

WordPress

+ React





Gregory Cornelius

@gcorne



Matías Ventura

@matias_ventura

WordCamp US 2015

What is the future
of WP Admin?

WordPress 2.0 "Duke" (2005)

The screenshot displays the WordPress 2.0 'Duke' admin interface for writing a post. At the top, a blue header bar contains the site name 'Blog' with a '(View site »)' link, and the user's name 'Howdy, admin.' with links for '[Sign Out, My Profile]'. Below this is a navigation menu with 'Dashboard', 'Write', 'Manage', 'Comments', 'Blogroll', 'Presentation', 'Plugins', 'Users', and 'Options'. The 'Write' menu is expanded to show 'Write Post' and 'Write Page'. The main content area is divided into two columns. The left column contains a 'Title' text box, a 'Post' section with 'Visual' and 'Code' tabs and a rich text editor toolbar, a 'Path:' text box, and a 'Tags (separate multiple tags with commas: cats, pet food, dogs)' text box. At the bottom of this column are three buttons: 'Save and Continue Editing', 'Save', and 'Publish'. The right column contains several settings boxes: 'Categories' with an 'Add' button and a checkbox for 'Uncategorized'; 'Discussion' with checkboxes for 'Allow Comments' and 'Allow Pings'; 'Post Password' with an empty text box; and 'Post Slug' with an empty text box. At the bottom left, an 'Upload' section is visible with fields for 'File' (containing a 'Browse...' button and 'No file selected.'), 'Title', and 'Description'.

HTML5

CSS3

pushState

localStorage

Animations



“The UI helps people understand and interact with the content, but never competes with it.”

iOS Human Interface Guidelines

Speed



WordPress 4.3.1 "Billie" (2015)

The screenshot displays the WordPress 4.3.1 'Billie' admin interface. The top navigation bar includes the WordPress logo, a home icon, the site name 'bit swapping', a notification icon with '2', and links for '+ New', 'View Post', and 'Performance'. On the right, it shows the user 'Howdy, Gregory Cornelius' with a profile picture, and dropdown menus for 'Screen Options' and 'Help'.

The left sidebar contains a menu with the following items: Dashboard, Jetpack, Posts (highlighted), All Posts, Add New, Categories, Tags, Media, Links, Pages, Comments (with a '2' badge), Appearance, Plugins, Users, Tools, Settings, Performance, and Collapse menu.

The main content area is titled 'Edit Post' with an 'Add New' button. The post title is 'React Component Composition, the Event Loop, a'. Below the title, the permalink is 'http://bitswapping.com/2015/11/react-component-composition/' with an 'Edit' button. There are also 'View Post' and 'Get Shortlink' buttons.

The editor includes an 'Add Media' button and two tabs: 'Visual' (selected) and 'Text'. The visual toolbar contains icons for bold, italic, text color, background color, list, ordered list, quote, link, unlink, table, and table of contents.

The main text area contains the following content:

Most of the time when working React, performance of the update cycle is good enough that developers don't need to think about it. What this means is that the time it takes to calculate differences between the current and next state of the virtual DOM and then apply the changes to the DOM takes less than 17 ms, which is the length of time of a single frame when rendering at 60 fps. As an application gets more complex, there will be times when 17 ms and something needs to be done.

There are two options:

1. Reduce the amount of computation that needs to be done during an update cycle (preferred).
2. Limit how often the update cycle runs to minimize the impact.

The bottom of the main content area features the heading **Uncontrolled vs. Controlled**.

The right sidebar contains the 'Publish' section with buttons for 'Save Draft' and 'Preview'. It shows the status as 'Draft' with an 'Edit' link, visibility as 'Public' with an 'Edit' link, and 15 revisions with a 'Browse' link. There is a 'Publish immediately' button with an 'Edit' link. Below this, it says 'Publicize: Not Connected' with a 'Show' link. At the bottom of the publish section are links for 'Purge from cache', 'Move to Trash', and a blue 'Publish' button.

Below the publish section are sections for 'Featured Image' with a 'Set featured image' link, 'Tags', and 'Categories'.



“...in the past WP hadn't done a good job of building the real-time experience that we as modern web customers have come to expect.”

Om Malik

Insert Media

Create Gallery

Featured Image

Insert from URL

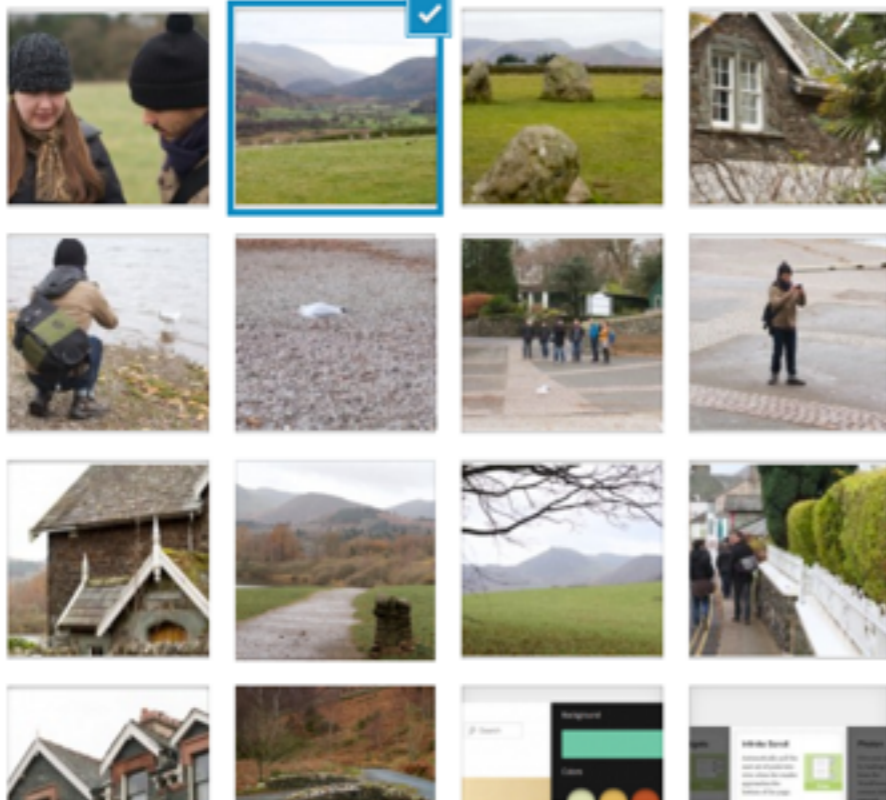
Insert Media

Upload Files **Media Library**

All media items

All dates

Search



1 selected
Clear



Themes 23

Add New

Search installed themes...

Active: 2015 Julio Customize

Twenty Eleven

Twenty Fifteen

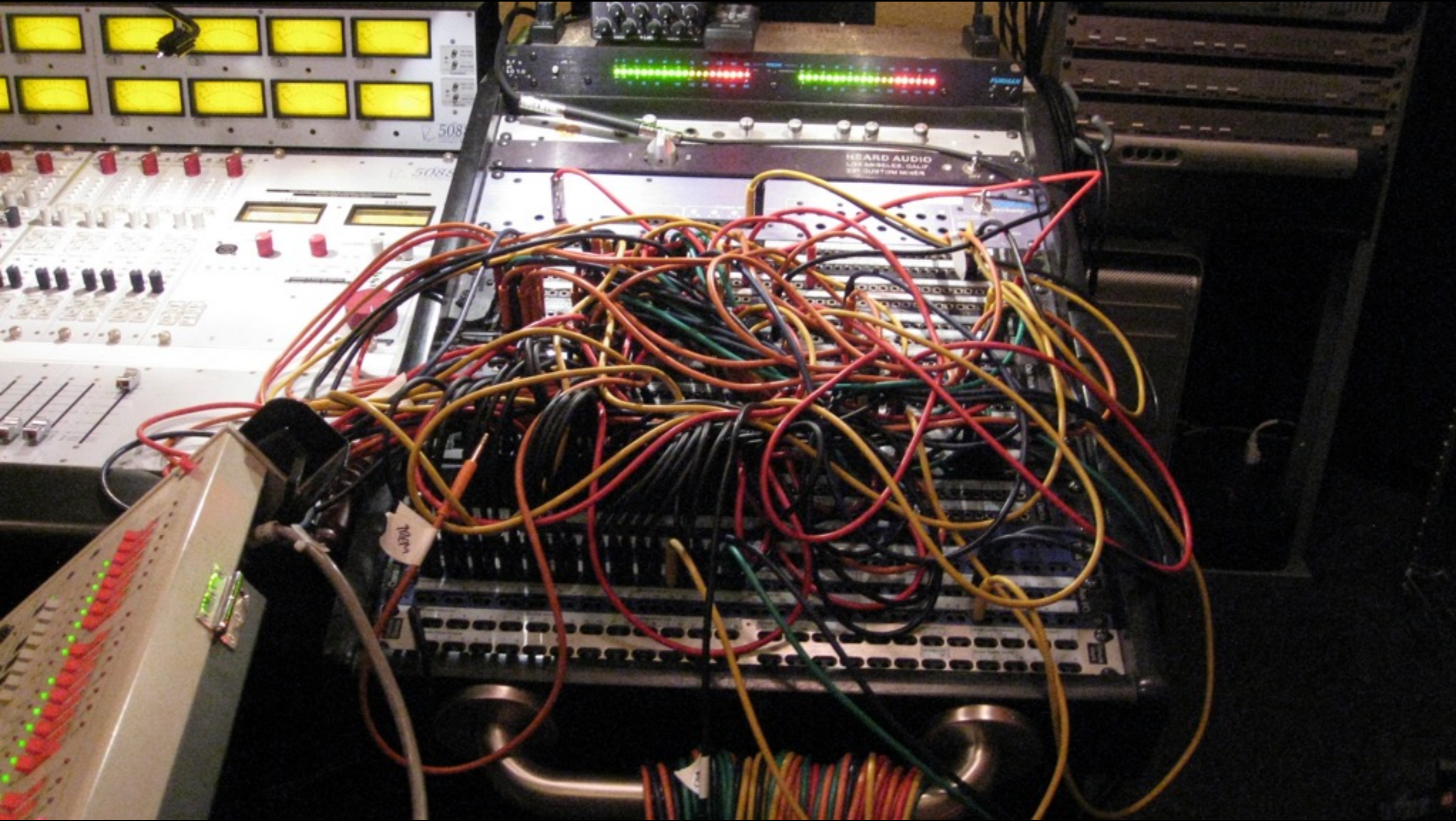
Twenty Fourteen

Twenty Ten

Twenty Thirteen

Twenty Twelve

Add New Theme





React

A declarative view layer written in pure **JavaScript** that simplifies the process of building complex UIs.



NETFLIX

YAHOO!



airbnb

HubSpot



WordPress.com

{ Calypso }

developer.wordpress.com/calypso



Single-Page App

Written in JavaScript

Driven by WP.com REST API

← Switch Site

All My Sites 4
Manage all my sites

Stats

Publish

Blog Posts Add

Pages Add

Personalize

Themes Customize

Configure

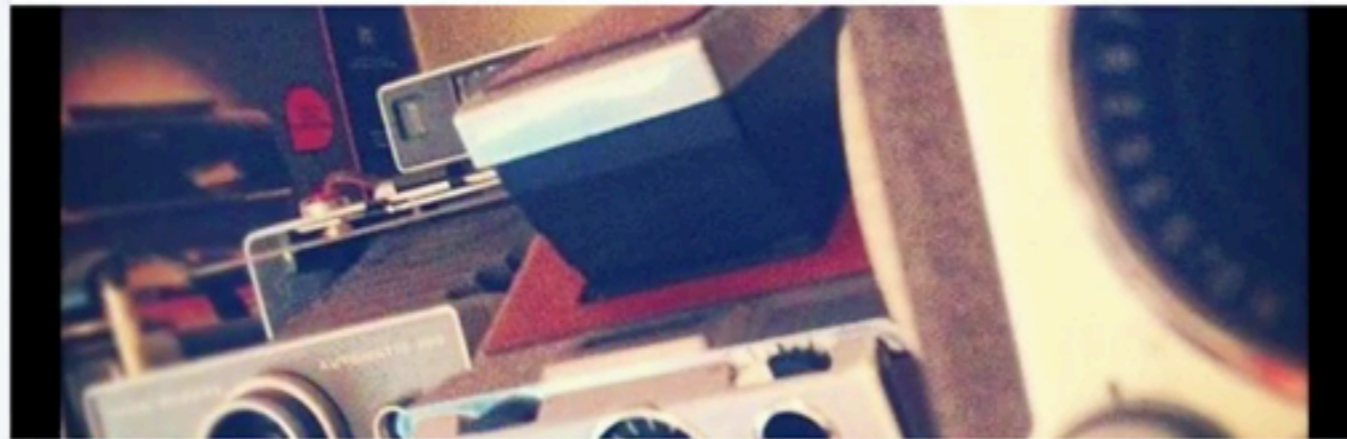
Plugins Add

Published Drafts Scheduled Trashed

Only Me Everyone



Dream in Retro



Old Polaroids

A month ago

[Comment] [Star] [Eye] 2

Edit

View

Stats

Trash



Dream in Retro



 Panda Raccoon Journal
By David Jackson

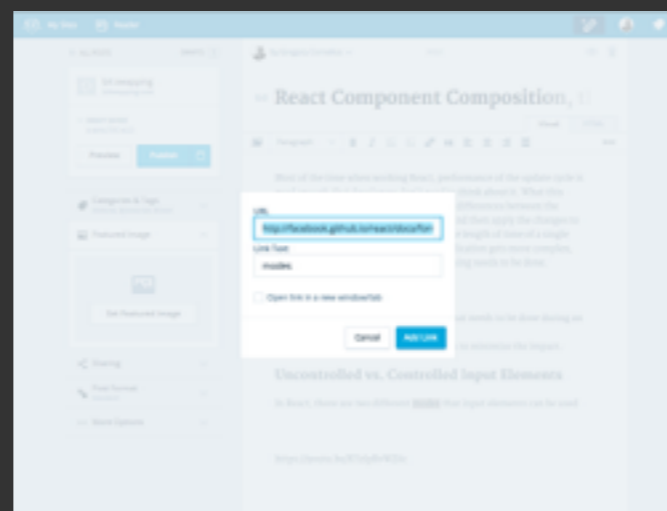
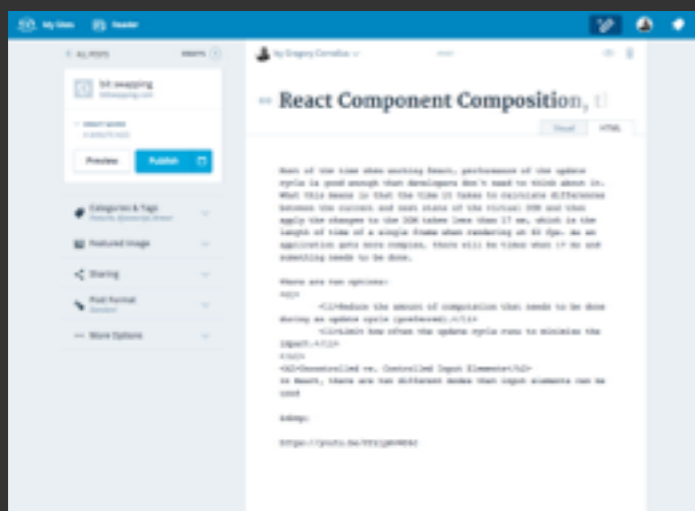
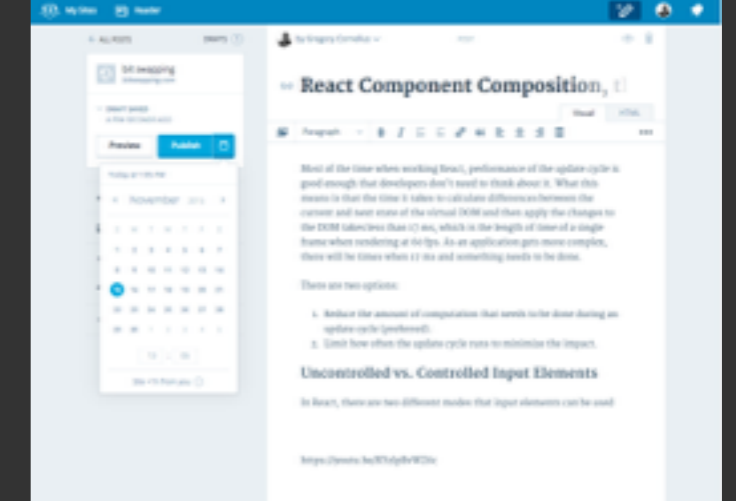
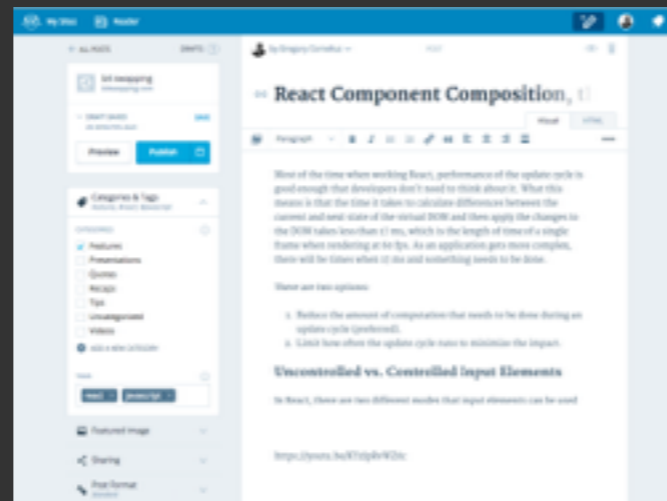
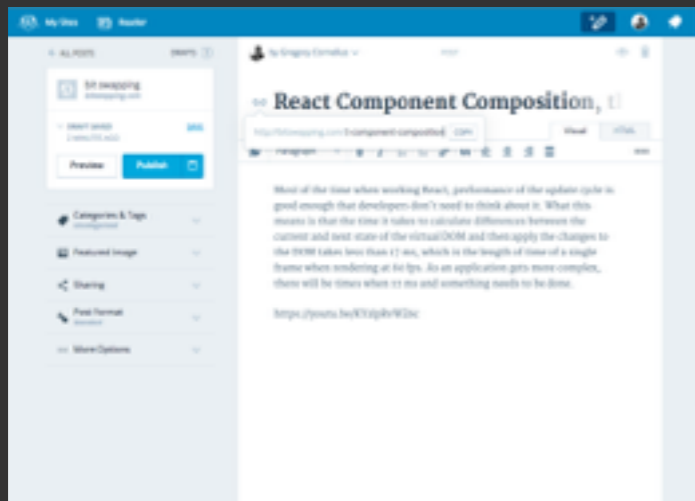
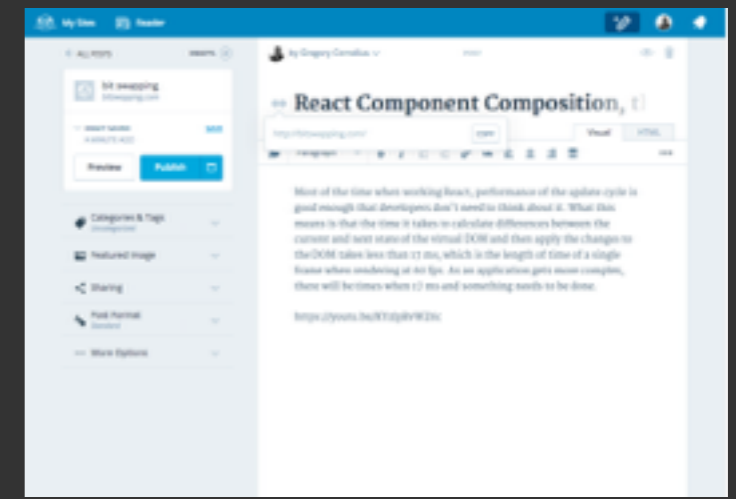
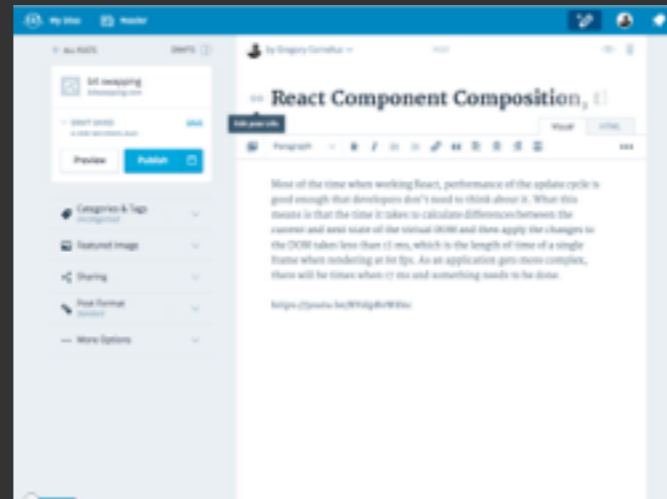
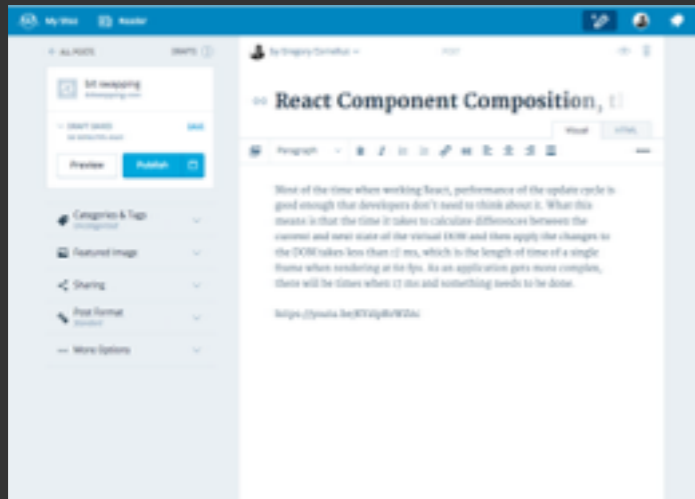


Traveling for the Holidays

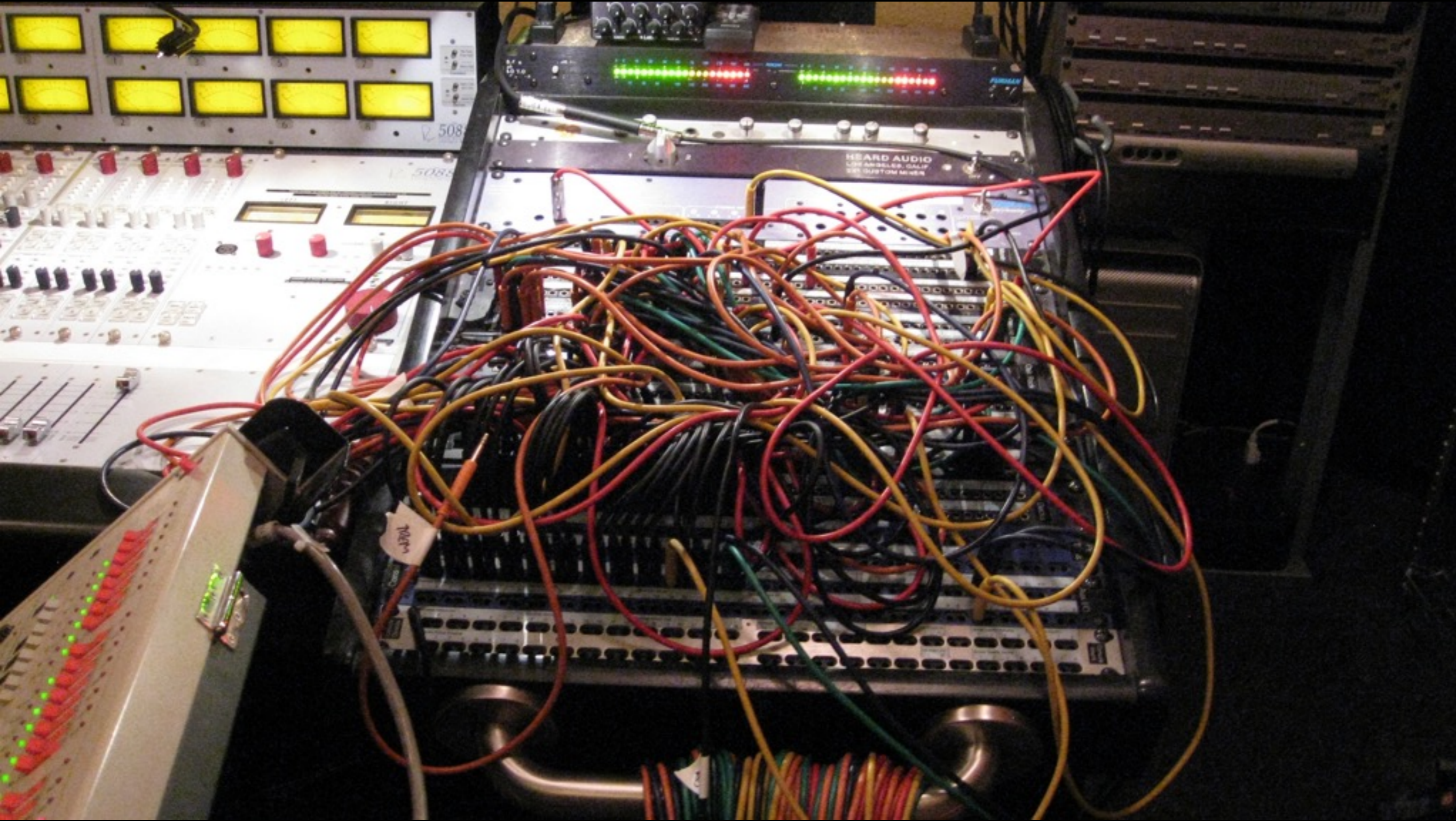
Holidays are around the corner, and with them come plans for travel. Most years I am blessed to not have to go out of the safety of my own home (I host my family for holiday dinners) but this year I decided to do somethi...

🕒 2 years ago

💬 ★ 👁



Editor UI States



React key features

- * Small API that is easy to learn.
- * No need for a separate template language. Embrace JavaScript.
- * Events are bound as element properties which limits need for DOM traversal.
- * Composition of components is simple and first-class.

Just `Render()`

React will be smart about updating the DOM.

```
const DOMNode = document.getElementById( 'hello-container' );

function HelloMessage( props ) {
  return React.DOM.h3( null, props.name );
}

ReactDOM.render(
  React.createElement( HelloMessage, { name: 'Hello, World!' } ),
  DOMNode
);
// DOMNode.innerHTML = '<h3>Hello, World!</h3>'

ReactDOM.render(
  React.createElement( HelloMessage, { name: 'Hello, World!' } ),
  DOMNode
);
// no DOM operations

ReactDOM.render(
  React.createElement( HelloMessage, { name: 'Howdy, Matías' } ),
  DOMNode
);
// DOMNode.firstChild.textContent = 'Howdy, Matias';
```

see <http://codepen.io/gcorne/pen/meNWPP>

{ functions > template language }

```
const HelloComponent = React.createClass({
  render() {
    return React.DOM.div(
      { className: 'hello' },
      React.DOM.h3( null, 'Hello, World!' )
    );
  }
});
```

```
ReactDOM.render(
  React.createElement( HelloComponent ),
  document.getElementById( 'hello-container' )
);
```

see <http://rauchg.com/2015/pure-ui/>

Increase readability via the **JSX** syntax extension

```
const HelloComponent = React.createClass({
  render() {
    return (
      <div className="hello">
        <h3>Hello, World!</h3>
      </div>
    );
  }
});

ReactDOM.render(
  <HelloComponent />,
  document.getElementById( 'hello-container' )
);
```

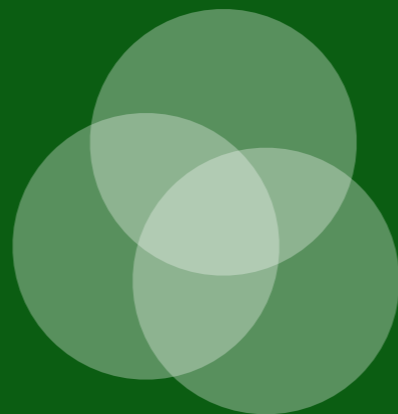
see <http://codepen.io/gcorne/pen/MaNmWd>

Compiling JSX



```
export default React.createClass( {
  getInitialState() {
    return {
      showPreview: false
    };
  },
  showPreview() {
    this.setState( { showPreview: true } );
  },
  render() {
    return (
      <div className="post">
        { this.state.showPreview &&
          <Preview post={ post } />
        }
        <span className="post__date">{ this.props.post.date }</span>
        <h4 className="post__title">
          <a className="post__title-link" href={ this.props.post.URL }>
            { this.post.title || '(no title)' }
          </a>
        </h4>
        <button onClick={ this.showPreview }>Preview</button>
      </div>
    );
  }
} );
```

Why do
components
matter?



Progressive Enhancement

JavaScript
(feature A)

JavaScript
(feature B)

JavaScript
(feature C)

JavaScript
(feature D)

JavaScript
(feature E)

CSS

HTML (form)

Components

Component
(feature A)

JavaScript

CSS

HTML

Component
(feature B)

JavaScript

CSS

HTML

Component
(feature C)

JavaScript

CSS

HTML

Component
(feature D)




















JavaScript

CSS

HTML

JSX + Sass

Branch: **master** [wp-calypso](#) / [client](#) / **components** /

 accordion	Initial commit of wp-calypso
 add-new-button	Initial commit of wp-calypso
 author-selector	Initial commit of wp-calypso
 bulk-select	fix "failed propTypes" warning for BulkS
 button-group	Initial commit of wp-calypso
 button	Initial commit of wp-calypso
 chart	Initial commit of wp-calypso
 comment-button	Initial commit of wp-calypso
 count	Initial commit of wp-calypso
 data	Purchases: Add `CancelPurchasePla
 date-picker	Initial commit of wp-calypso
 dialog	Framework: Use classnames in place
 domains	Initial commit of wp-calypso
 drop-zone	Initial commit of wp-calypso
 email-verification	Initial commit of wp-calypso
 emojify	Initial commit of wp-calypso
 external-link	fix unique key warning for ExternalLin
 flag	Initial commit of wp-calypso
 foldable-card	Initial commit of wp-calypso

Live components gallery

Calypso Docs


 Search

 The Calypso Guide

 Contributing

Live Docs

 UI Components

 Icons

 Search components...

Notices

Compact Notices

 I'm an `is-info` notice.




 I'm an `is-error` notice.



 I'm an `is-warning` notice with an action.

Update

 I'm an `is-success` notice with an arrow link.

Preview 

Button

Compact Buttons

Button

 Icon button



Disabled button

Scary button

 Scary icon button



Scary disabled button

Composition

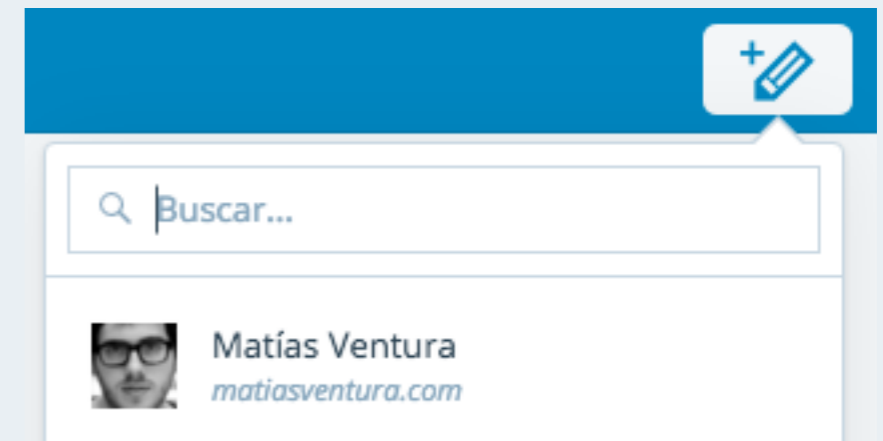
<Site>



<Sidebar>



<SitePopover>



<EditorGroundControl>



Components begin to
define app level **semantics**.

NATIVE HTML

<header>

APP

<Site>

Defines a language that
transcends the medium,
one which can be applied to
mobile, the **web**, and **desktop**.

This is just the **beginning**...

github.com/Automattic/wp-calypso

Using React in **Plugins & Themes**



{ WP React Boilerplate }

github.com/gcorne/wp-react-boilerplate

Thoughts

- * Library of WP UI components?
- * A new WP Admin UI powered by components?
- * React Native WordPress apps that are pluggable and accessible to more contributors?
- * How do plugins look when they can be API focused, client focused, or both?

do()

- * Install **Calypso** and play around with it.
- * If you consider yourself a **PHP** developer that knows jQuery, consider investing in learning more **JavaScript**.
- * Try dipping your toes in the water using **React** for something.
- * Take a look at React Native.

Thank you.

Gregory Cornelius

@gcorne

Matías Ventura

@matias_ventura