SITECORE

# Sitecore Headless Development with Next.js

Next.js support has been added to the 16.0 release of the Headless Rendering Suite

**PRESENTED BY:**
**Thomas Desmond &**
**Nick Wesselman**

© 2021 Sitecore Corporation A/S. All Rights Reserved.

# Today's speaker

## Thomas Desmond

**Javascript Technical Evangelist, Sitecore**

- Started at Sitecore in January
- Long time frontend developer
- Located in San Diego, California

@ThomasJDesmond

www.thetombomb.com

© 2021 Sitecore Corporation A/S.

# Today's speaker

## Nick Wesselman

**Product Manager, Developer Experience, Sitecore**

- Manage Sitecore CLI, ASP.NET Core, JSS, Sitecore Docker Tools
- Working with Sitecore for over 12 years
- Lives in Asheville, North Carolina, USA

@techphoria414
Nick.Wesselman@sitecore.com

© 2021 Sitecore Corporation A/S.

# Agenda

01 Next.js in JSS

02 Jamstack

03 Next.js Benefits

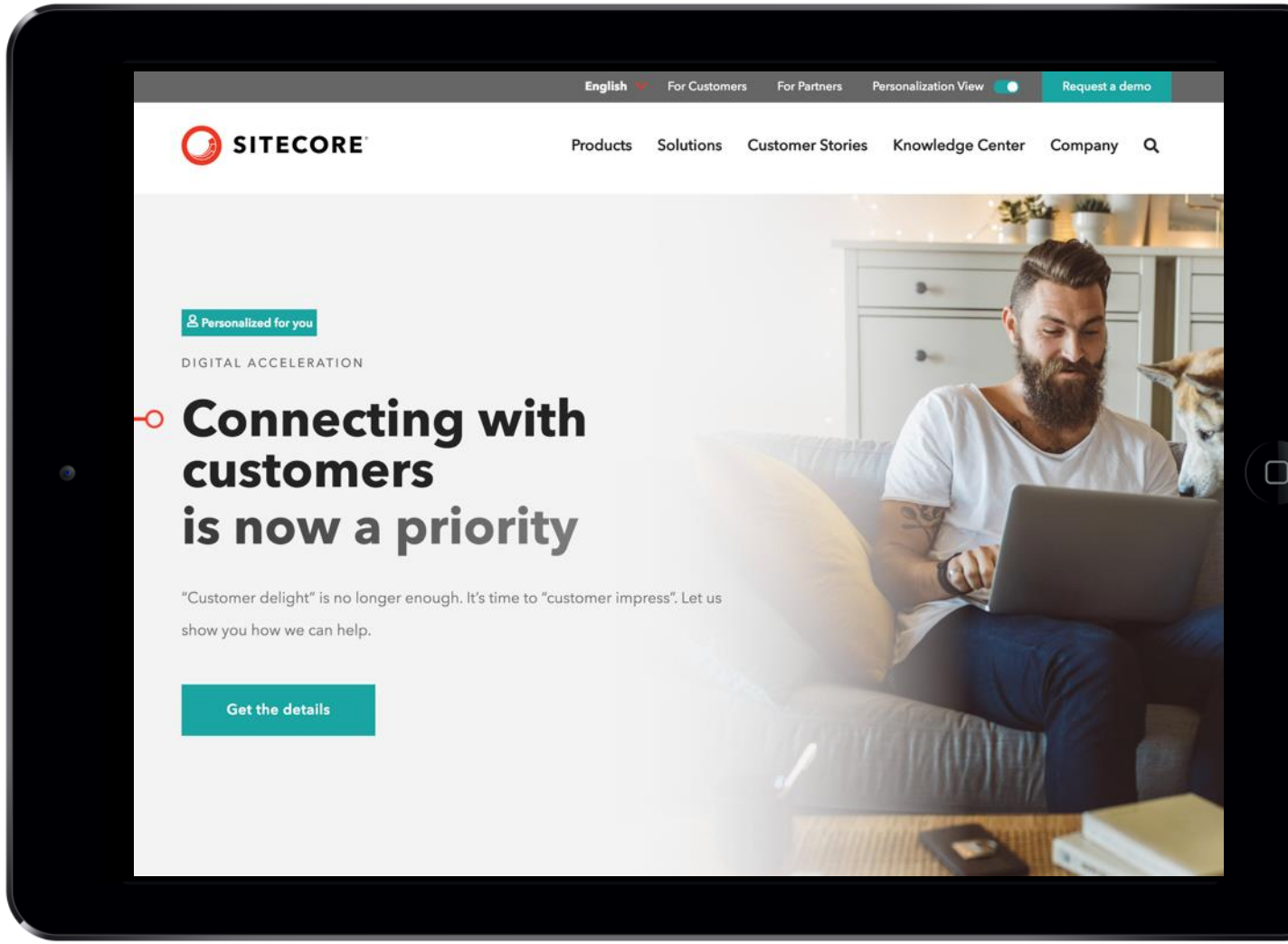
04 Demo

© 2021 Sitecore Corporation A/S.

# The Headless Rendering Suite

## Headless Rendering Suite

- JSS 16.0
- ASP.NET 16.0
- Headless Module 16.0



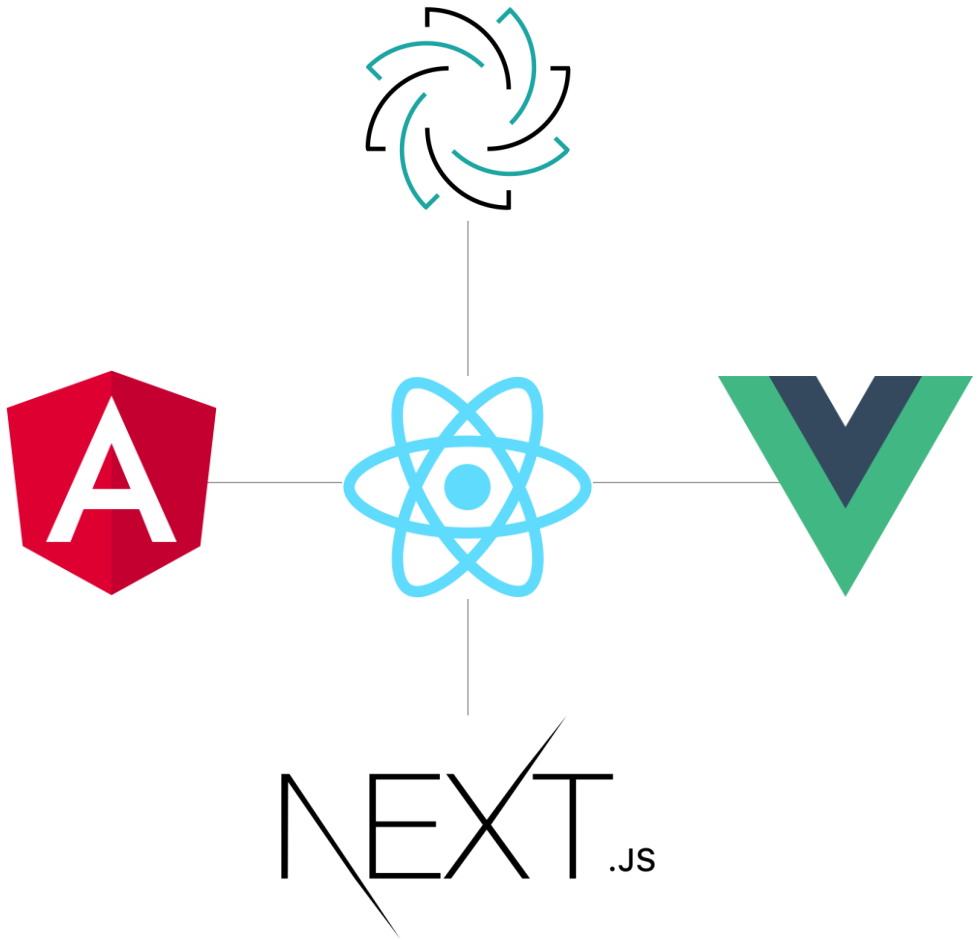

© 2021 Sitecore Corporation A/S.

SITECORE

# Next.js support in JSS 16.0

© 2021 Sitecore Corporation A/S.

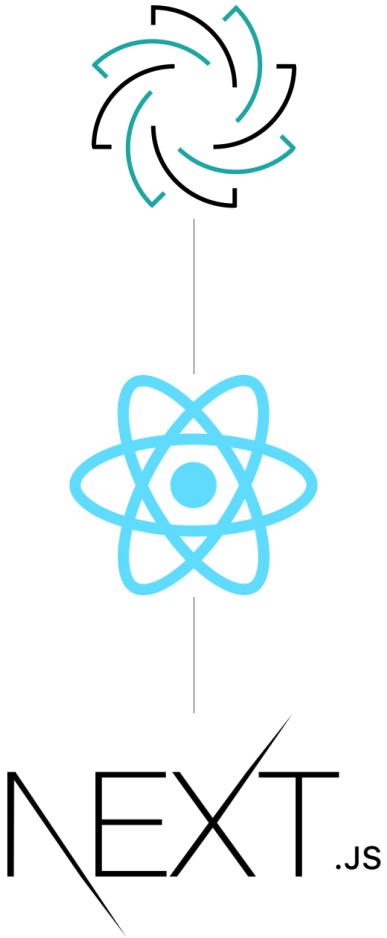
# Next.js support in JSS



- **New npm package**: sitecore-jss-nextjs
- New Next.js sample app

© 2021 Sitecore Corporation A/S.

# Next.js support in JSS



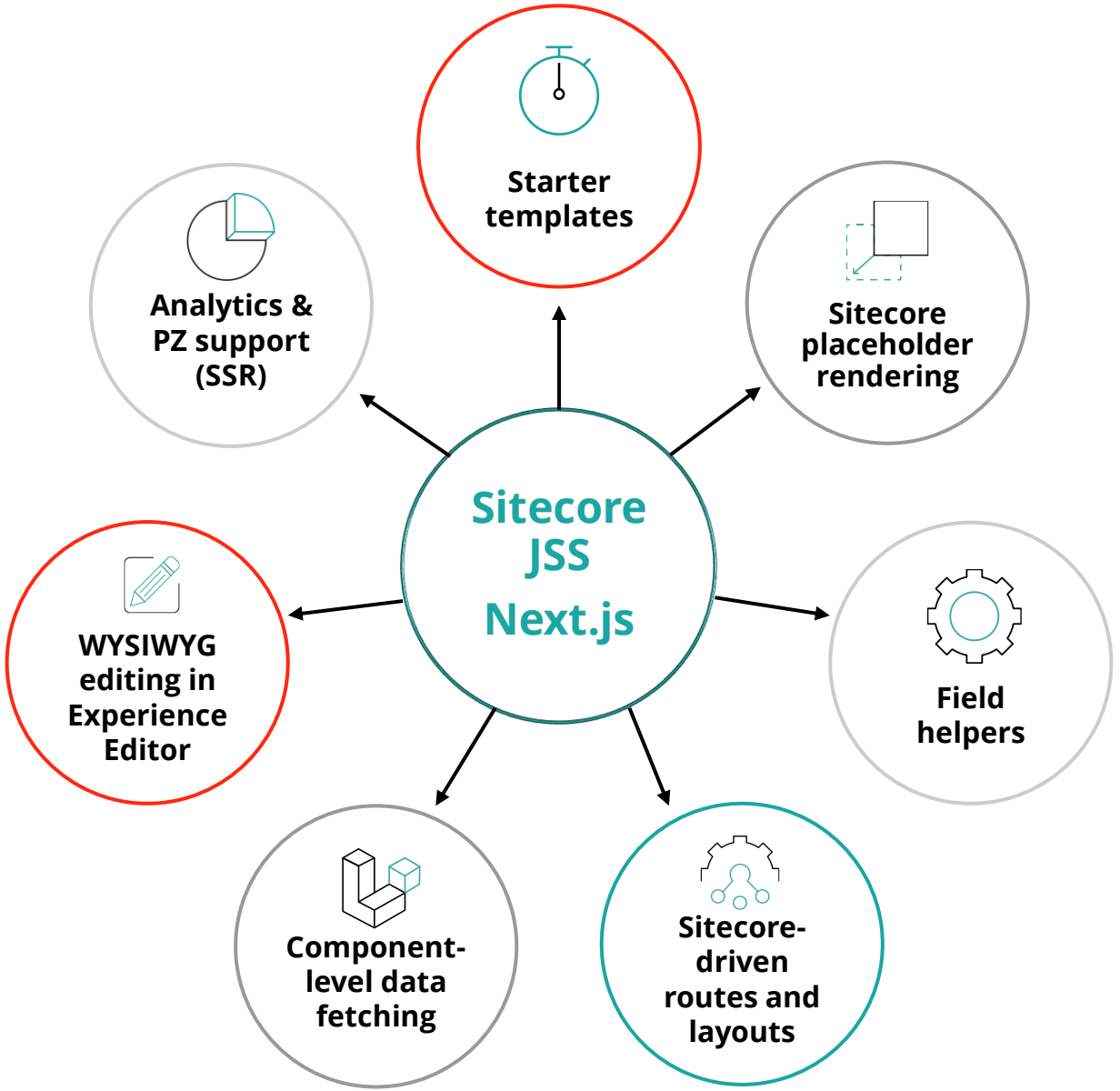

- **New npm package**: sitecore-jss-nextjs
- New Next.js sample app

**Continued support for existing features in JSS and Sitecore**

- Developer workflows (disconnected & connected)
- Tracking analytics & personalization (with SSR)
- Inline content editing & preview with Experience Editor

© 2021 Sitecore Corporation A/S.

# Next.js SDK features




© 2021 Sitecore Corporation A/S.

# Next.js simplifies JSS development

## Next.js server is built for production pre-rendering

- No sample boilerplate for SSR or server/client state management
- “Headless SSR proxy” is not needed for production deployment

## Next.js has built in “preview mode” for editor integration

- No requirement to install node.js on Sitecore CM
- No need to deploy your JavaScript bundle to CM
- No extra setup required to use the “rendering host” / HTTP render engine – it’s the only way!

## Container-based getting started option is available

- Quick start for Windows-based and Sitecore-first development
- “jss create” with mock Sitecore services is still available as well

© 2021 Sitecore Corporation A/S.

SITECORE

# Why have we added support for Next.js

© 2021 Sitecore Corporation A/S.

# Motivation behind Next.js support

SITECORE

The **need** for **speed**.

# Motivation behind Next.js support


**JavaScript**
Dynamic functionalities are handled by JavaScript. There is no restriction on which framework or library you must use.




**APIs**
Server-side operations are abstracted into reusable APIs and accessed over HTTPS with JavaScript. These can be third party services or your custom function.


**Markup**
Websites are served as static HTML files. These can be generated from source files, such as Markdown, using a Static Site Generator.

© 2021 Sitecore Corporation A/S.

# Motivation behind Next.js support

J

A

M

"Jamstack" is **not an actual tech stack** that describes implementation technology

It's a high-level classification of an **architectural approach**.

**Jamstack anchors**

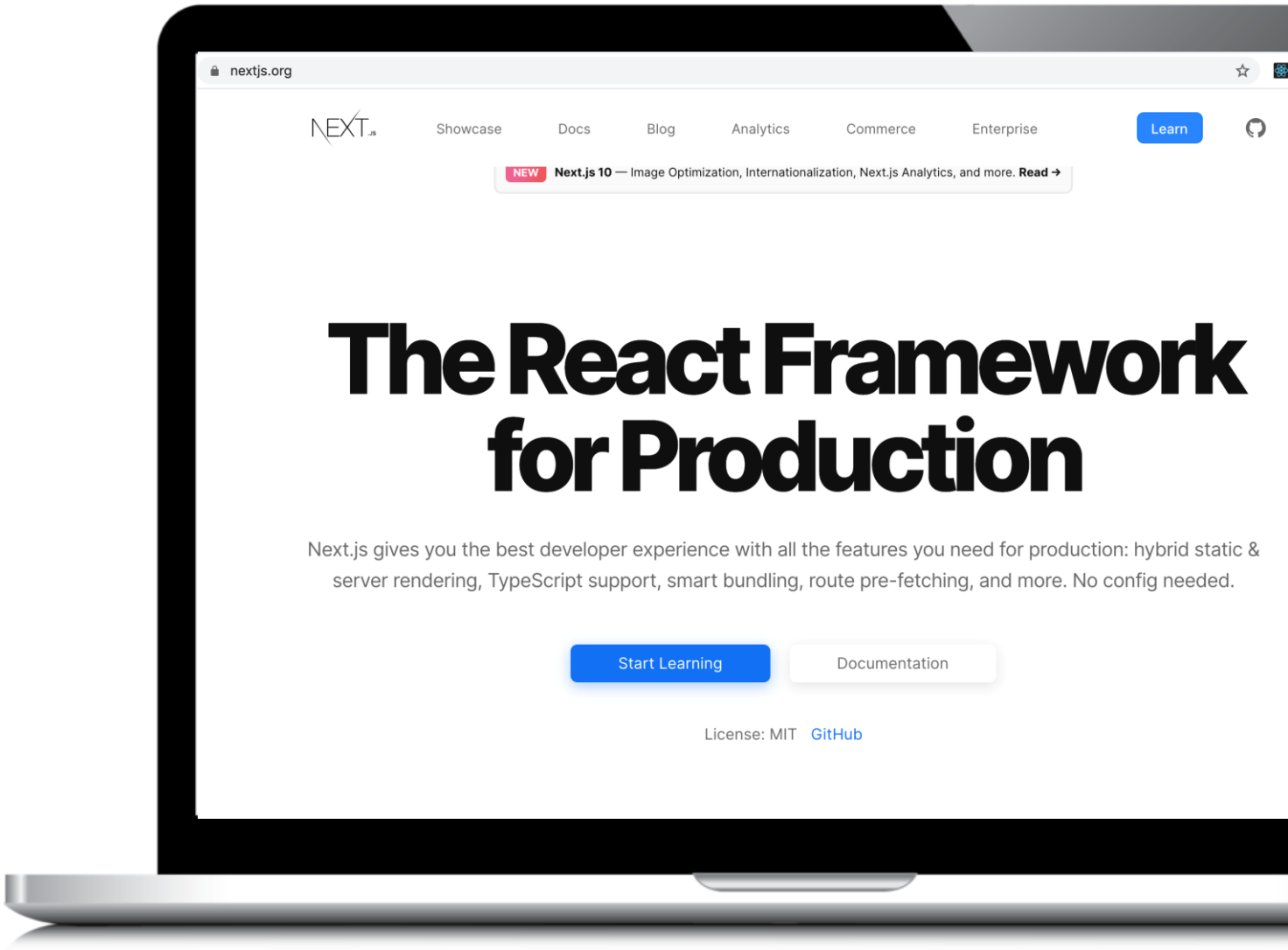
Pre-rendering

Fast delivery

Rehydration

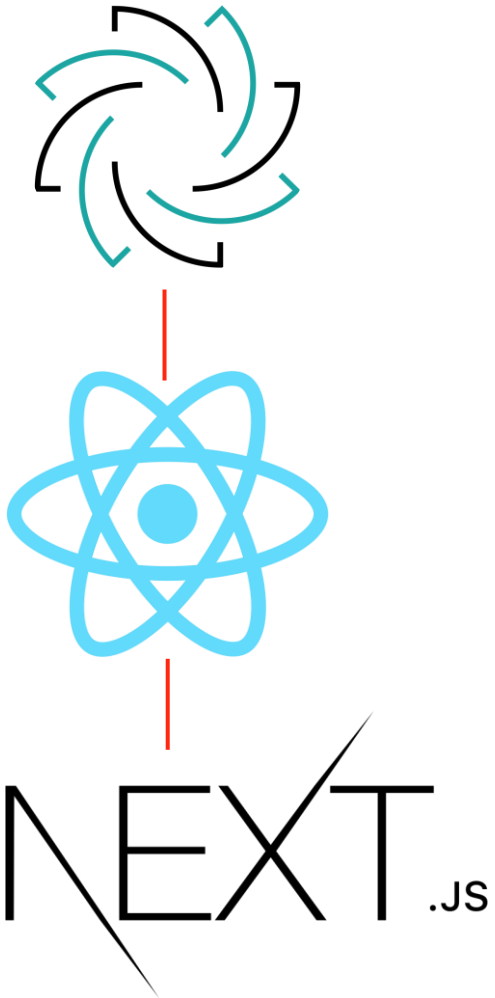
© 2021 Sitecore Corporation A/S.

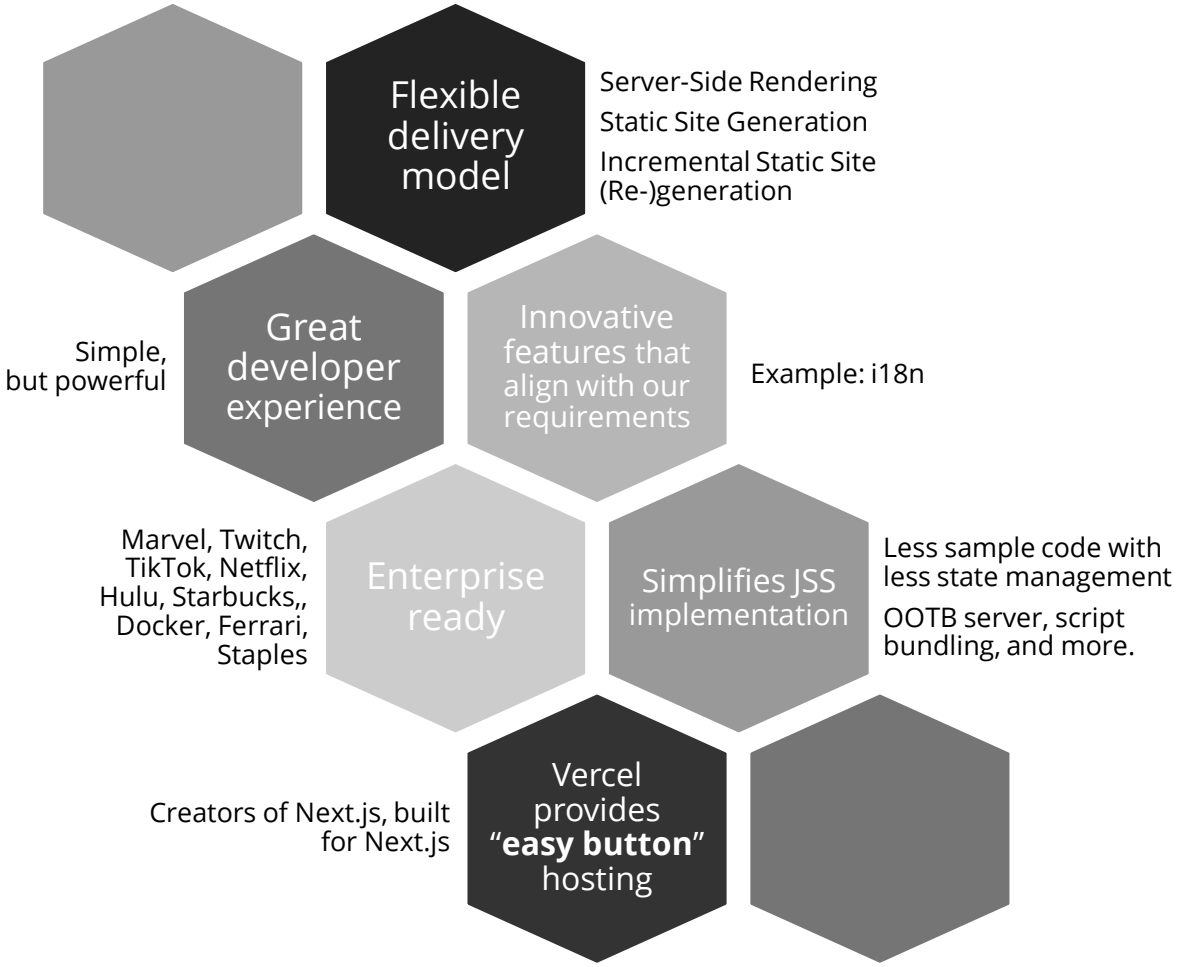
# Next.js Advantages




© 2021 Sitecore Corporation A/S.

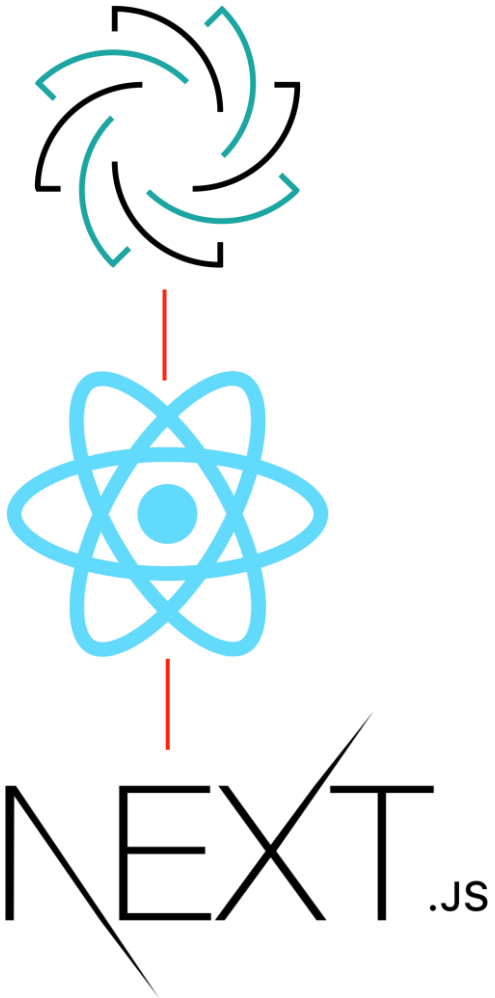
# Next.js Advantages

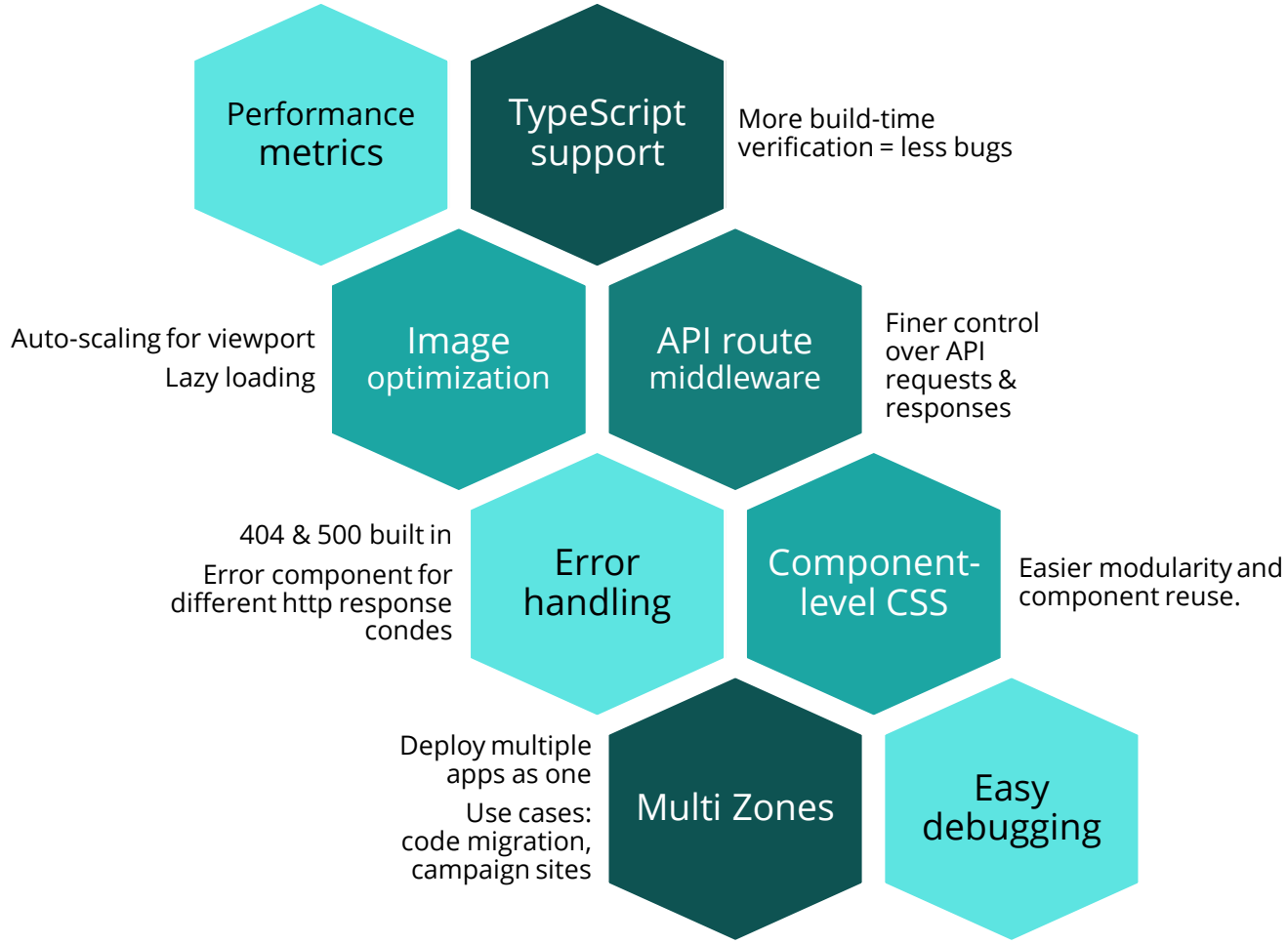



© 2021 Sitecore Corporation A/S.

# Next.js Advantages




© 2021 Sitecore Corporation A/S.

# The “M” in Jamstack – (static) markup

**What?**

**SSG** is the practice of **reducing or eliminating** the **processing of app-rendering code** at the time when the **end-user interacts with a page**.

**How?**

This is achieved by **pre-rendering server responses ahead of time**.

**Why?**

**SSG benefiits**

- excellent performance
- reduced load on back-end servers
- client-side app stability

© 2021 Sitecore Corporation A/S.

# Pre-rendering in Next.js

SITECORE

| Server-Side Rendering | Static Generation |
| --- | --- |
| **getServerSideProps**<br>Fetch data on each request | **getStaticProps**<br>Fetch data at build time<br><br>**getStaticPaths**<br>Pre-render dynamic routes based on data |

© 2021 Sitecore Corporation A/S.

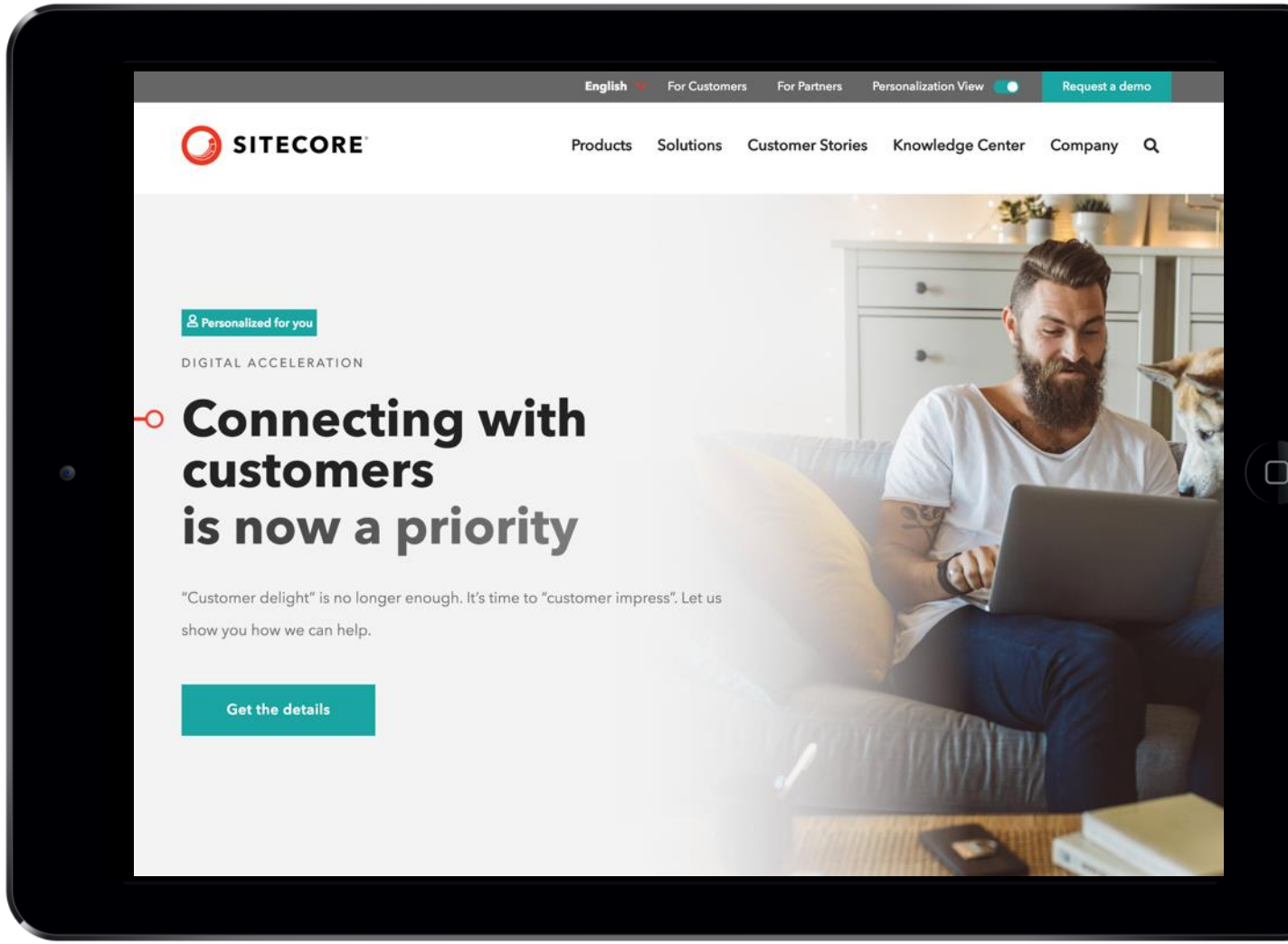
# Next.js Review

Next.js in JSS 16.0

Jamstack

- Pre-rendering
- Fast delivery
- Rehydration

Next.js SSG & SSR




© 2021 Sitecore Corporation A/S.

# Today's speaker

## Thomas Desmond

**Javascript Technical Evangelist, Sitecore**

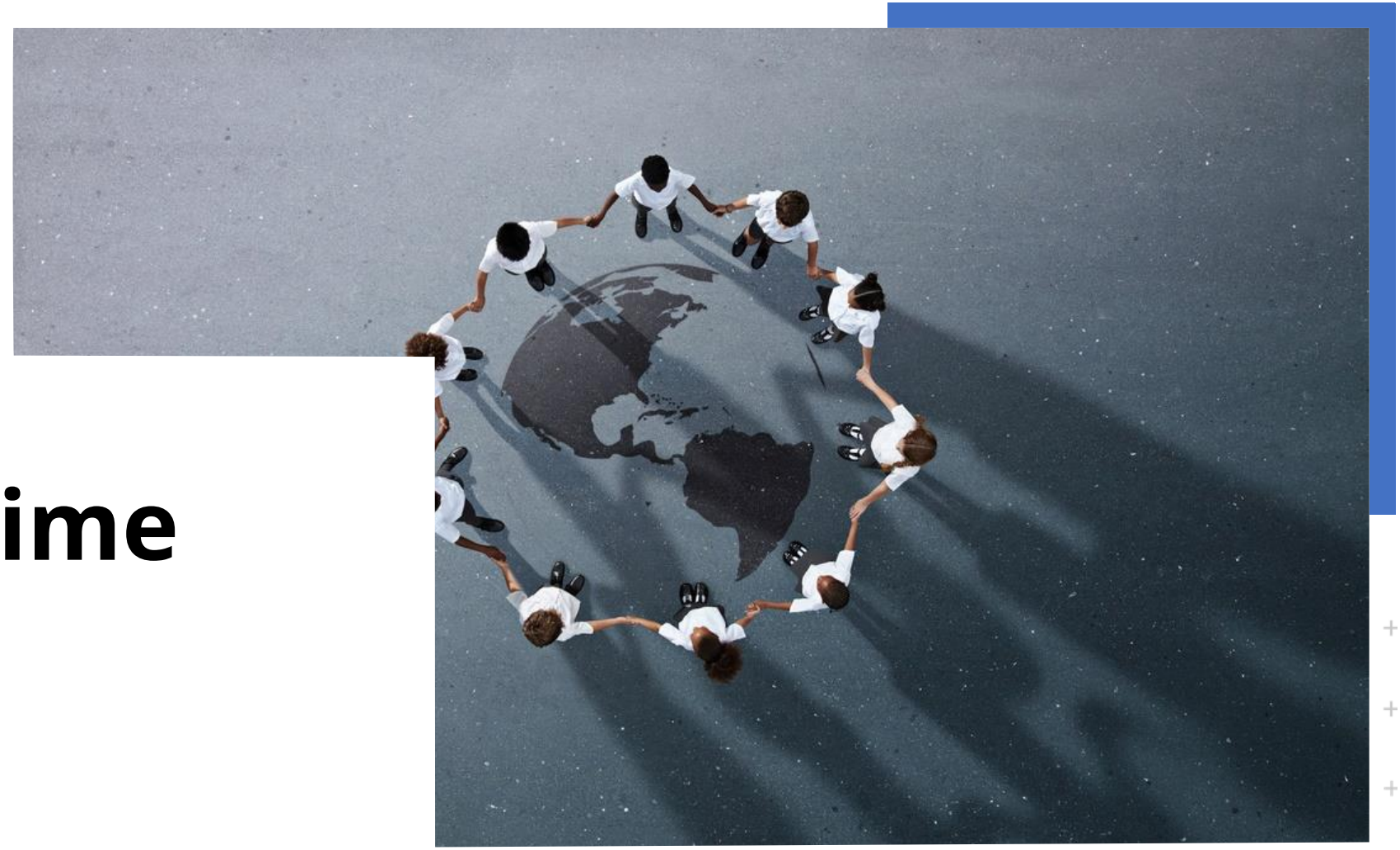
- Started at Sitecore in January
- Long time frontend developer
- Located in San Diego, California

@ThomasJDesmond

www.thetombomb.com

© 2021 Sitecore Corporation A/S.

SITECORE



# Demo Time

© 2021 Sitecore Corporation A/S.