Building Drupal Sites

with Components

DEREK “tachyon” REESE

Senior Drupal Developer

I makes tools for artists to tell compelling stories with technology.

I have been working with Drupal since Drupal 5.

- @derekreese
- /in/derekreese
- slideshare.net/derekreese

About

Mediacurrent helps organizations build highly impactful, elegantly designed Drupal websites that achieve the strategic results they need.

- Single-source provider
- Specializing in Drupal since 2007
- Headquartered in Atlanta, GA
- Team of 60+ Drupal Experts including development, design and strategy
- Clients include: Large Enterprise and high-profile global brands

Agenda

1. A Brief Overview of Components
2. Introducing the Problem
3. Finding the Solution
4. Building it in the Real World
5. The Future of Components

1 A Brief Overview of Components

A Crash Course in Visual Language

“Digital media are any media that are encoded in a machine-readable format.”

Digital Media

Unlimited Number of forms

Interpreted in a Large Amount of Ways

Credit: David Hiser, 1937; U.S. National Archives and Records Administration; Public Domain

“Design is purpose, planning, or intention that exists or is thought to exist behind an action, fact, or material object.”

Visual Language

Visual Language for Digital Media Design

Atomic Design

Atomic Design

Brad Frost

Modular Design

->

Atomic Design

Atomic Design

Atoms

tags, content, smallest attributes (colors etc.)

Molecules

Organisms

Templates

Pages

```
<a
href="http://mediacurrent.
com">Atom</a>
```

Atomic Design

Atoms

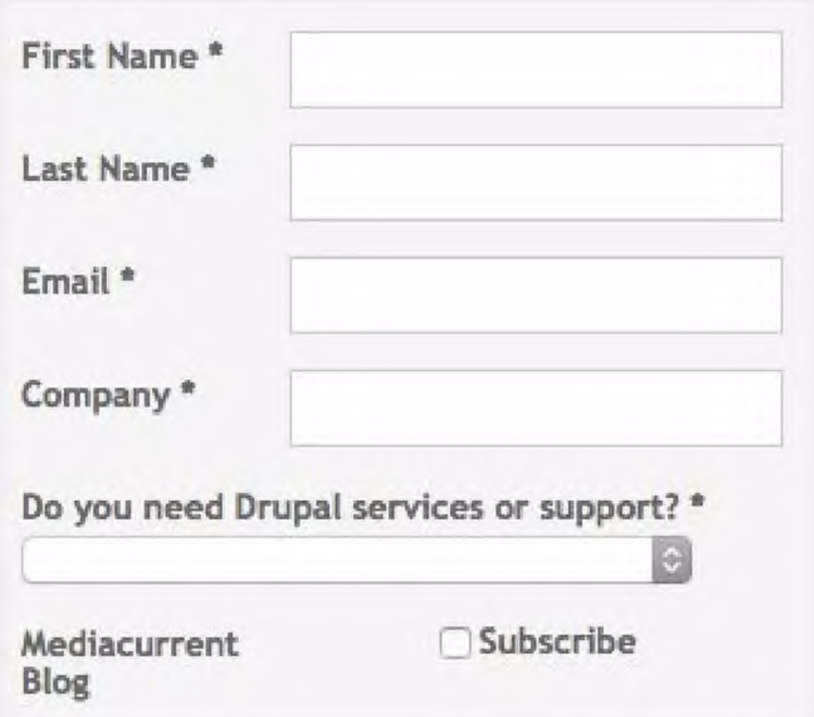
Molecules

forms, paragraphs, menus, media items

Organisms

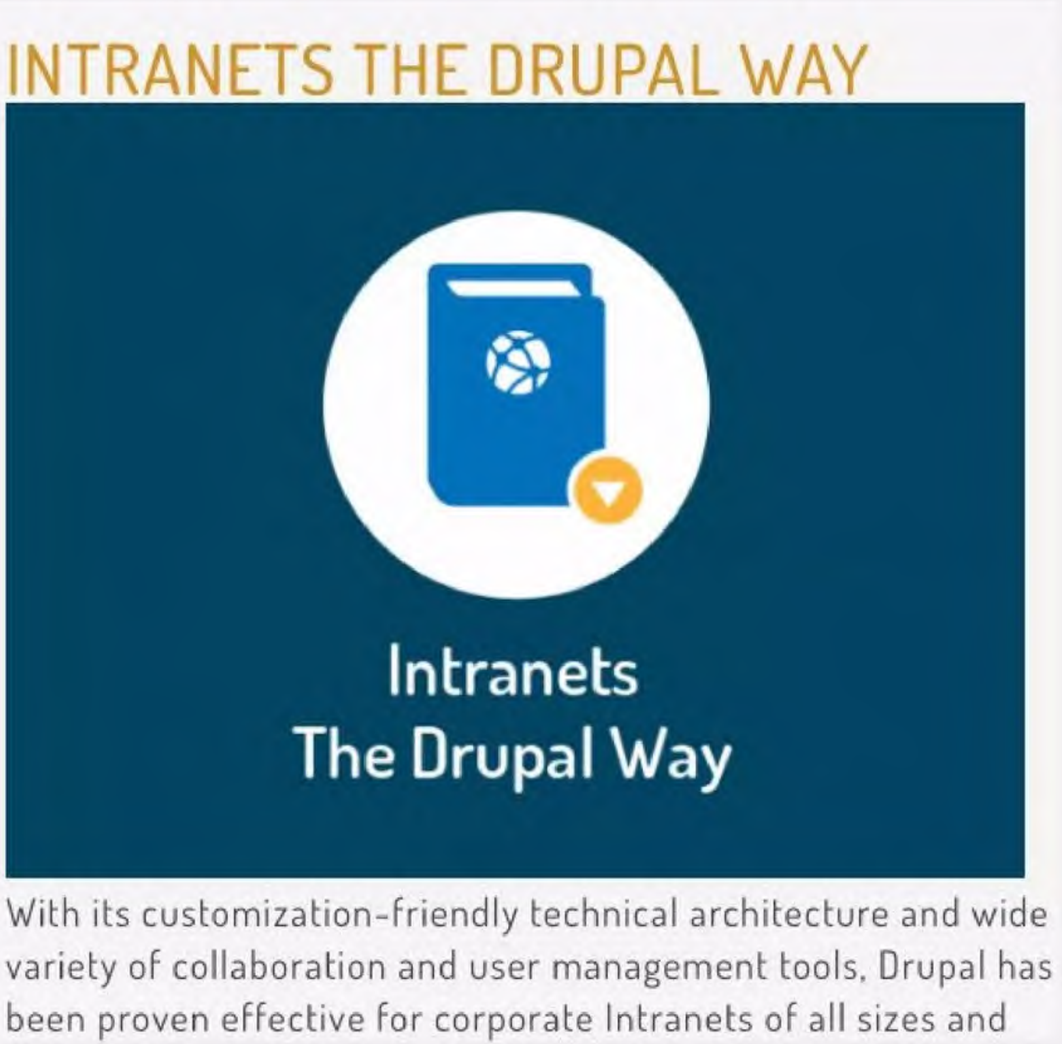
Templates

Pages



Atomic Design

Atoms
Molecules
Organisms
headers, footers, heroes, page bodies
Templates
Pages



Atomic Design

Atoms

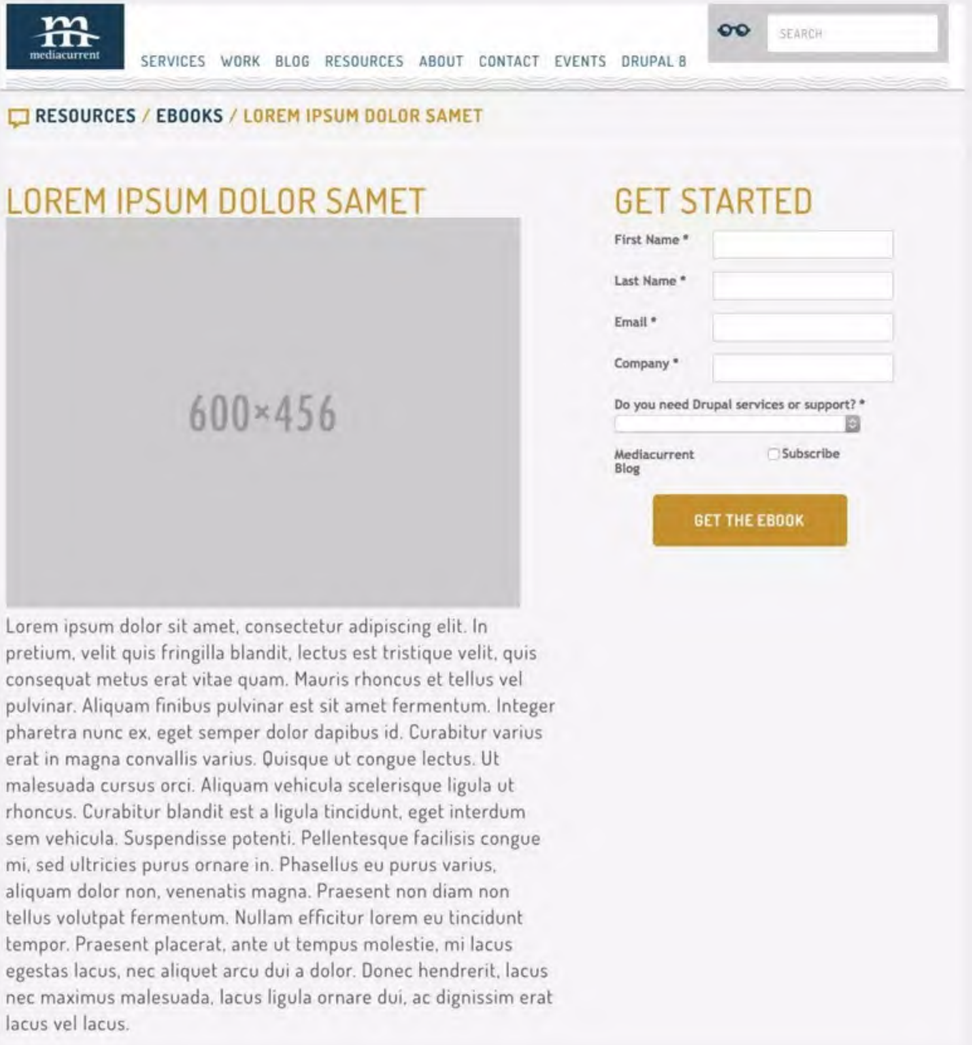
Molecules

Organisms

Templates

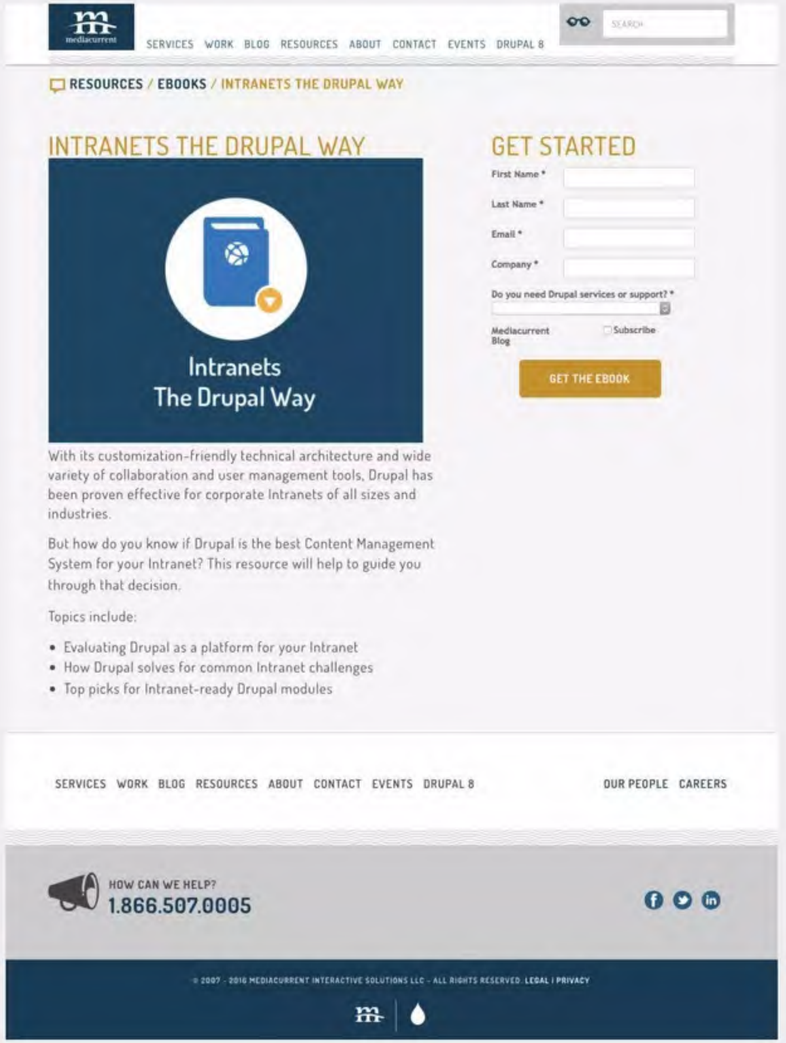
layouts with context

Pages



Atomic Design

Atoms
Molecules
Organisms
Templates
Pages
specific instances of templates with real content



Atomic Design

Atoms
Molecules
Organisms
Templates
Pages

http://atomicdesign.bradfrost.com/

Components

Made of *Atoms*

Correlate to Molecules, Organisms

Media Item

Media Item used is various modules such as Featured Content

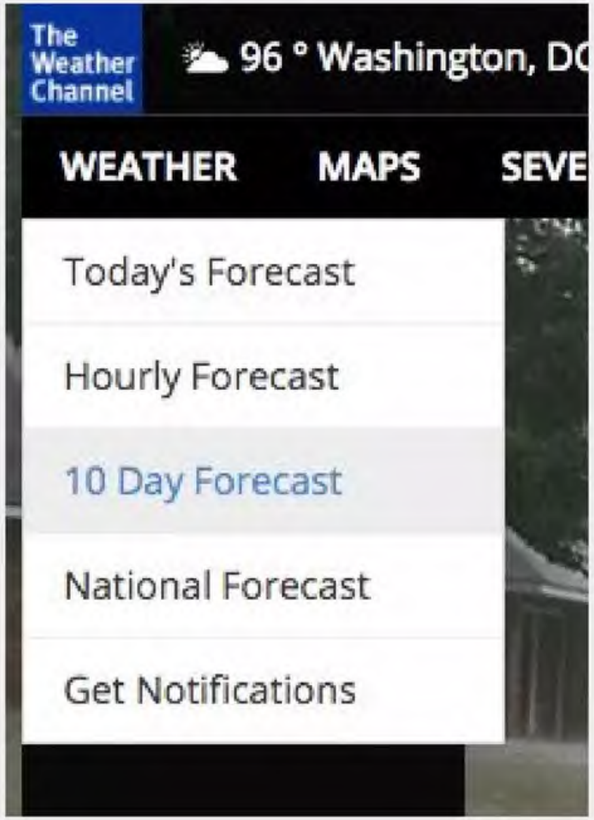
Media item type

Media item title

Components in the World Wide Web

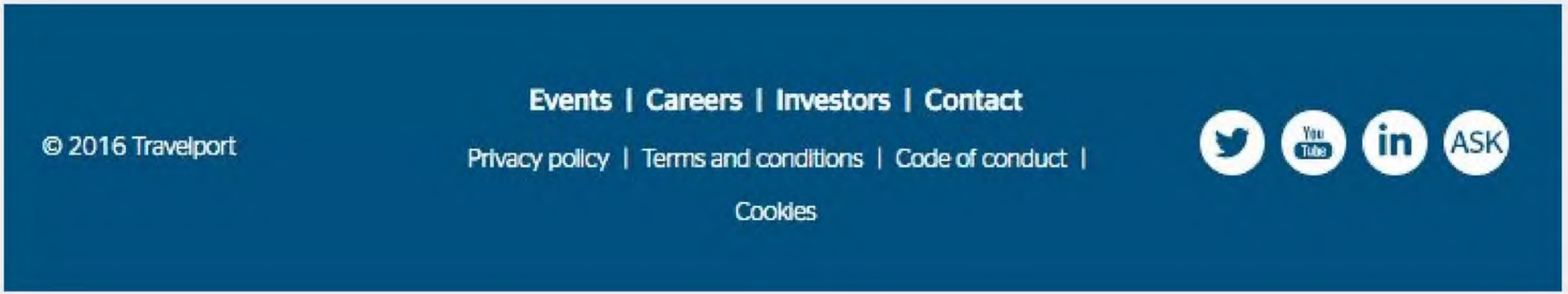
Search Form

Menu


The Weather Channel
96 ° Washington, DC
WEATHER
MAPS
SEVE
Today's Forecast
Hourly Forecast
10 Day Forecast
National Forecast
Get Notifications

Header

Footer


© 2016 Travelport
Events | Careers | Investors | Contact
Privacy policy | Terms and conditions | Code of conduct |
Cookies
You Tube
ASK

Hero

We help organizations build highly impactful, elegantly designed
Drupal websites that achieve the strategic results they need.

Media Item

Ignite Your Passion For Innovation With GPB And Discovery Education

You're invited to a powerful evening of networking and concise and energizing presentations.

August 12, 2016

A Crash Course in Visual Language

Components in Drupal

Fields

Image *

Alternative text *

Lorem ipsum dolor sit amet, consectetur adipiscing

This text will be used by screen readers, search engines, or when the image cannot be loaded.

fun-fact.jpg (52.19 KB) **Remove**

Fact *

Lorem ipsum dolor sit amet, consectetur adipiscing

Blocks / Paragraphs / Menu Blocks

BLOCK	CATEGORY	REGION	OPERATIONS
Header Place block			
Site branding	System	Header	Configure
Primary menu Place block			
Main navigation	Menus	Primary menu	Configure
Highlighted Place block			
Status messages	System	Highlighted	Configure
Content Place block			
Help	Help	Content	Configure
Primary admin actions	core	Content	Configure

Nodes & Node View Modes

FIELD	LABEL	FORMAT	
Hero Media*	Above	Rendered entity	Rendered as Default
Links		Visible	
Image	Above	Image	Original image
Fun Facts	Above	Rendered entity	Rendered as Default

2 Introducing the Problem

Credit: Yoram E. Shamir, 1970's; Public Domain

Credit: Yoram E. Shamir, 1970's; Public Domain

Drupal Site Build with Components

Leads to duplicate work - matching Drupal theme templates to components

Or

Spending time writing CSS to visually match Drupal output to components

Drupal (Content Model) Components vs Design (Semantic) Components

.field--quote-text

```
<div class="block
block--quote">
  <div class="field
field--quote-text">Lorem ipsum
dolor samet.</div>
  <div class="field
field--quote-author">Gratis
verga</div>
</div>
```

.quote__text

```
<div class="quote">
  <div
class="quote__text">Lorem ipsum
dolor samet.</div>
  <div
class="quote__author">Gratis
verga</div>
</div>
```

Translating Data to Design

How to get Drupal Content into HTML/CSS/JS Atomic Components?

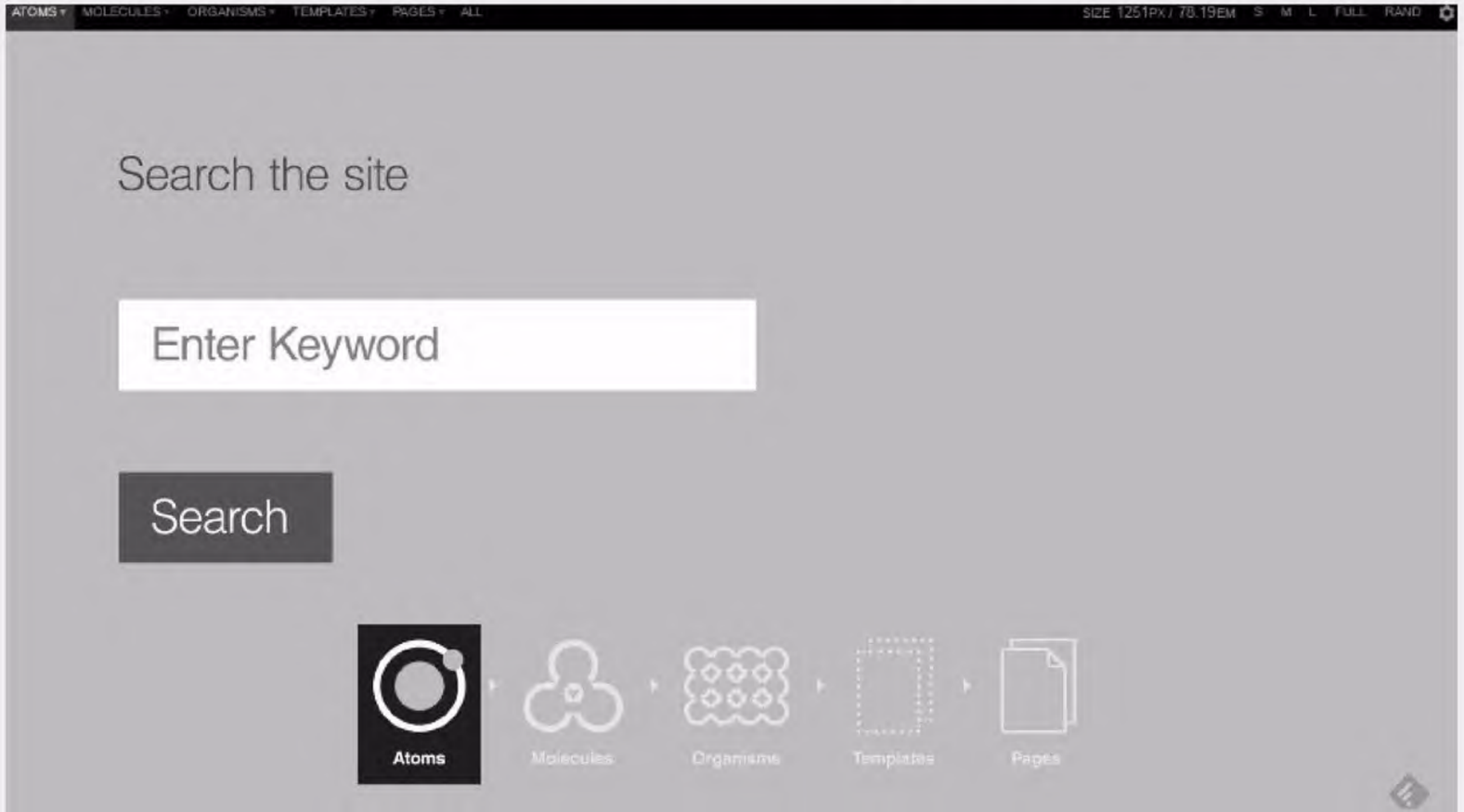
3 Finding the Solution

Creating Design Components

Start by working backwards

Pattern Lab

Dave Olsen and Brian Muenzenmeyer, with occasional input from Brad Frost



KSS

Kyle Kneath @kneath

Knyle Style Sheets

Documentation for CSS (SASS/SCSS etc.)

http://warpspire.com/kss/

```
// Media Item
//
// Media Item is used is various modules.
//
// Markup: components.media-item.twig
//
// Style guide: components.media-item

.media-item {
 max-width: 550px;
 position: relative;
}
```

KSS Node

John Albin

https://github.com/kss-node/kss-node

Kalastatic - Kalamuna

https://github.com/kalamuna/kalastatic

Drupal Content to KSS/Pattern Lab Components

Creating a Translation Layer

- Needs to work with Twig Components
- Needs to work with Drupal
- Needs to fit in project budgets & timelines

Drupal Content to KSS/Pattern Lab Components

Drupal Solution (out of the box with D8)

Twig Includes

Gives control to the design

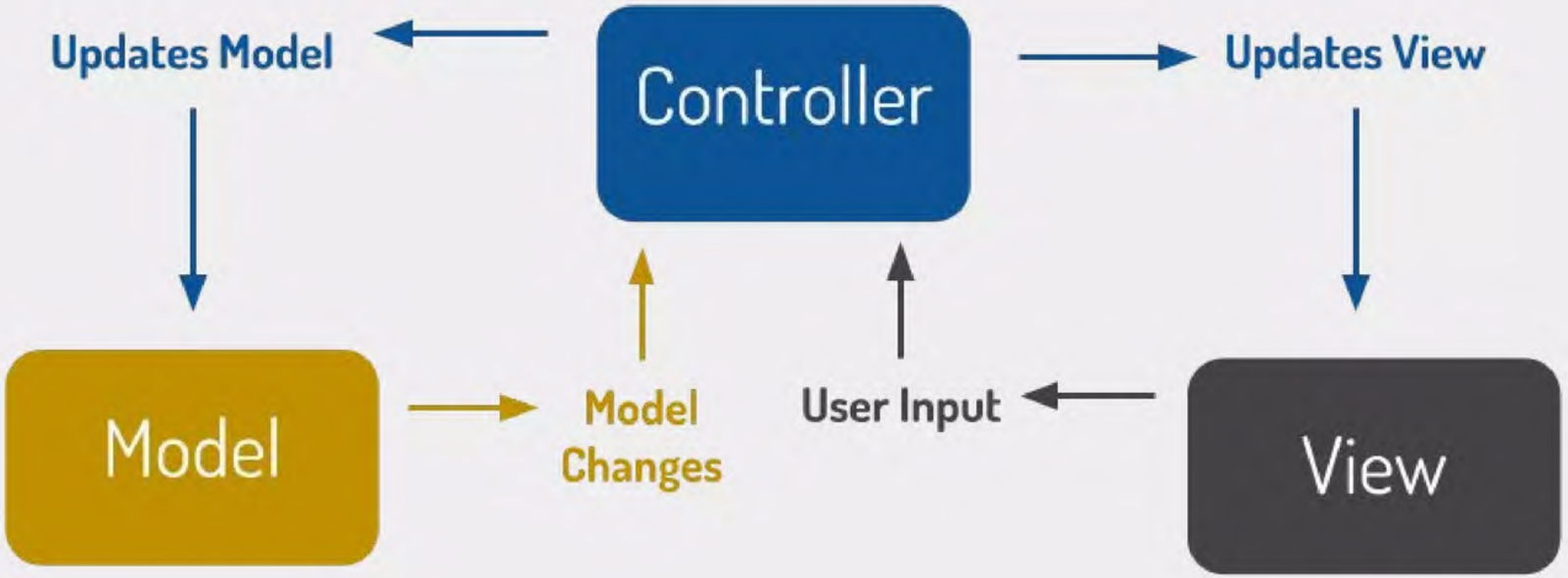
Copyright Sensio Labs

Synchronicities

Drupalcon New Orleans
Micah Godbolt & Redhat Teams
John Albin
Mediacurrent

MVP

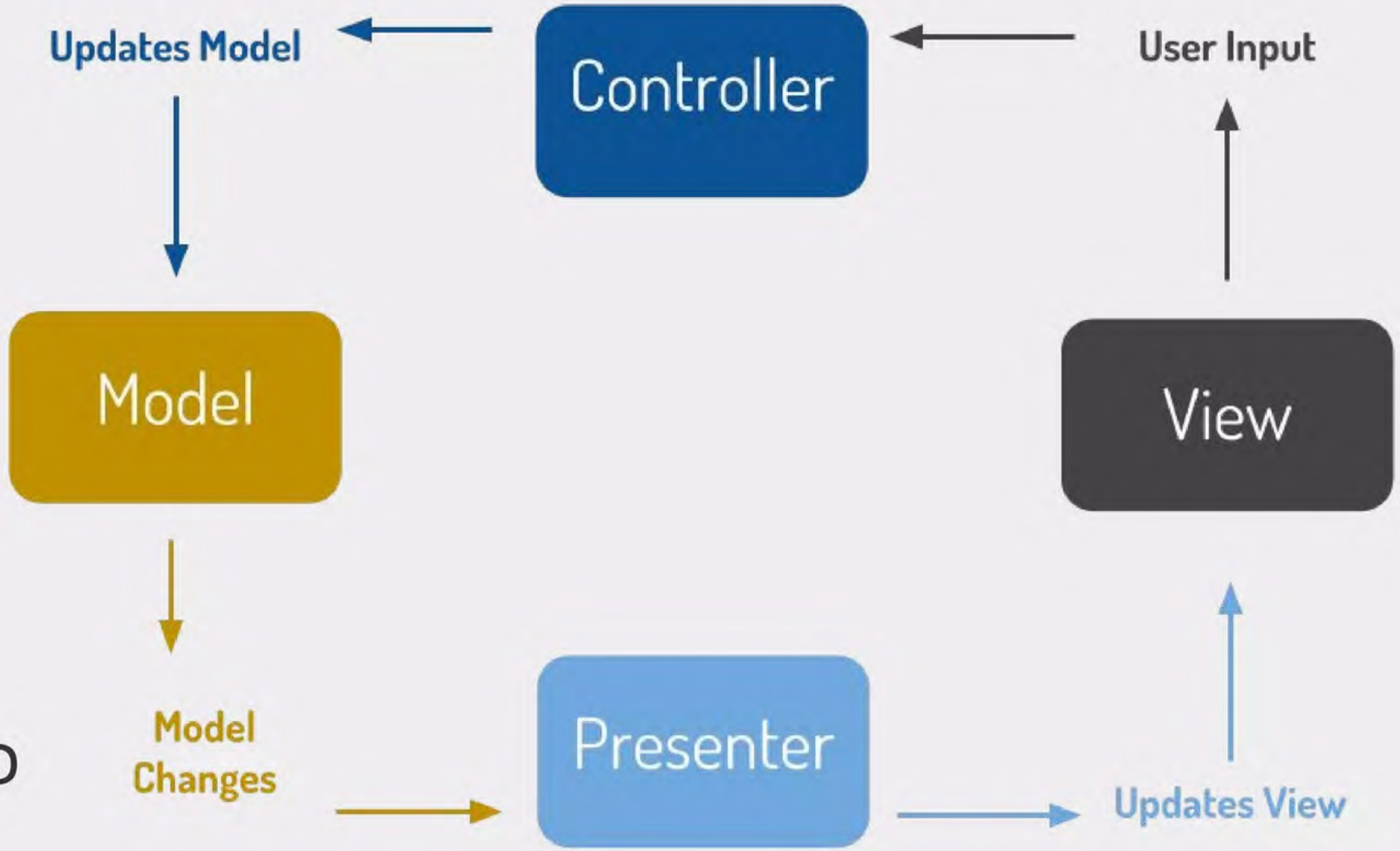
Model View Presenter
Not Exclusive
Taligent {Apple + HP + IBM}
IBM Smalltalk
Microsoft
Apple



MVC

MVP

Model View Presenter
Not Exclusive
Taligent {Apple + HP + IBM}
IBM Smalltalk
Microsoft
Apple



MVC + MVP

4 Building it in the Real World

Theme Driven Development

Sound familiar?

Additional Work

Development -> Theming

Theme Driven Development

Sound familiar?
Additional Work
Development -> Theming

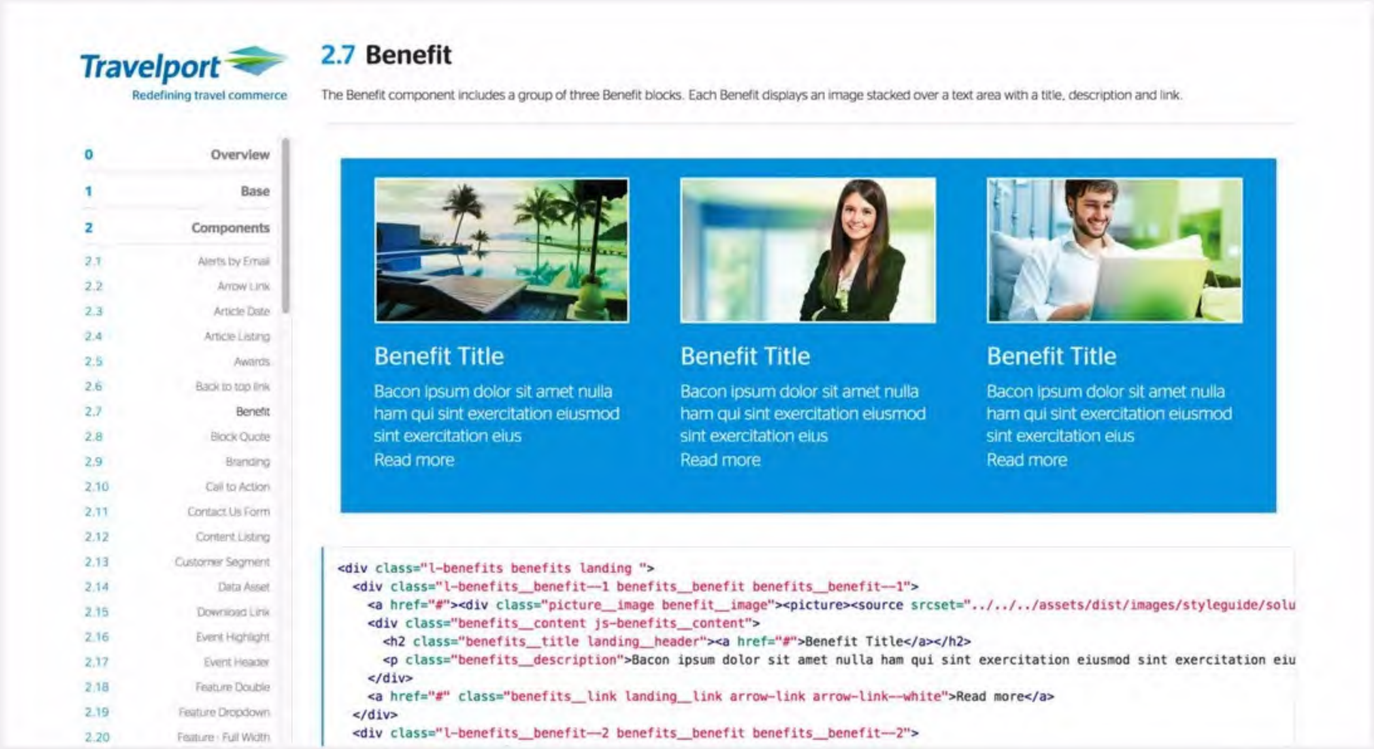
Missing Ingredient: The Presenter Layer

Drupal 7 + KSS Styleguide

Immediate Value Add

Downsides:

- Extra work
- Twig for Drupal 7
- Removes control from Drupal UI

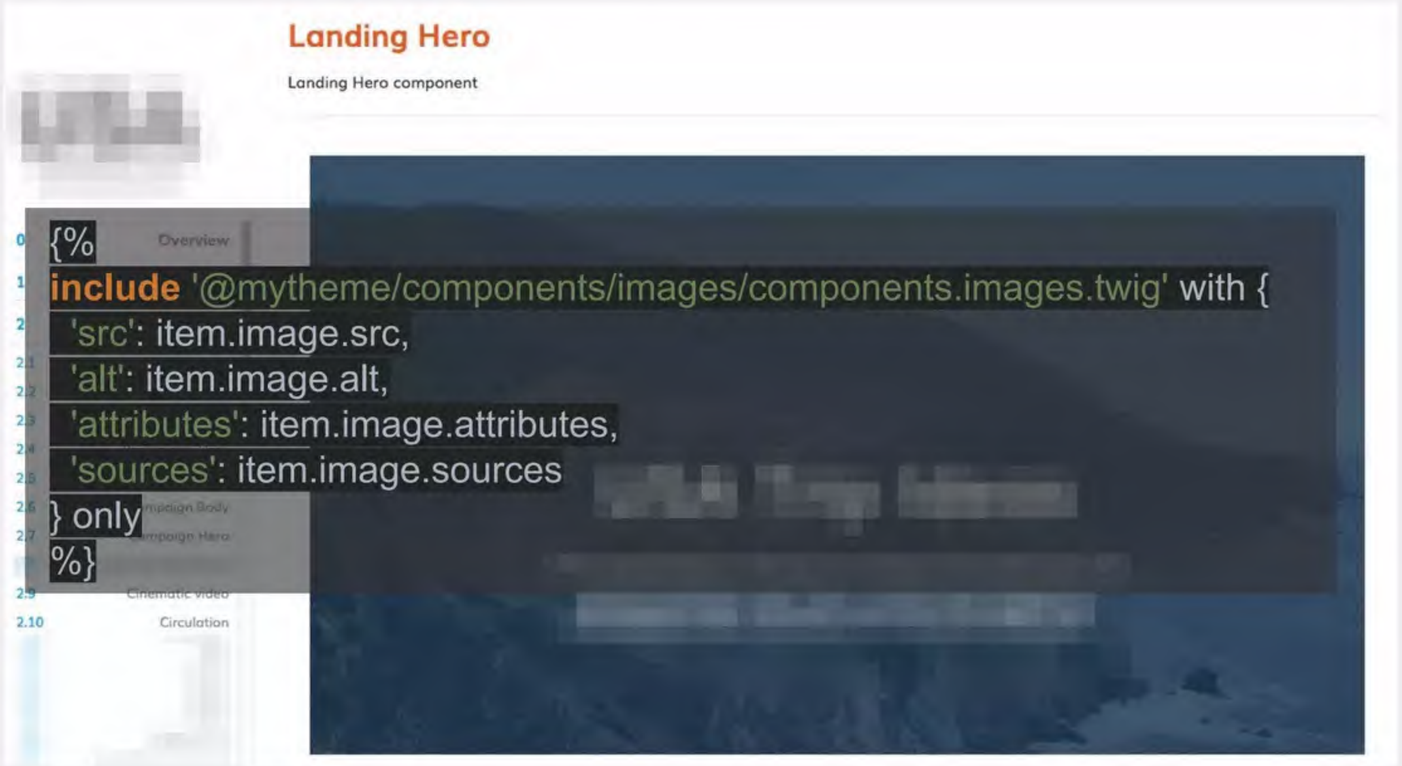


Drupal 8 + KSS Styleguide + Presenter Layer

Twig Includes

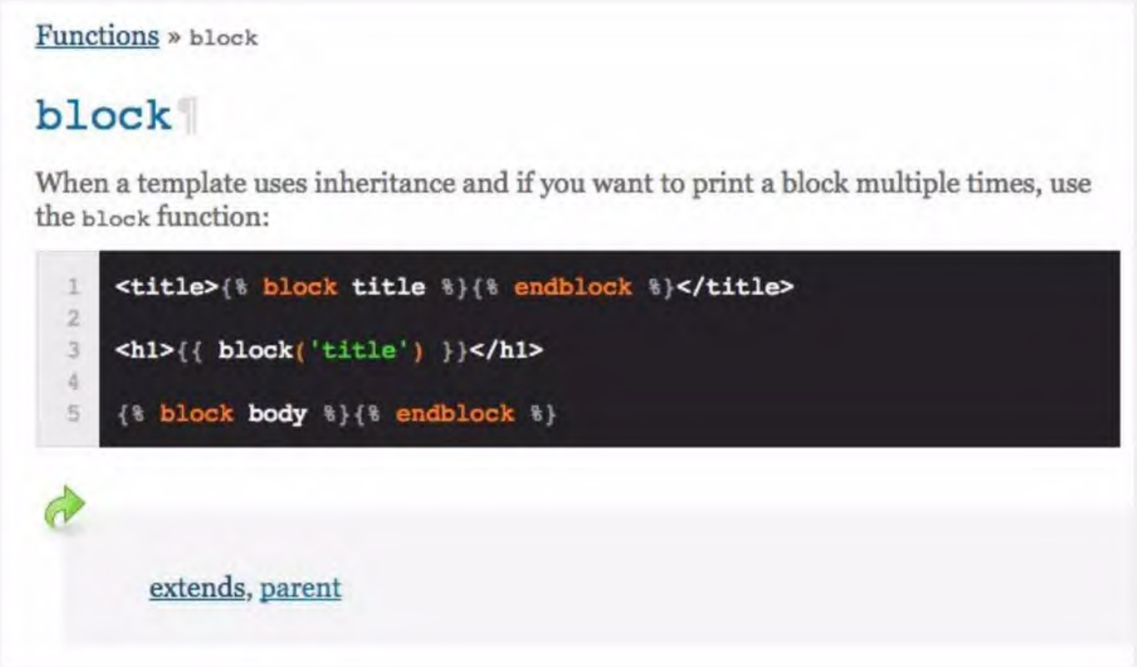
Less Overall Work

http://drupal.org/project/components



Drupal 8 + KSS + Twig Blocks

+ Keeps Drupal Render Inheritance
+ Passes control back to Drupal
- Components Specifically for Drupal



Functions » block

block

When a template uses inheritance and if you want to print a block multiple times, use the block function:

```
1  <title>{% block title %}{% endblock %}</title>
2
3  <h1>{{ block('title') }}</h1>
4
5  {% block body %}{% endblock %}
```

extends, parent

mytheme/templates/block/block--entity-field--node--field-homepage-hero-media.html.twig

```
<div{{ attributes }}>
 {% embed '@mytheme/components/hero/components.hero.twig' with {
   'heading': label,
   'items': content.field['#items'],
   'content': content
 } only
 %}
   {% block item -%}
     {{ content.field[(loop.index0)] }}
   {%- endblock item %}
 {% endembed %}
</div>
```

mytheme/lib/components/hero/components.hero.twig

```
<div class="hero">
 <div class="hero__slider slider--dots">
  {% set item_class = 'hero__slider--item' %}
  {% for item in items -%}
   <div class="{{ item_class }}">
    {% block item -%}
     {% include '@mytheme/components/hero-item/components.hero-item.twig' with {
      'item': item
     } only %}
    {%- endblock item %}
   </div>
  {%- endfor %}
 </div>
</div>
```

`mytheme/lib/components/hero-item/components.hero-item.twig`

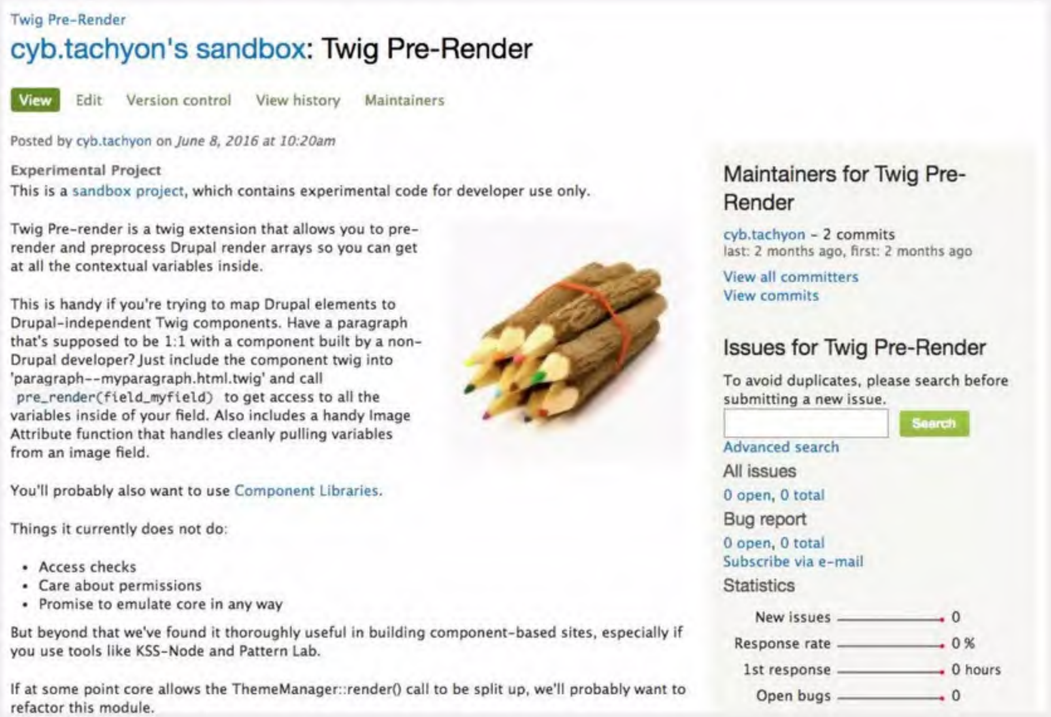
```
<h1 class="hero__title">{{ item.title }}</h1>
<div class="hero__slider--image">
 {%
 include '@mytheme/components/images/components.images.twig' with {
    'src': item.image.src,
    'alt': item.image.alt,
    'attributes': item.image.attributes,
    'sources': item.image.sources
 } only
 %}
</div>
{% if item.url|default -%}
 <div class="hero__view-more-button">
   <a href="{{ item.url }}" class="button">{{ item.viewmore }}</a>
 </div>
{%- endif %}
```

Drupal 8 + KSS + Twig Pre-Render Module

Can use Twig Components As-is

Still a WIP

Not a good substitute for Core



https://www.drupal.org/sandbox/cybtachyon/2744911

Building it in the Real World

`mytheme/templates/block/block--entity-field--node--field-facts-module.html.twig`

```
{{ attach_library('mytheme/fun_facts') }}
{% set facts = [] %}
{% for item in content.field['#items'] %}
 {% set fact = pre_render(content.field[(loop.index0)]) %}
 {% if fact.field_title|default %}
   {% set image = image_attr(fact.field_image) %}
   {%
     set fact_item = {
       'title': fact.field_title['#items'][0].value,
       'image': image
     }
   %}
   {% set facts = facts|merge([fact_item]) %}
 {% endif %}
{% endfor %}
{%
include '@mytheme/components/fun-fact/components.fun-fact.twig' with {
 'heading': 'Fun Fact'|t,
 'items': facts
} only
%}
```

`mytheme/lib/components/fun-fact/components.fun-fact.twig`

```
<div class="fun-fact module-container" style="background-image: url('{{ items[0].image.src }}')">
  <h2 class="fun-fact__heading">{{ heading }}</h2>
  <div class="fun-fact__facts">
    {% for fact in items -%}
      <div class="fun-fact__fact content-container">
        <div class="fun-fact__image">
          {%
          include '@mytheme/components/images/components.images.twig' with {
            'src': fact.image.src,
            'alt': fact.image.alt,
            'sources': fact.image.sources,
            'attributes': fact.image.attributes
          } only
          %}
        </div>
        <h4 class="fun-fact__title">{{ fact.title }}</h4>
      </div>
    {%- endfor %}
  </div>
</div>
```

When to use a Drupal Components Build

Non-Drupal FE Devs

Large Sites w/ Many Reusable Components

Sharing components between Applications (Mobile App/Desktop/Web)

When **not** to use a Drupal Components Build

Single Developer / Maintainer Scenarios

Decoupled / REST Builds (for now)

Small Budget / Low Client Investment Builds

Component Issue Workflow

Parallel Issues can be worked on at the same time

Component

The component ticket encapsulates the SASS, HTML, and style guide work necessary to create the full component entry.

Build

The build ticket includes the backend development, such as site-building and custom code necessary for the component.

Theme

The theme ticket is for building the base theme preprocessing and twig for any Drupal-level elements that make up the component.

Finalize

The finalization ticket combines all of the previous issue work and allows for any final tweaks before presentation to the client.

TIP: There can be multiple Build and Theme issues for each Component Issue, and some Components such as those containing menus or images may share Theme and Build issues between them.

Component Issue

HTML

CSS / SCSS

JS

We need a Fun Facts component that allows us to show off fun facts about our site.

User Stories
Optional - user perspective

Acceptance Criteria
Fun facts component and JS function as per design & functional spec. Facts can be navigated via next and previous arrows.

Technical Requirements
Fun facts component directory created. Deliverables: Twig component, example JSON, and KSS + Drupal JS.

Links
Component:
http://demo.dev/themes/custom/mytheme/styleguide/dist/section-components#fun-facts

Build Issue

Content Types & View Modes

Blocks (Custom & Otherwise)

Paragraphs

We need a Fun Facts component that allows us to show off fun facts about our site.

User Stories
Optional - editor perspective

Acceptance Criteria
Paragraph / block created with appropriate fields, and embedded in node display. Demo content scripted.

Technical Requirements
Fields for Fact text and Image added to the content model. Demo content generated for review.

Links
Content Fields:
http://demo.dev/admin/structure/types/manage/demo/fields

Theme Issue

The Presenter Issue

Twig Template

We need a Fun Facts component that allows us to show off fun facts about our site.

Acceptance Criteria
Paragraph / block in node display properly shows Fun Facts component.

Technical Requirements
Presenter twig created for node field and set to include Fun Facts component.

Links
Demo Content: http://demo.dev/demo-page#facts

Finalize Issue

CSS / JS Tweaks

Stakeholder Approval

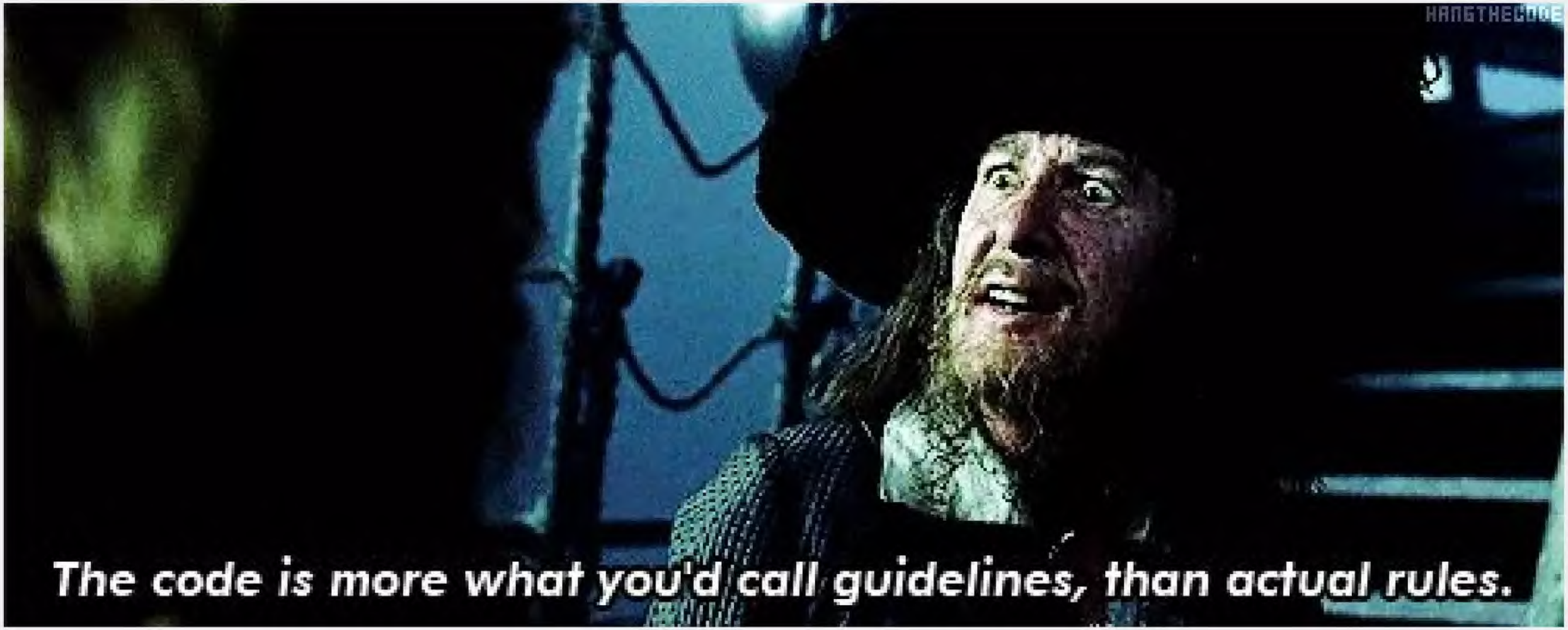
We need a Fun Facts component that allows us to show off fun facts about our site.

User Stories
Include a summary of the final user stories here.

Acceptance Criteria
Client sign-off of component in context with example or real content.

Links
Demo Content: http://demo.dev/demo-page#facts

Let's demo this!



Design

Fun Fact Design

Fun Fact

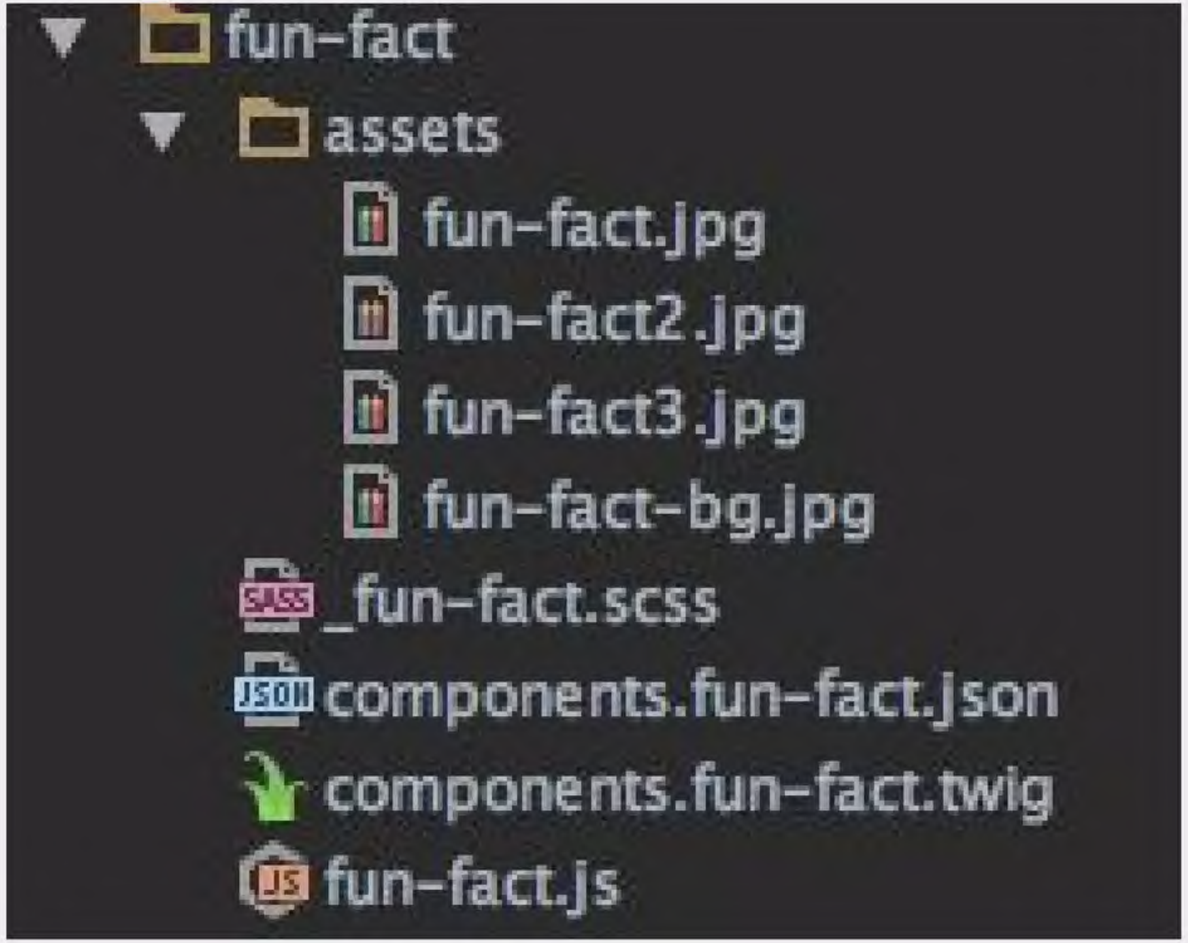
Sed posuere consectetur est at lobortis. Donec sed odio dui. Morbi leo risus, porta ac consectetur ac, vestibulum at eros. Sed posuere consectetur est at lobortis. Donec sed odio dui. Morbi leo risus, porta ac consectetur ac, vestibulum at eros.

Component

Theme Structure

```
themes
  > custom
    > mytheme
      > assets
      > lib
        > base
        > components
        > layout
        > utils
          _init.scss
          _shame.scss
          _mytheme.main.scss
      > css
      > fonts
      > js
      > node_modules
      > styleguide
        > builder
        > dist
      > templates
        .gitignore
        gulpfile.js
        package.json
        README.md
        mytheme.breakpoints.yml
        mytheme.info.yml
        mytheme.libraries.yml
        mytheme.theme
```

Fun Fact Component Structure



mytheme/lib/components/fun-fact/_fun-fact.scss

```
// Fun Fact
//
// Fun fact component
//
// Markup: components.fun-fact.twig
//
// Style guide: components.fun-fact

.fun-fact {
 @include breakpoint(max-width $md) {
  @include opacity-overlay(0.85, $color-dark-blue);
  background-position: center;
  background-size: cover;
  min-height: 250px;
  padding-bottom: 80px;
  padding-top: 80px;
 }

 @include breakpoint($md) {
  background-image: none !important;
  min-height: 350px;
  overflow: hidden;
 }

 .slick-list {
  overflow: visible !important;
 }

 .slick-arrow {
  font-size: 60px;
  left: auto;
  right: $content-container-padding-md;

  @include breakpoint(max-width $md) {
   display: none !important;
  }
 }

 .slick-prev {
  top: 130px;
 }

 .slick-next {
  top: 70px;
 }
}
```

`mytheme/lib/components/fun-fact/components.fun-fact.twig`

```
<div class="fun-fact module-container" style="background-image: url('{{ items[0].image.src }}')">
  <h2 class="fun-fact__heading">{{ heading }}</h2>
  <div class="fun-fact__facts">
    {% for fact in items -%}
      <div class="fun-fact__fact content-container">
        <div class="fun-fact__image">
          {%
          include '@mytheme/components/images/components.images.twig' with {
            'src': fact.image.src,
            'alt': fact.image.alt,
            'sources': fact.image.sources,
            'attributes': fact.image.attributes
          } only
          %}
        </div>
        <h4 class="fun-fact__title">{{ fact.title }}</h4>
      </div>
    {%- endfor %}
  </div>
</div>
```

`mytheme/lib/components/fun-fact/components.fun-fact.json`

```
{
 "heading":"Fun Fact",
 "items":[
  {
   "title":"Nullam quis risus eget urna mollis ornare vel eu leo.",
   "image":{
    "src":"../../lib/components/fun-fact/assets/fun-fact2.jpg",
    "alt":"Fun fact image"
   }
  },
  {
   "title":"Sed posuere consectetur est at lobortis. Donec sed odio dui. Morbi leo risus, porta ac consectetur ac, vestibulum at eros. Sed posuere consectetur est at lobortis. Donec sed odio dui. Morbi leo risus, porta ac consectetur ac, vestibulum at eros.",
   "image":{
    "src":"../../lib/components/fun-fact/assets/fun-fact3.jpg",
    "alt":"Fun fact image"
   }
  }
 ]
}
```

Building it in the Real World

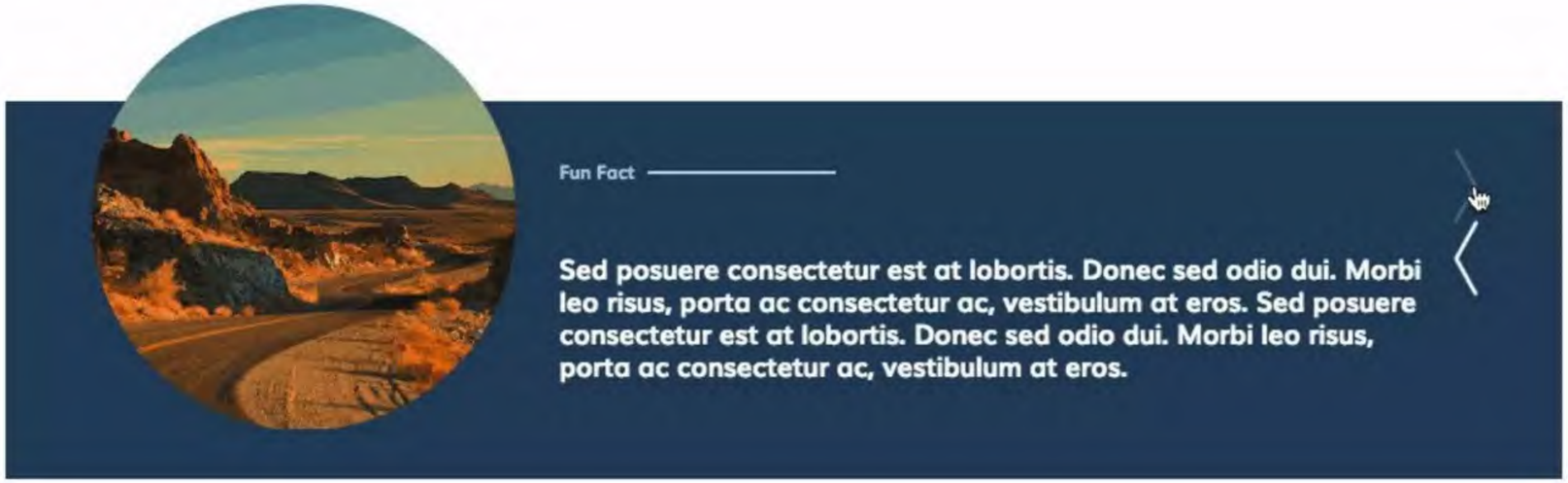
`mytheme/lib/components/fun-fact/fun-fact.drupal.js`

```
(function ($) {
 'use strict';
 Drupal.behaviors.funFactSlider = {
   attach: function (context, settings) {
     /* triggers fun facts slider */
     $(window).on('debouncedresize', function(e) {
       if (window.matchMedia('(min-width: 851px)').matches) {
         $('.fun-fact__facts').not('.slick-initialized').slick({
           autoplay: false,
           dots: false,
           arrows: true,
           infinite: false,
           mobileFirst: true,
           prevArrow: "<a href='#' class='slick-prev icon-carat-left'><span>Previous</span></a>",
           nextArrow: "<a href='#' class='slick-next icon-carat-right'><span>Next</span></a>"
         });
       }
       else {
         $('.fun-fact__facts').filter('.slick-initialized').slick('unslick');
       }
     }).trigger('resize');
   },
 };
})(jQuery);
```

Fun Fact Styleguide Entry

Fun Fact

Fun fact component



Show Markup Example

Source: `components/fun-fact/_fun-fact.scss`, line 1

Drupal Build

Fun Fact Paragraph Entity

Manage fields ☆

Edit | Manage fields | Manage form display | Manage display | Translate paragraphs type | Devel

Home » Administration » Structure » Paragraphs types » Facts

+ Add field

LABEL	MACHINE NAME	FIELD TYPE	OPERATIONS
Fact	field_title	Text (plain)	Edit
Image	field_image	Image	Edit

Drupal Theme

`mytheme/mytheme.info.yml`

```
name: My Theme
description: Default theme for my components demo.
type: theme
core: 8.x
version: VERSION
base theme: false
# libraries
libraries:
 - mytheme/global-styles
component-libraries:
 mytheme:
   paths:
     - lib

# regions
regions:
 header: Header
 primary_menu: 'Primary menu'
 page_top: 'Page top'
 page_bottom: 'Page bottom'
 content: Content
```

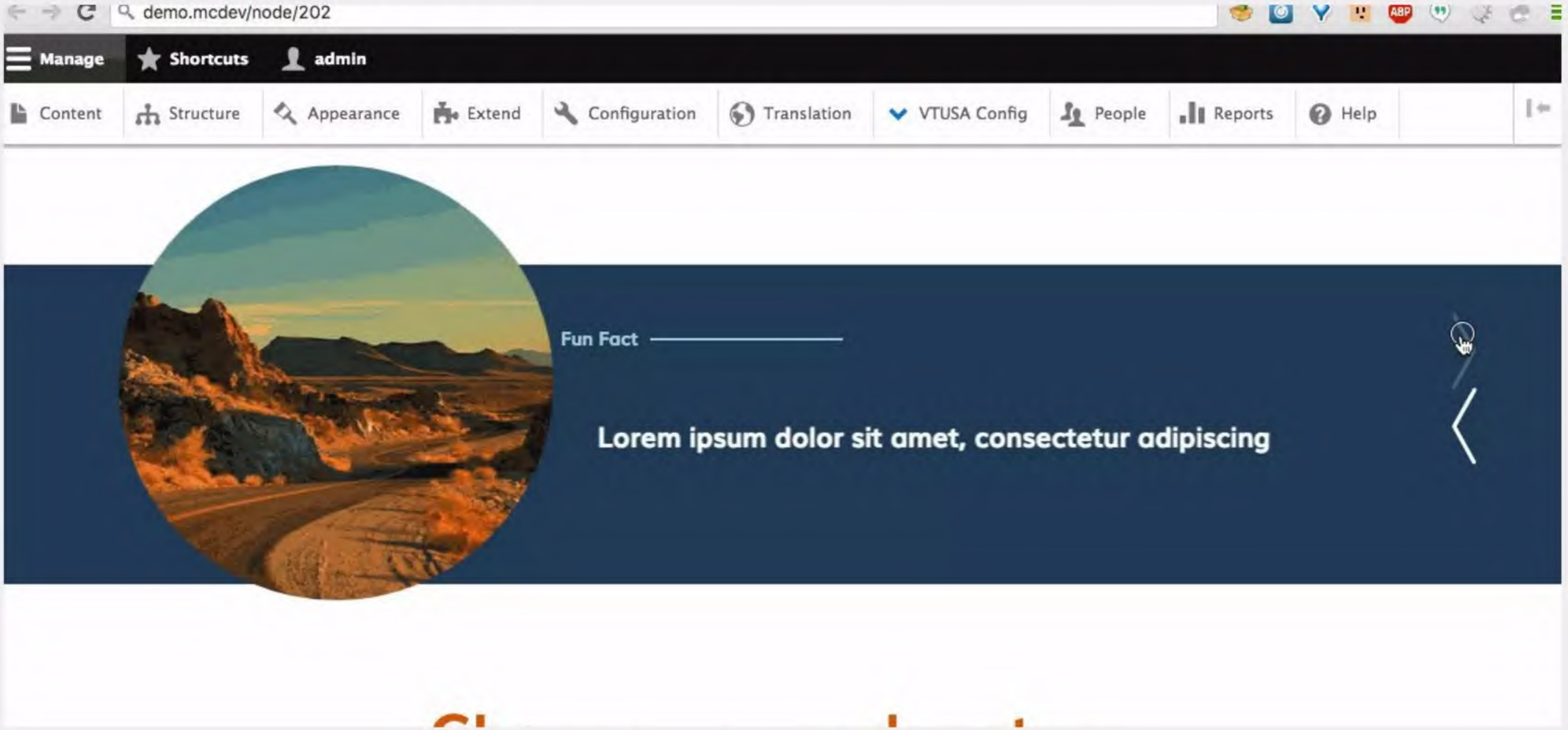
mytheme/mytheme.libraries.yml

```
global-styles:
 version: 1.x
 css:
  theme:
   css/mytheme.main.css: {}
 js:
  js/vusa-scripts.js: {}
 dependencies:
  - core/jquery
  - core/drupal
  - core/drupalSettings

fun_facts:
 version: 1.x
 js:
  lib/components/fun-fact/fun-fact.js: {}
 dependencies:
  - mytheme/slick

slick:
 version: 1.x
 js:
  styleguide/dist/kss-assets/slick.min.js: {}
```

`mytheme/templates/block/block--entity-field--node--field-facts-module.html.twig`

```
{{ attach_library('mytheme/fun_facts') }}
{% set facts = [] %}
{% for item in content.field['#items'] %}
 {% set fact = pre_render(content.field[(loop.index0)]) %}
 {% if fact.field_title|default %}
   {% set image = image_attr(fact.field_image) %}
   {%
     set fact_item = {
       'title': fact.field_title['#items'][0].value,
       'image': image
     }
   %}
   {% set facts = facts|merge([fact_item]) %}
 {% endif %}
{% endfor %}
{%
include '@mytheme/components/fun-fact/components.fun-fact.twig' with {
 'heading': 'Fun Fact'|t,
 'items': facts
} only
%}
```

Final Drupal Result



5 The Future of Components

In Contrib

- Component Libraries
  https://www.drupal.org/project/components
- Decoupled Blocks
  https://www.drupal.org/project/pdb
- Twig Pre-Render
  https://www.drupal.org/sandbox/cybtachyon/2744911
- Web Components API
  https://www.drupal.org/project/web_components

In Development

Multiple Applications / Same Responsive Components:

- Website
  - Drupal
- Mobile App
  - Ionic + Angular2
- Desktop App
  - Node.js

Drupal Core

New Theme System

Presenter Layer

Graph QL Presenters

9.0

Graph QL Presenter Example

```
{%
include '@mytheme/components/body/components.body.twig' with graphql(
 {
   media: node(id @nid) {
     title,
     image: field_image,
     media: field_featured_media {
       video: field_video_link,
       length: field_video_length,
       credit: field_credit,
       copy: field_body,
     }
   }
 },
 {'@nid': node.nid}
) only
%}
```

GraphQL Presenters in Drupal Core

+ Cleaner
+ Nuke render arrays in 9.0
+ Nuke form API
+ Standard & Easy
+ Keeps Twig presenters decoupled

- Experimental renderer
- Add GraphQL
- Build full render system (preprocess)
- Potential performance concerns (needs data requirement bubble-ability)
- Need an experimental GUI for presenters

Drupal Core

New Base Theme using Blocks Presenter Strategy

https://www.drupal.org/node/2759849

Drupal Web Components Google Summer of Code Project

Isolating Regions, Blocks etc.

Reusable & Embeddable Drupal Components

https://blog.radiumz.org/en/post/27/my-road-drupal-and-gsoc
https://summerofcode.withgoogle.com/projects/#6580831237701632

Web Components with modern JS Frameworks

React

Riot.js

Angular 2

Polymer

In Browsers

The Clock

```
<clock format="simple" />
```

Thank you!

@derekreese
slideshare.net/derekreese
https://github.com/cybtachyon/twig-standards

@Mediacurrent

slideshare.net/mediacurrent

Mediacurrent.com