

Gatsby.js布教計画

[NGK2020S 昼の部 LT大会](#)
[hpp\(@osaremochi\).](#)



だれ？



hpp

名古屋大学情報学部コンピュータ科学科 3 年

- GitHub: [hppRC](https://github.com/hppRC)
- Qiita: [hppRC](https://qiita.com/hppRC)
- Twitter: [@osaremochi](https://twitter.com/@osaremochi)



だれ？

- 趣味で React とか Go とか Rust とか
- 学生実験でコンパイラを書いたり
- AtCoder 水(1269)
- Gatsby.js Maintainer

今日話したいこと

JAMStack



Gatsby.js



JAMStack って？

Netlify の創業者が提唱した
新しいアーキテクチャ

- JavaScript
- APIs
- Markdown

JAMStackとは

よくある「動的」な処理を外部に移した静的な Web サイトの構成

サーバに依存しないフロントエンドの技術構成

JAMStackでよく使うもの。。。

- headless CMS
- markdown からブログ記事をビルド
- 各種SaaS (Algolia とか firebase とか)
- あとはホスティングサービスに置くだけ

なるほど

Gatsby.jsって？

- React製の静的サイトジェネレータ
- 高度なパフォーマンス最適化をよしなにやってくれる
- Webpackなどの煩雑な設定を隠蔽、Web制作を楽に
- JAMStackの申し子

実はかなりGithubの★が
多かったりする

StaticGen

A List of Static Site Generators for JAMstack Sites



[About](#) [Contribute](#) [About JAMstack](#) [Need a Static CMS?](#)

Filter

Any Language ▾ Any Template ▾ Any License ▾

Sort

Stars (7 days) ▾

Next.js

★ 44106 ⚡ 279 ⛓ 6070 🐦 N/A
+216 +20 +78

A framework for statically-exported React apps (supports server side rendering)

Languages: JavaScript
Templates: React
License: MIT

Deploy to Netlify

Hugo

★ 40952 ⚡ 448 ⛓ 4622 🐦 9014
+140 +9 +13 +59

A Fast and Flexible Static Site Generator.

Languages: Go
Templates: Go
License: Apache-2.0

Deploy to Netlify

Docusaurus

★ 15139 ⚡ 57 ⛓ 1484 🐦 1415
+133 +9 +27 +12

Easy to Maintain Open Source Documentation Websites

Languages: JavaScript
Templates: Markdown
License: MIT

Deploy to Netlify

Get started with one click!

For generators with the "Deploy to Netlify" button, you can deploy a new site from a template with one click. Get HTTPS, continuous delivery, and bring a custom domain, free of charge.

Want your own Deploy to Netlify button? [Learn more here](#).

Gatsby

★ 41472 ⚡ 591 ⛓ 7026 🐦 44897
+132 -12 +44 +417

Build blazing fast, modern apps and websites with React

Languages: JavaScript
Templates: React
License: MIT

Nuxt

★ 24858 ⚡ 250 ⛓ 2160 🐦 32941
+88 +7 +10 +146

A minimalistic framework for serverless Vue.js applications.

Languages: JavaScript
Templates: Vue
License: MIT

Eleventy

★ 4049 ⚡ 165 ⛓ 126 🐦 3088
+59 +10 +1 +58

A simpler static site generator

Languages: JavaScript
Templates: Liquid, Nunjucks, Handlebars, Mustache, EJS, Haml, Pug, JavaScript, HTML, Markdown

Hexo

★ 29304 ⚡ 231 ⛓ 3818 🐦 2229
+57 -1 -- +5

Hexo is a fast, simple and powerful blog framework.

Languages: JavaScript
Templates: EJS, Pug, Haml, Swig, Nunjucks, Mustache, Handlebars,

Gatsbyの何がいいの？

Gatsbyの何がいいの？

早い・安い・うまい(?)



早い?

エコシステムが整っている 

パフォーマンスが良い 

早い!

- starter(Pull Requestで誰でも追加できる)
- theme(Wordpressにあるような機能)
- 静的サイトならではのデプロイの簡単さ
- PWA対応などがプラグインすぐに対応

安い?

ランニングコストが安い！ 

安い!

Gatsby製のサイトは静的なReactアプリケーション

- 動的なデータ取得はビルド時に終わっている
- つまりサーバ不要、静的ファイルを置くだけでサイトになる
- もちろんクライアントサイドでのデータ取得もできる

安い!!

静的サイトホスティングサービスは大量に存在する

- **Netlify**(おすすめ) 
- GitHub Pages
- firebase hosting
- S3 ...

多くは無料、手軽に始められる

うまい?

色々な技術に触れることができる

→ 美味しい

うまい!!!

- データを取得する操作はGraphQLで統一
- 内部の状態管理にReduxを利用(外からは意識しない)
- mdx, Server Side Renderingの考え方

GraphQLとか難しそう...

GraphQLとか難しそう...

大丈夫です 

ple-image.tsx

sx

a.tsx

png
ongU
●
●
M

```
14  export const useSiteMetadata = () => {
15    const data = useStaticQuery<Props>(`graphql`  
16      query {  
17        site {  
18          siteMetadata {  
19            siteTitle  
20            siteTitleAlt  
21            siteHeadline  
22            siteUrl      hpp, 12 days ago • Initial commit  
23            siteDescription  
24            siteLanguage  
25            author  
26          }  
27        }  
28      }  
29    `);  
30  
31    return data.site.siteMetadata;  
32  };  
33  
34  export default useSiteMetadata;
```

とっても簡単 

React初心者にも上級者にもオススメです

煩わしい部分を取り除いて開発体験を上げてくれるフレームワーク

どんなものが作れるの？

React

ユーザインターフェース構築のための JavaScript ライブライ

[スタートガイド](#)[チュートリアルを見る >](#)

宣言的な View

React は、インタラクティブなユーザインターフェイスの作成にともなう苦痛を取り除きます。アプリケーションの各状態に対応するシンプルな View を設計するだけで、React はデータの変更を検知し、関連するコンポーネントだけを効率的に更新、描画します。

宣言的な View を用いてアプリケーションを構築することで、コードはより見通しが立ちやすく、デバッグのしやすいものになります。

コンポーネントベース

自分自身の状態を管理するカプセル化されたコンポーネントをまず作成し、これらを組み合わせることで複雑なユーザインターフェイスを構築します。

コンポーネントのロジックは、Template ではなく JavaScript そのもので書くことができるので、様々なデータをアプリケーション内で簡単に取り回すことができ、かつ DOM に状態を持たせないようにすることができます。

一度学習すれば、どこでも使える

React と組み合わせて使用する技術に制限はありません。React を使って新しい機能を追加する際に、既存のソースコードを書き換える必要はありません。

React は Node を使ったサーバー上でもレンダーできますし、React Native を使うことでモバイルアプリケーションの中でも動きます。

シンプルなコンポーネント

React コンポーネントを作成するには `render()` メソッドを実装します。このメソッドは、受け取った入力データを元に、表示する内容を返す役割を担

LIVE JSX EDITOR

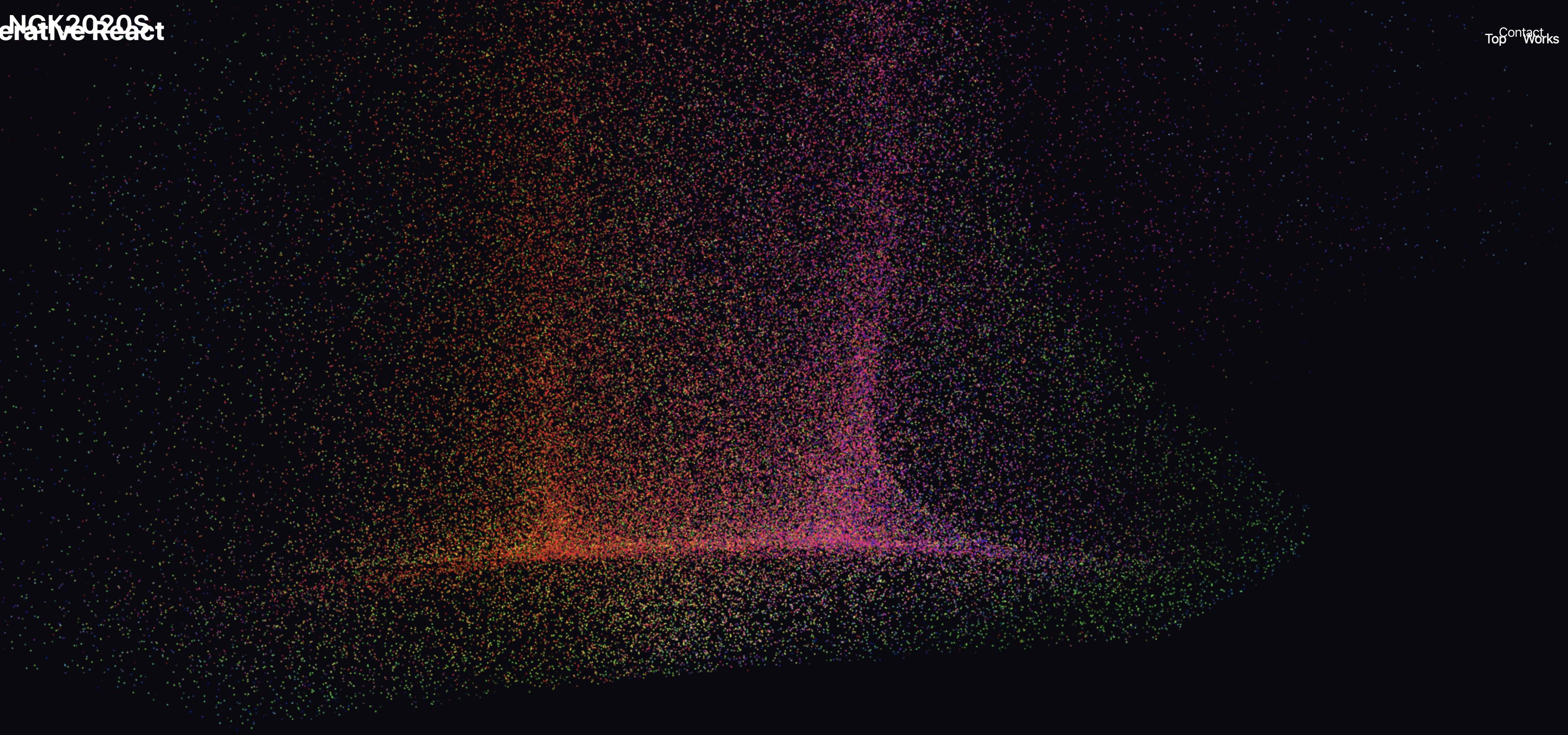
 JSX?

```
class HelloMessage extends React.Component {
  render() {
    return (
      <div>
        Hello {this.props.name}
      </div>
    )
  }
}
```

RESULT

Hello Taylor

React公式サイト



Generative React

クリエイティブコーディングで遊ぶ



WebGLやp5.jsで作る作品置き場

<https://github.com/hppRC/generativeract>

このスライド

MDXで書いてGatsbyでビルド
Netlifyにデプロイしています

ngk2020s.hpprc.com

あなたも是非Gatsby.jsで
楽しく開発しませんか？

今回紹介したのはGatsbyの魅力の一部🎃

ご清聴頂き

ありがとうございました！

参考資料

- [JAMstackってなに？実践に学ぶ高速表示を実現するアーキテクチャの構成](#)
- [React.js](#)
- [Gatsby.js](#)
- [MDX-DeckとCode Surferでスライドを作ろう！](#)
- [<https://jamstack.org/>](#)