

Engineering leaders and managers typically make decisions about how to build the sites and products that fall under their ownership. These decisions are based on things like cost, complexity, team needs, business requirements, and the success metrics they are accountable for.

These stakeholders need to be convinced that Gatsby can offer them the balance of efficiency, results, and low risk that will satisfy their team, their executives, and their users.

Some specific things that engineering leaders and managers care about include:

- **Performance and security** No one wants their product to be noticeably slow or the source of a major security vulnerability. Getting these two key features right can be complicated, even for experienced engineering teams.
- **Flexibility and scalability** Managers want to pick tools that can grow with their user base and with their long-term product roadmap. They do not want to choose a technology that will limit their growth or risk having to start over if (and when) that tool proves to be too rigid.
- **Not reinventing the wheel** While many managers appreciate and sympathize with their team's or colleagues' desire to try out cool new tech, they'd rather not spend valuable time and resources on a new thing when the job is already being done to their satisfaction. As the saying goes, "If it ain't broke, don't fix it."
- **Helping their team work smarter** New tech is interesting to managers when it removes obstacles for their team, helps them be more effective, and/or reduces [yak shaving](#). However, these benefits have to outweigh the cost of integrating a new tool and/or re-implementing an existing product.

Basic explanation

Here's an example of a basic explanation of Gatsby for engineering leaders and managers:

Gatsby is a free, open-source framework for building websites and applications. It's extremely developer-friendly and streamlines the setup and configuration of your build. Gatsby can pull data into your UI from any and all of the sources you currently use; and exceptional performance, added security, and current web best practices are built into Gatsby so your team can focus on delivering an exceptional experience for your users and customers.

Specific benefits

Gatsby has many benefits that will appeal to engineering leaders and help them meet their goals. Some specific talking points are listed below. (A lot of these benefits are also covered in Preston So's article [Enterprise Gatsby: How to Reduce Your Digital Total Cost of Ownership \(TCO\) with Gatsby](#), which is definitely worth sending to your engineering leaders.)

Performance

From its inception, Gatsby was built to optimize performance. Gatsby sites are consistently 2-3x faster than similar sites built with other tools, with pages loading in milliseconds rather than seconds. Gatsby's automated performance optimizations include pre-fetching resources, code splitting, progressive image loading, statically generating HTML, and Google's [PRPL Pattern](#). If you want to take a deeper dive into Gatsby's performance features, check out Kyle Mathews's article, [Web Performance 101 – also, why is Gatsby so fast?](#) or Dustin Schau's article, [Behind the Scenes: What makes Gatsby Great](#).

Security

Gatsby is a modern site generator that outputs static HTML pages at build time. That means you don't need a database or content management system running on your public servers, both of which are common sources of security breaches. Static content is much easier to secure. This does not, by any means, eliminate all security

vulnerabilities for Gatsby sites, but they will be significantly reduced. Learn more about Gatsby security in Alex Moon's article on [Security for Modern Web Frameworks](#) and the Docs page on [Answering IT & Security Questions](#).

Scalability

The scalability of your traffic handling, product complexity and features, and your development process will all benefit from Gatsby. Gatsby sites can be entirely deployed to a CDN, effectively eliminating your risk of downtime due to traffic spikes. If your site or application needs to support new functionality, Gatsby can pull content from as many sources as you need. With other tools, integrations with external resources are an afterthought and a common source of frustration for developers, causing project delays and outages. Gatsby streamlines the integration process, eliminating many of these headaches and points of failure. Finally, Gatsby equips dev teams to make their own components that can be reused across projects with minimal reworking, giving them more time to work on new features.

Lower costs

CDN hosting for static sites is much, much less expensive than traditional hosting costs. Read more about this here: [Enterprise Gatsby: How to Reduce Your Digital Total Cost of Ownership \(TCO\) with Gatsby](#). You'll also save money on team resources as your developers will spend much less time working on performance optimization and configuring integrations. Many managers have also found it easier and less expensive to recruit for Gatsby projects because they don't require specialized CMS skills or the advanced expertise needed to handle complex tooling and development environments.

Improved development process

Gatsby [unifies your stack](#) and eliminates a lot of the complicated and time-consuming configuration steps that come with most development processes. This means happier developers and clearer division between frontend and backend tasks. Frontend developers can spend more time making great UIs and backend developers can focus on adding the features and integrations that make your product better. You can find more information on this subject in Sam Bhagwat's article [How Gatsby Changes Teams' Website Development Workflow](#)

Common concerns

You may find that your engineering leaders and managers have some concerns about Gatsby. Here are some common examples and responses you can provide to help alleviate those concerns.

Support for open source software is often unreliable and relatively few projects offer long-term viability. How is Gatsby different?

Gatsby has an extremely active and communicative community with over 2,000 contributors and tens of thousands of developers building Gatsby sites. The process for filing issues and [contributing to Gatsby](#) is well-documented. [Gatsby's extensive documentation](#) also includes various guides, tutorials, plugin and starter libraries, troubleshooting, and additional resources. Last but not least, the Gatsby open source project is supported by [Gatsby, Inc.](#) whose team is committed, full-time, to developing and maintaining Gatsby. Gatsby, Inc. also offers support services, training, and partnership opportunities.

Our site uses a lot of dynamic content and components. Didn't you say Gatsby creates static sites?

Yes, but there's more to it. Gatsby statically generates HTML content using React DOM and server-side APIs – it's an important part of how Gatsby delivers exceptional speed and better security. However, this static HTML content can

then be enhanced with client-side JavaScript via React hydration. You can learn more about this in Dustin Schau's blog post [Beyond Static: Building Dynamic apps with Gatsby](#).

Our content team and other less-technical members of my organization need to be able to make site updates. Are they going to be able to work with a Gatsby site?

Gatsby can pull content from any source, including all-in-one CMSs, like WordPress and Drupal, and headless CMSs. Your content team can work with whichever content editor suits their needs and preferences, including the CMS they're already using. You can also enable more collaboration between your content and development teams with [Gatsby Preview](#).

Case studies

Case studies are an excellent way to build an engineering leader's confidence in Gatsby's capabilities. Here are some case studies you might want to share:

- [Beyond Static: Haptic Media Uses Gatsby to Build a Dynamic Web App](#)
- [IBM Uses Gatsby to Manage Enterprise-Level Content](#)
- [How we're migrating a government open data site to Gatsby](#)

For examples of Gatsby sites, [check out the Showcase](#).

Conclusion

When you're trying to win over a stakeholder, your first instinct may be to focus completely on Gatsby's benefits. However, when you're speaking to engineering leaders and managers, it's important to be positive while still being honest about Gatsby's limitations. Setting expectations before they go out to do their own research will help them focus on finding solutions rather than faults. There is ample proof that Gatsby works well for many websites - don't be afraid to help your engineering leaders dig into pros and cons. (And it doesn't hurt to get a few of their team members on your side. Check out the [Winning Over Developers](#) page for more on that subject.)