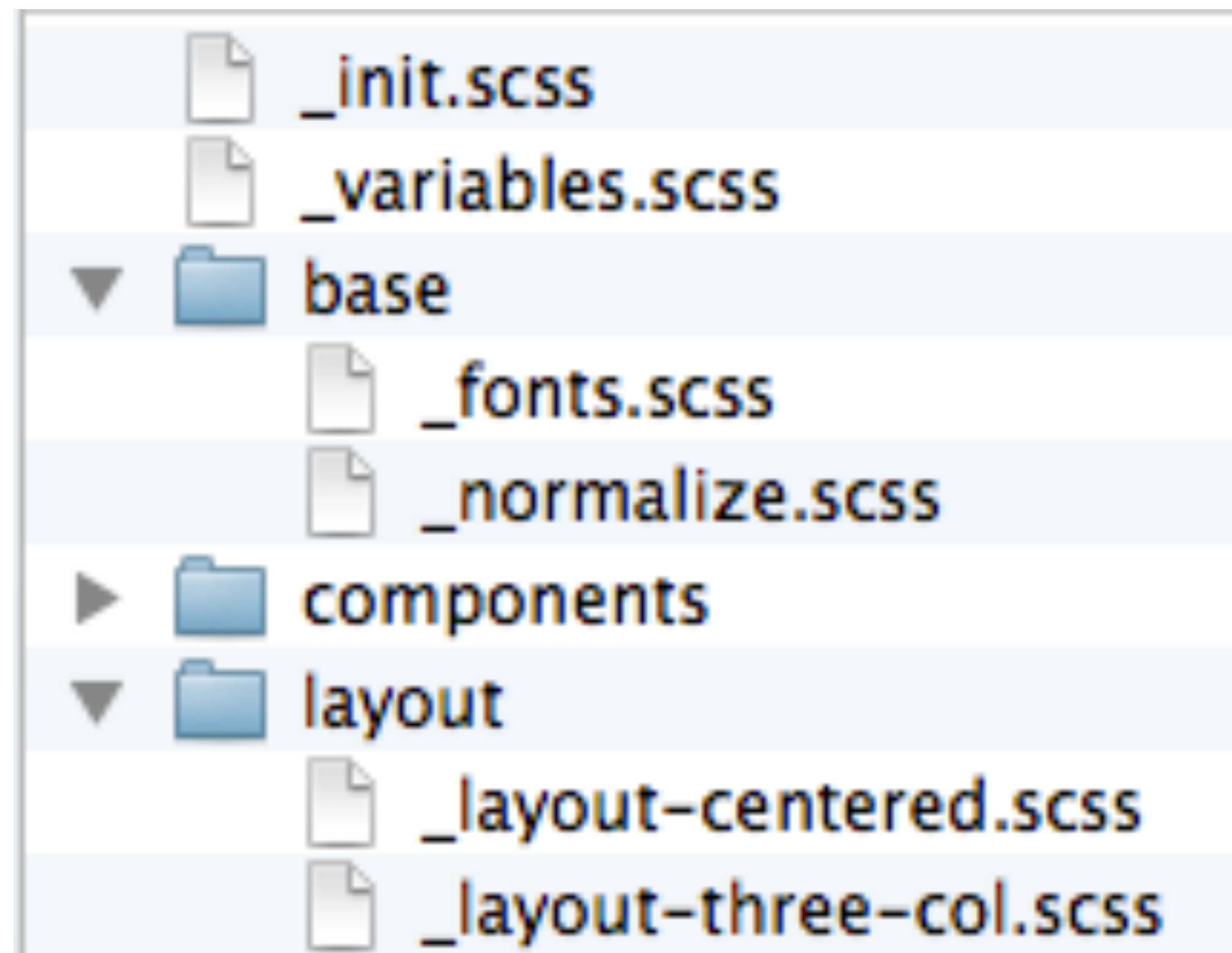
} **BEM**

DRUPAL 8 CSS CODING STANDARDS

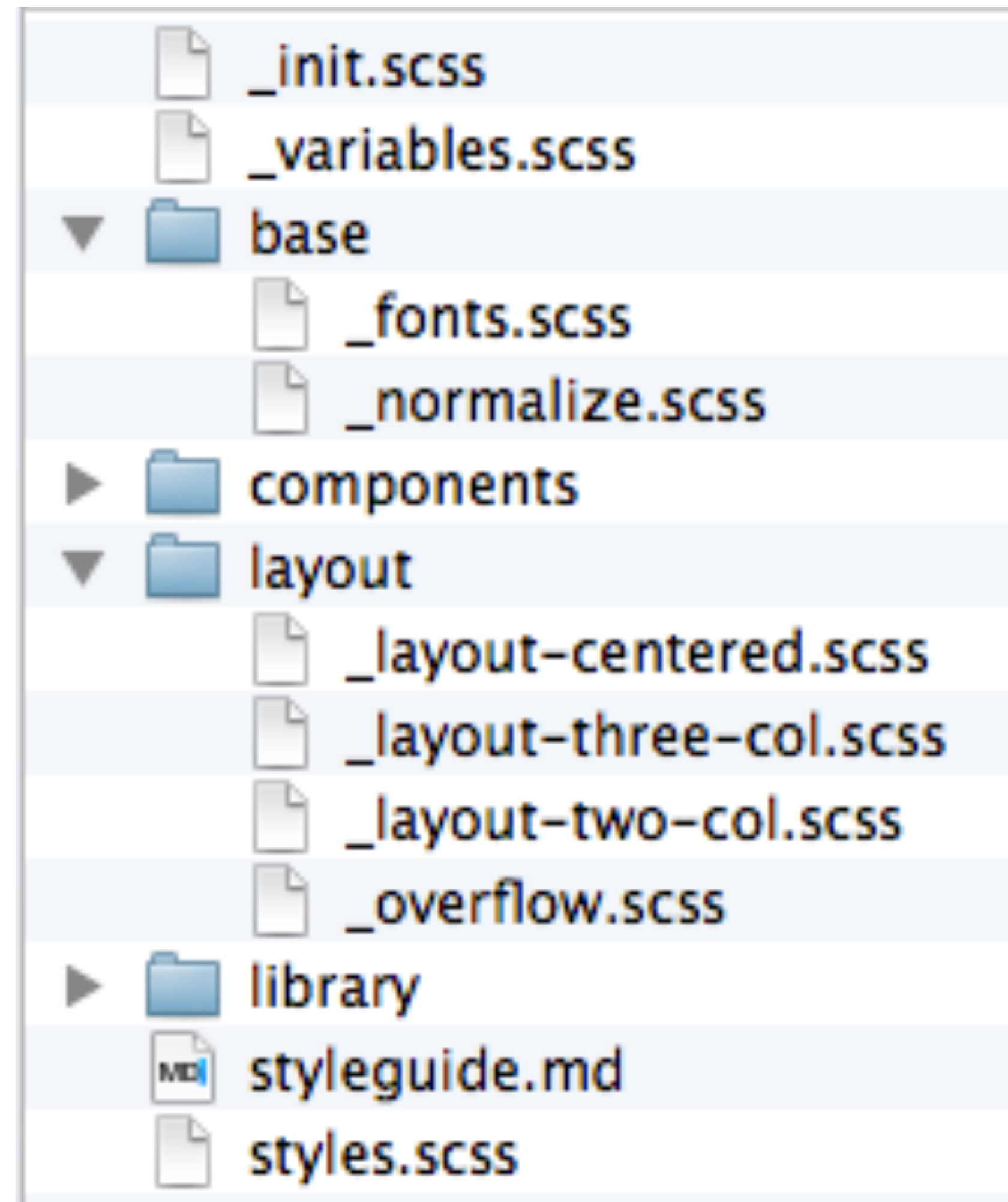
www.drupal.org/node/1886770



FILE ORGANIZATION = 3 SMACSS CATEGORIES



SIMPLE SASS SETUP



STYLES.SCSS

```
/* Import Sass mixins, variables, Compass modules, etc. */
@import "init";

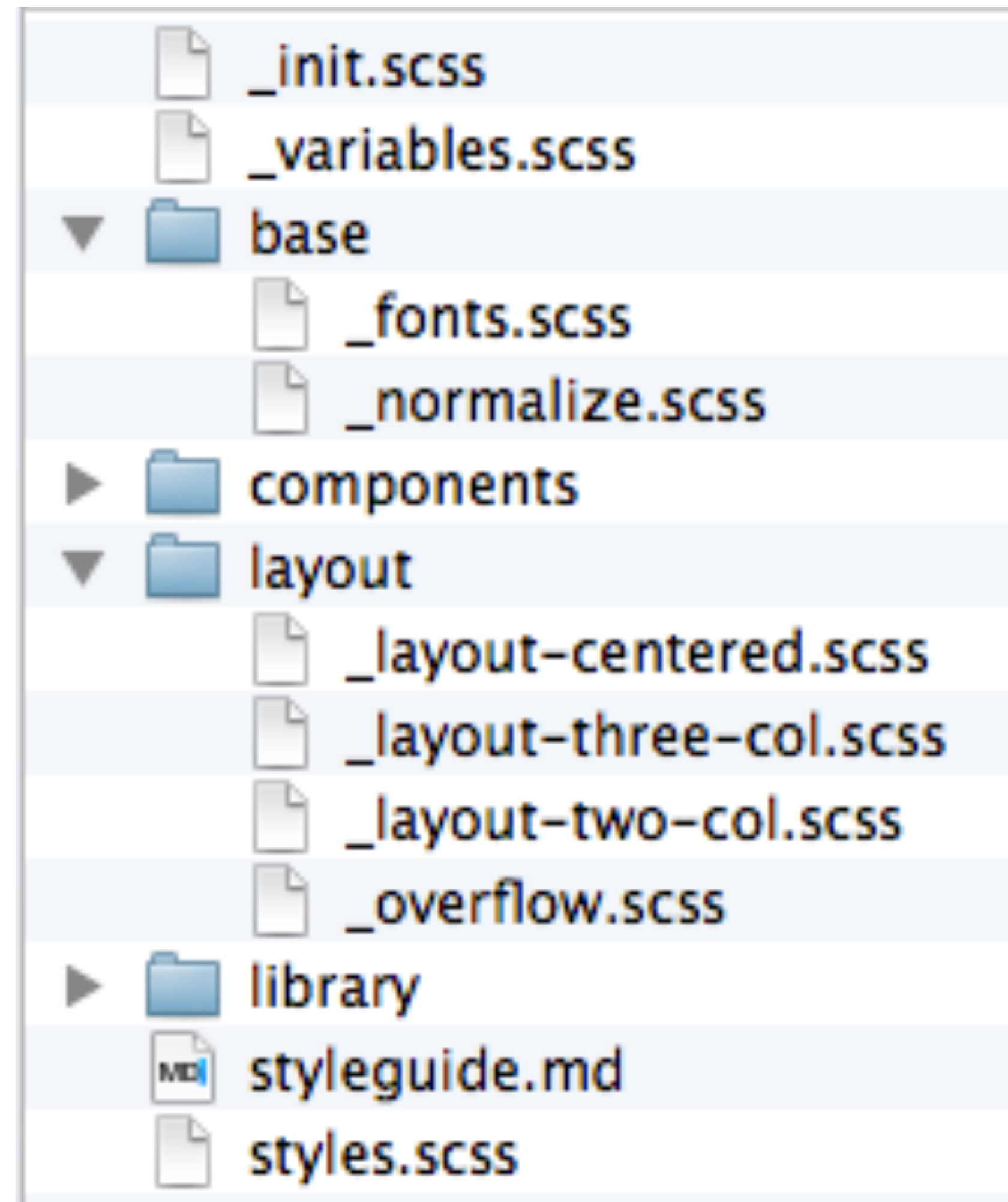
/* HTML element (SMACSS base) rules */
@import "base/normalize";

/* Layout rules */
@import "layouts/l-centered";
@import "layouts/l-general";
@import "layouts/l-header";
@import "layouts/l-home";
@import "layouts/l-story";
@import "layouts/l-vertical";

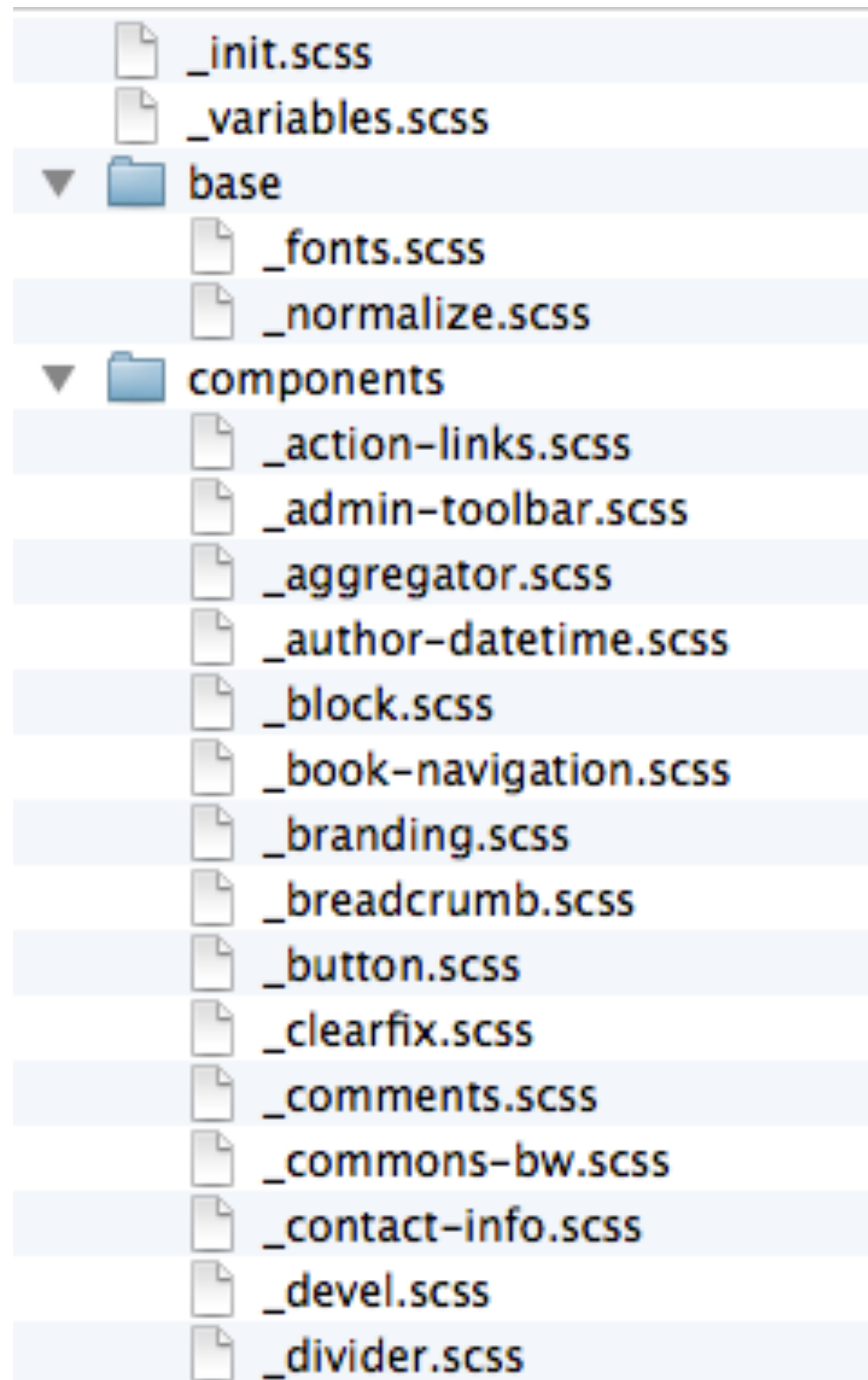
/* Component (SMACSS module) rules */
@import "components/clearfix";
@import "components/visually-hidden";
@import "components/button";
@import "components/engagement-badge";
@import "components/feature-large";
```



SIMPLE FOLDER STRUCTURE



EASY TO FIND YOUR COMPONENTS.



- **Inspect the DOM.**
- **Find the CSS class on the component.**
- **Look for a file with that name in the components folder.**



THE “FUGLY” SELECTOR HACK



THE “FUGLY” SELECTOR HACK

```
.feature__title a {  
  &:link,  
  &:visited {  
    @extend %feature__title-link;  
  }  
  &:hover,  
  &:focus {  
    @extend %feature__title-link-is-hover;  
  }  
}
```

Selector in DOM
I couldn't change

Class name I wish
I could use in DOM



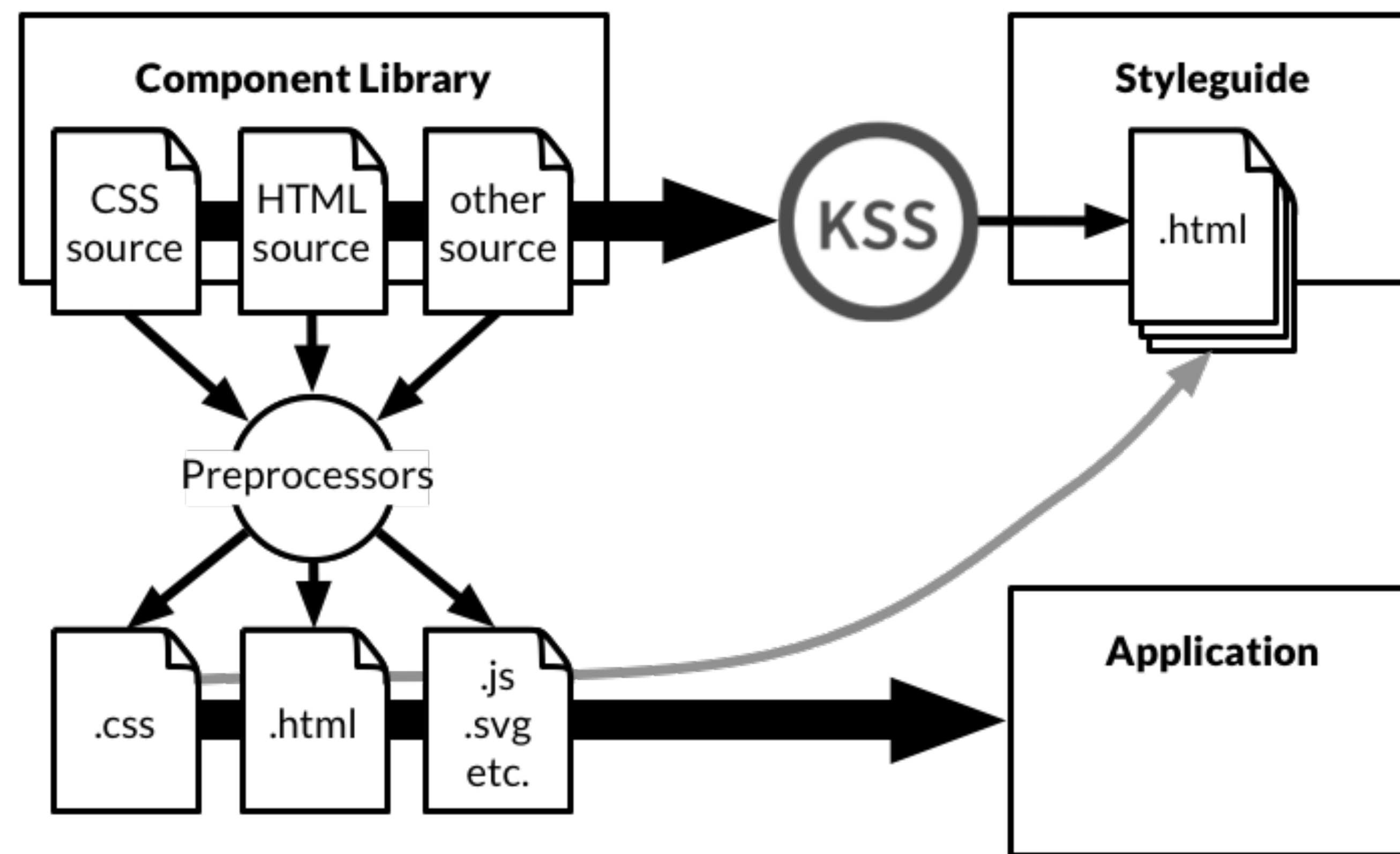
AUTOMATED STYLE GUIDE

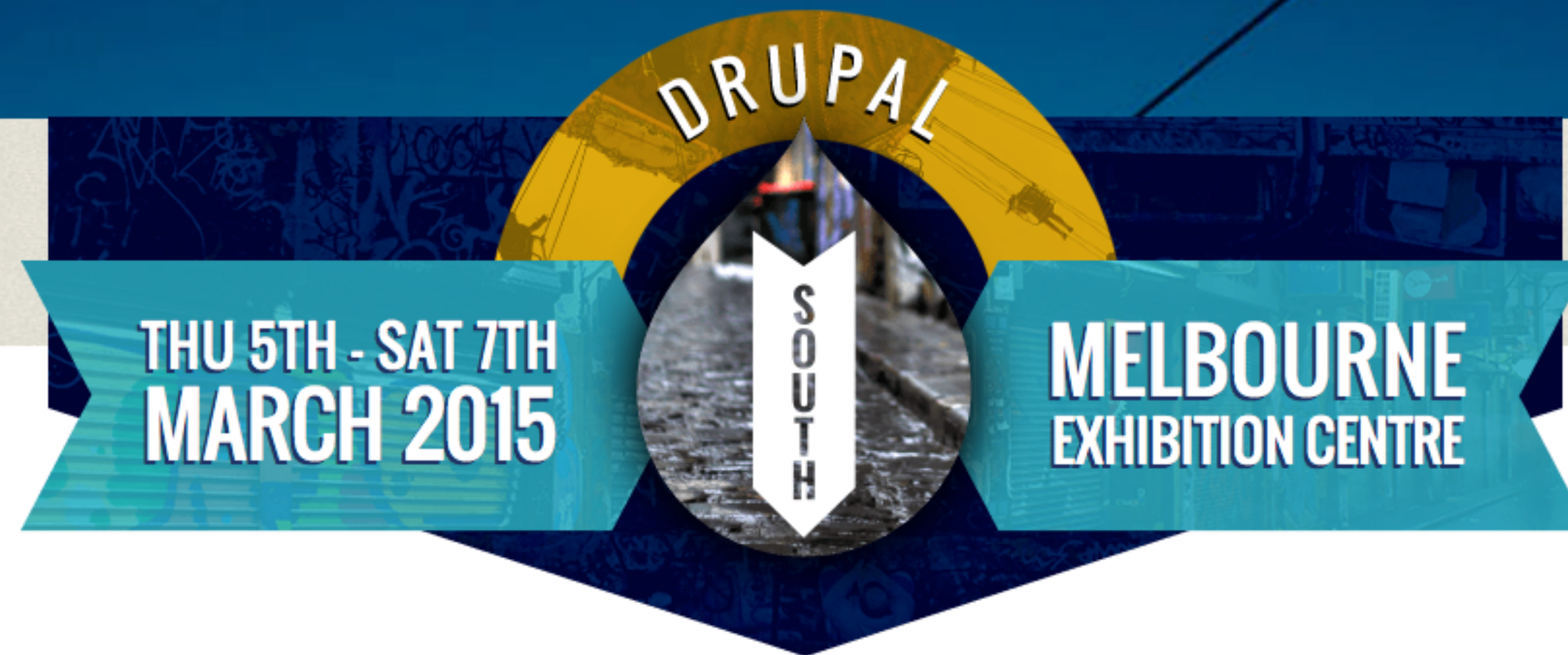


kss-node
github.com/kss-node/kss-node



AUTOMATED STYLE GUIDE





THANK YOU!

slides available at: (will post URL on Twitter)

PLATINUM SPONSOR

PREVIOUSNEXT®