

The State of Jamstack 2021 Report.

In partnership with

Luminary

Table of contents.

About Kontent & About this report.	3
Introduction.	4
The state of the Jamstack in 2021.	7
The Jamstack survey results.	13
Conclusion.	32
Overview of our respondents.	33

About Kontent.

Kontent by Kentico is the world's leading cloud-based headless CMS platform. Robust APIs, webhooks, and flexible content models give developers full control over the structure of their content and how it's delivered to their websites and applications. The APIs deliver clean, structured content that can be pulled into any experience and technology stack.

On top of this headless foundation, Kontent includes an intuitive authoring experience that lets marketers focus on content creation and business workflows without requiring developer involvement. This frees developers to focus on what they do best—building engaging online experiences that look great in every channel.

Companies across a wide range of industries have built their online experiences on top of Kontent, including Vogue, PPG, Oxford University, AC Milan, and Konica Minolta. Additionally, Kontent by Kentico is regularly included in key industry reports published by Gartner, Forrester Research, and G2.

About this report.

Last year, we conducted a survey among 530 developers in four countries to find out what they thought of the Jamstack, a modern web development architecture. This year, we decided to go bigger and surveyed more than **750 developers in Europe, North America, Australia, and Asia** to see how their opinions have changed since 2020.

Are developers still choosing the Jamstack? Are they using the same tools as last year? Keep reading to find out that and other interesting facts, such as:

- How long developers around the world have been working with this architecture
- Which technologies they use, and where they host their Jamstack projects
- Which static site generator is the most popular one in 2021
- What developers love and hate about today's Jamstack tooling

This survey has evolved into a fun annual tradition, so we hope you'll enjoy this report as much as we did preparing it. Let us know what you think on [Twitter](#), [LinkedIn](#), or [Facebook](#)!



Introduction.

“

First they ignore you, then they laugh at you, then they fight you, then you win.”

— Mahatma Gandhi

Last year, I wrote about how the traditional way of building sites is expensive, requires a lot of maintenance effort, and always ends up in caching for the sake of performance optimizations. Back then, the Jamstack was not that widely used, and it seemed like all it brings are benefits. That attracted many developers, including those working on large projects, and we started to see some of the benefits turn into drawbacks.

With the growing amount of content, we've experienced not only long production builds but also decreased local development speed. Oftentimes, we had to implement some typical server-side tasks like personalization in client-side JavaScript or manually extract dynamic functionality into serverless functions, which further complicated the CI & CD pipelines. Some integration implementations or webhook processing forced us into creating single-purpose proxy functions. And even though we were able to get it all work, there was always the outstanding question: “Does this feel like the right thing to do?”

But the Jamstack frameworks and the Jamstack itself evolved too. Next.js allows us to configure how each part of our site should be rendered. Picture an e-commerce solution where it's now possible to pre-generate the main site and often visited categories and leave the product pages to a serverless function that is executed on-demand and caches each generated page. Gatsby now features fast incremental builds and tailored hosting service Gatsby Cloud that helps with quick content previews. We see increased interest from .NET developers as the static site generator Statiq evolves and allows to generate not only HTML but literally any type of output.

The Jamstack is no longer an alternative but a competitor to traditional websites. A friend of mine, Andy Thompson, wrote in [his article](#), comparing the headless and traditional CMS from the SEO perspective, that using headless is like driving a car



with manual transmission. If you know how to handle it, you'll be faster. I like that, and it seems we're getting to the same point with the Jamstack. It brings the development experience to a whole new level, as it allows us to create simple sites that are hosted for free but also gives us options for site growth.

The future of the Jamstack.

Some Fortune 500 companies have already successfully adopted the Jamstack, and I believe we will see this trend increase and make its way into the mid-market too. These days, we're noticing a rise in new microservices for personalization, e-commerce, recommendations, search, form submissions, and other typical features of mid-to-large size websites. Vendors are putting a lot of effort into making their integrations as seamless as possible. Hosting platforms are adjusting their offerings to support all new features of the ever-changing Jamstack frameworks. All this points to the prediction that more developers will adopt the Jamstack as their default choice of building websites.

From the technical perspective, I believe optimizations will be the largest topic. We are already seeing attempts at CDN-level caching and bringing server-side processing to the edge, as close to the visitors as possible. But we'll also witness improvements on the client-side. With the new web vitals and increased focus on accessibility, the frameworks will try to optimize the size of the production bundles. And we will hopefully see more packages and Jamstack frameworks that fully support native ESM.

To understand the quote by Gandhi at the beginning of this introduction, take a look at the report and find out how developers see the Jamstack today. I think the Jamstack is now somewhere between fighting and winning.



Ondrej Polesny.
Developer Evangelist, Kontent by Kentico



About Luminary.

Luminary is a global leader in the implementation of Kentico and the premier Kentico agency in Australia. Led by CTO and Kentico MVP Andy Thompson, Luminary's development team directly supported Kentico in building the initial boilerplate for Kentico and continues to be involved in testing and refining the platform. Luminary Technical Lead Emmanuel Tissera also launched the world's first Kentico website.

Luminary is home to the largest and most experienced Kentico development team in Australia, across both Kentico and Xperience by Kentico. One of Australia's longest-standing independent digital agencies, Luminary's longevity and depth of experience with Kentico has earned it a reputation as the agency of choice for many of the biggest names in the Australian government, retail, and corporate sectors.

Luminary

The state of the Jamstack in 2021.

For the second year running, Luminary has partnered with Kontent to produce The State of the Jamstack report. Here are our thoughts one year on.

A few years ago, [Jamstack was something of a buzzword](#). One short year ago, we wrote that it was [time for the Jamstack to shine](#). A lot can change in twelve months! Let's take a look at the state of the Jamstack in 2021.

Jamstack for Enterprise.

Increasingly, digital teams of all shapes and sizes are migrating toward the Jamstack as a web technology platform. While it's been popular for smaller websites for years now, Enterprise-level decision-makers such as CTOs, CIOs, and CMOs are now coming to us with a microservices + Jamstack architecture in mind.

The Jamstack consists of three key elements: JavaScript, APIs, and Markup. Often overlooked is the API component, which represents the rich ecosystem of SaaS platforms, third-party integrations, and custom APIs behind any Jamstack project of reasonable complexity. This is even more true for Enterprise users who typically have varied and complex integration requirements beyond simple web publishing.

In the past year, we've also seen a group of Enterprise technology vendors band together to get behind a new acronym and variation on the Jamstack: [MACH](#).

MACH is short for Microservices, API-first, Cloud-native, and Headless. This architecture approach is very similar to and espouses most of the same benefits as the Jamstack, but it places a greater emphasis on these critical services providing the grunt behind the Markup and JavaScript, particularly cloud-first Headless CMS or Content-as-a-Service providers.



A year of trade-offs.

In recent years, the ‘6 Ss’—[Speed](#), [Stability](#), [Scalability](#), [Security](#), [Serviceability](#), and [Simplicity](#)—have been compelling reasons, or benefits, for organizations to consider the Jamstack as an Enterprise platform. However, it has almost always involved a number of trade-offs in these areas, often due to the reliance on these ‘behind-the-scenes’ APIs and third-party services.

For example, the scalability benefits of being able to instantly scale and respond to massive spikes in traffic versus the bottlenecks experienced when scaling up a site’s content library from hundreds to thousands or tens of thousands of pages. Or the lightning speed page loads of a statically-generated website versus the necessary delay in publishing while that static website is re-generated and re-deployed.

Maturing services.

While the intrinsic benefits of the Jamstack remain as strong as ever, it is in the innovation and evolution of these ‘behind-the-scenes’ APIs and supporting services that we have seen the biggest leaps forward in the past year.

A couple of years ago, hosting a Jamstack site meant simply dropping the static files on any cheap hosting platform you like. Today, there are emerging leaders in fit-for-purpose Jamstack DevOps and hosting platforms that handle everything from build and deploy through to high-performance hosting and even serverless functions for your custom APIs. Niche providers such as [Netlify](#), [Vercel](#), and [Gatsby Cloud](#), and even the biggest names in town are getting on board, such as Microsoft with its [Azure Static Web Apps](#) offering and its deep integration with [GitHub Actions](#).

Similarly, more and more integrated services are treating Jamstack as a first-class citizen by going ‘API-first’, including long-term holdouts such as e-commerce providers and even traditional, monolithic CMS platforms.

Revisiting the ‘6 Ss’.

While they were already key strengths of the Jamstack as a platform, there’s always room for improvement! A year later, the biggest leaps forward have been in the areas of serviceability, speed, and simplicity.

Serviceability.

Serviceability refers to the degree to which the servicing of an item can be accomplished with given resources and within a specified timeframe. The Jamstack improves serviceability by essentially opening up the option of selecting ‘best-of-breed’ tools and services, so each individual task that a content editor, developer, marketer, or product manager might need to perform can be done in the best tool for the job.

This disconnect between services can tend to introduce some disadvantages, however. For example, an ‘all-in-one’ system such as Digital Experience Platform can potentially provide a more unified and easy-to-learn or use interface, compared to a ‘suite’ of separate tools.

Headless CMS or Content-as-a-Service platforms have seen huge improvements in serviceability for Jamstack websites in the last year. Where before using a Headless CMS could mean having to tolerate being disconnected from your web channel and having to hope things looked right when you published them, the leading vendors are now offering impressive features that bridge the gap between headless and traditional content management, such as real-time web page preview and on-page editing.

Simplicity.

In previous years, producing a Jamstack website with all the options available could mean cobbling together a disparate set of potentially confusing tools and being left with something of a ‘house of cards’ architecture that could fall apart if any of its components ran into trouble. Now there are a number of fit-for-purpose, dedicated frameworks and services specifically for developing, building, testing, deploying, hosting, and monitoring Jamstack websites with Enterprise levels of security, performance, support, and SLAs.

Integration scenarios can also get extremely complex and fragile in a microservices world, with data flow diagrams starting to resemble a plate of spaghetti. While integration options are unlimited once every service exposes fully-featured APIs, it can also mean that Solution Architects are required to design and maintain a large number of integrations. Now, software vendors are increasingly supporting ‘out-of-the-box’ integration scenarios with each other or intermediary integration platforms such as [Zapier](#), [Azure Logic Apps](#), or [Pipedream](#).

Speed.

Site speed is unquestionably a major advantage of the Jamstack architecture. However, to get those instant page loads and incredibly quick page transitions, there is always a trade-off to be made. Commonly, this comes in the form of a publishing delay, while you wait for your Static Site Generator (SSG) to be rebuilt and deployed after content has changed in your CMS or one of your other connected data sources. Or if you’re using a Single Page App (SPA) that needs to load and render quite a bit on initial page load, you could be running into issues with Google’s Core Web Vitals metrics.

This year, we’ve seen major releases from leading SSGs Gatsby and Next.js targeting drastically decreased site generation time. Tight integration with CI platforms such as Netlify and Vercel allow for smart caching and incremental page generation, which can get publishing times down to mere seconds, making them as fast or even faster than traditional CMS publishing.

Frameworks such as Next.js support a hybrid of static and dynamic site generation, allowing you to run a real-time secure preview site linked to your CMS but also generate a static site for your production environment that comes with all the benefits of the Jamstack, giving you the best of both worlds.

Enhancements in the actual Jamstack front-end rendering frameworks in use are also unlocking some of the latest techniques and modern browsers, such as dynamic component import and native lazy loading.

A mature platform.

With the advancements and innovation in the Jamstack ecosystem in the past year, we strongly believe that the Jamstack is ready for Enterprise. It's now over to Enterprise organizations to ask themselves if they are ready. A move to the Jamstack requires change across the organization, and organizations need to take this change seriously or risk missing out on some, or all, of the benefits that come with an effective Jamstack architecture.



Adam Griffith.
Managing Director, Luminary



Andy Thompson.
CTO, Luminary



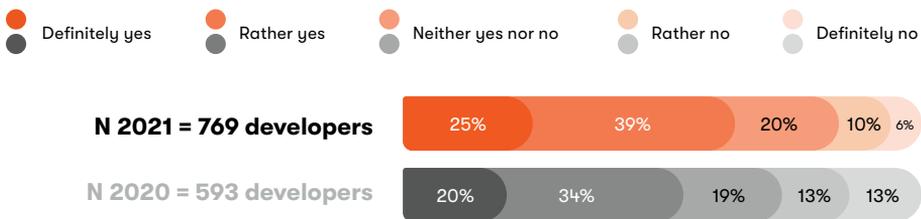
The Jamstack survey results.

The Jamstack is a modern web development architecture based on pre-generated HTML markup where dynamic and interactive functionalities are handled by JavaScript, and server-side operations are abstracted into reusable APIs. “JAM” = “JavaScript, APIs, Markup”

1 Are you familiar with this definition of the Jamstack?



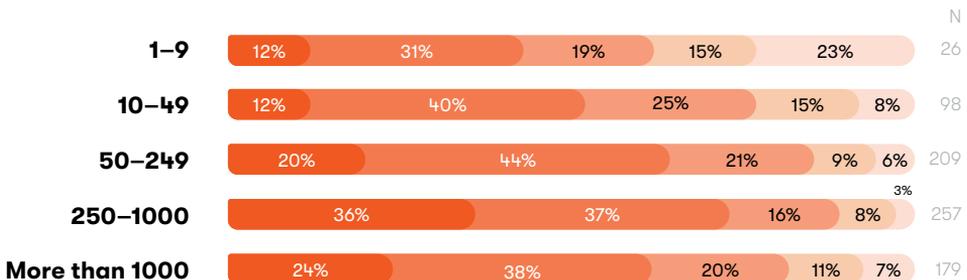
The respondents could select only one answer.



We can gain valuable insights if we compare results from our survey last year (531 respondents) with data from 2021 (769 respondents). 64% of developers, which is 10% more than last year, are familiar with the given definition of the Jamstack.

Interestingly, the larger the company is, the more familiar the developers seem to be with the definition.

Company size (employees)



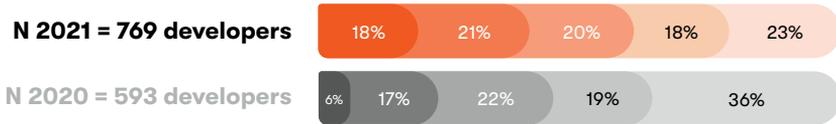
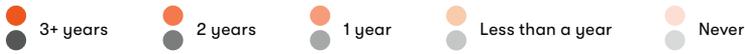
N = 769 developers



2 How long have you been working with the Jamstack?



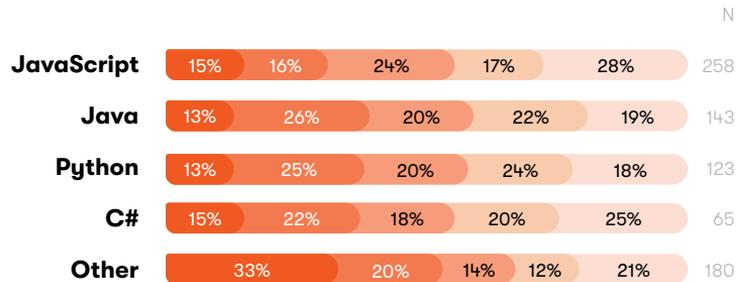
The respondents could select only one answer.



77% of respondents have worked with the Jamstack; 59% have worked with it for one year or longer. One-quarter of respondents do not have any experience with the Jamstack.

Respondents from Europe and Australia have been using this modern architecture the longest, together with respondents aged 35 to 44 and those from larger companies (250 to 1,000 employees—see the next page).

Primary programming language (developers)

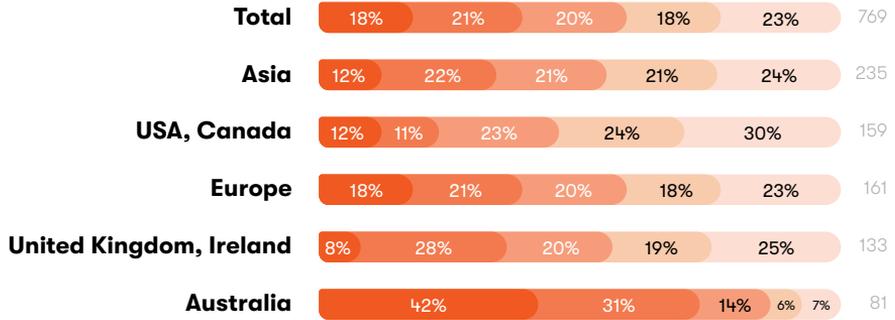


N = 769 developers



● 3+ years
 ● 2 years
 ● 1 year
 ● Less than a year
 ● Never

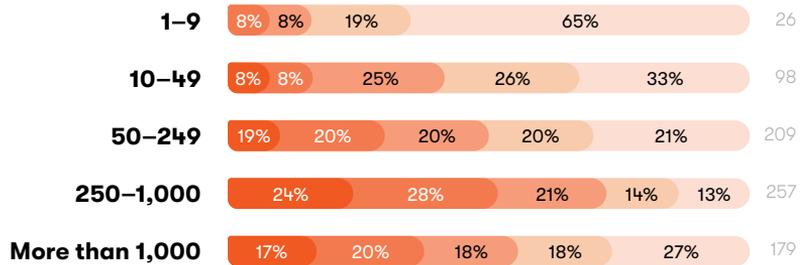
Country



Age



Company size (employees)



N = 769 developers



“

Jamstack is becoming synonymous with the modern web. The core concepts of decoupling the front end from the back end and making it easy for developers to reason about the state of their applications remain truer than ever. New build strategies and compute primitives are unlocking extremely large-scale sites and hyper-dynamic, real-time web applications. The future is bright for Jamstack development teams, and the companies that deploy them will have larger competitive advantages than ever.”

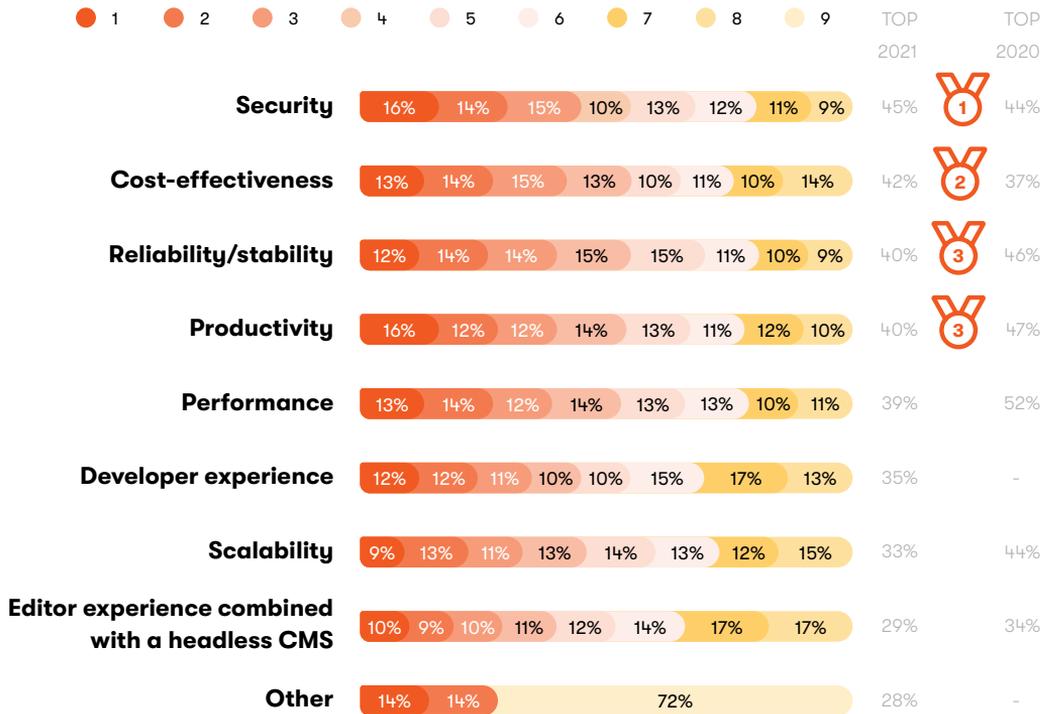


Chris Bach.
Co-Founder, Netlify

3 What are most likely the reasons for choosing the Jamstack to build an app with?



Respondents were asked to put the items in order (the most likely reasons at the top, the least likely reasons at the bottom of the list).



N = 592 developers working with the Jamstack

For 45% of developers, security is the #1 reason for choosing the Jamstack, followed by cost-effectiveness and reliability. On the other hand, the editor experience combined with a headless CMS is the least important reason for using the architecture.



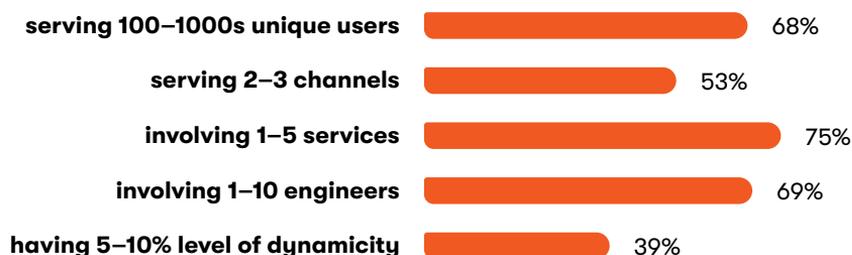


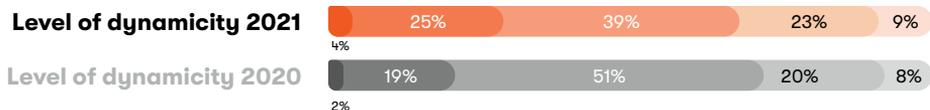
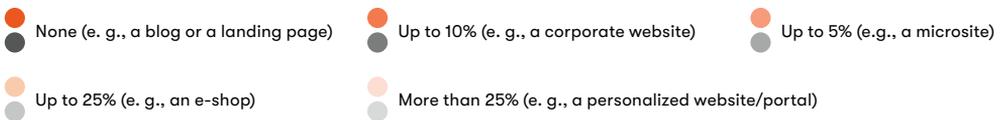
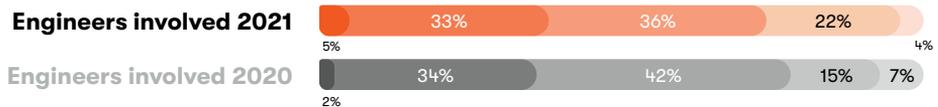
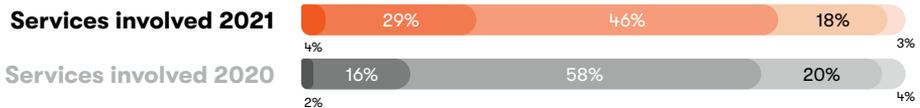
4 Think of the most complex project you've built with the Jamstack. Try to describe its complexity (choose one item in each line):

Respondents could select only one item in each line.

Unique users served	10s	100s	1,000s	100,000s	1,000,000s
Channels served	1 e.g., web	2–3 e.g., web + mobile + social media	> 3 e.g., web, mob, social, chatbot, IoT, AR/VR, digital signage		
Services involved (integrations/microservices/serverless functions)	0	1	2–5	> 5	> 10
Engineers involved	1	1–4	5–10	> 10	> 20
Level of dynamicity	None e.g., a blog or a landing page	Up to 5% e.g., a microsite	Up to 10% e.g., a corporate website	Up to 25% e.g., an e-shop	More than 25% e.g., a personalized website/portal

The majority of developers working with the Jamstack describe their most complex projects as:





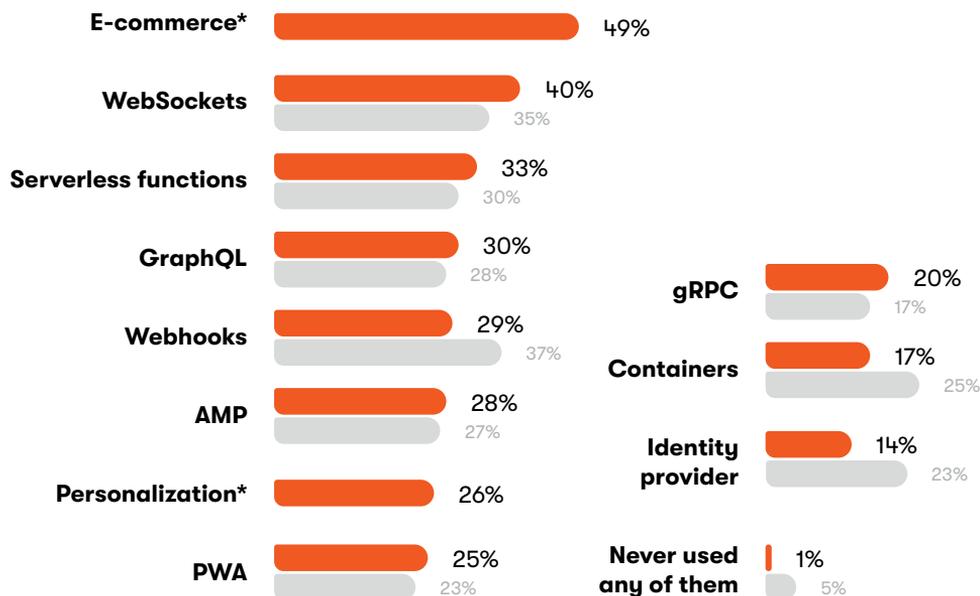
N (2021) = 592 developers working with the Jamstack
 N (2020) = 339 developers working with the Jamstack



5 Please, select the technologies you've ever used in a Jamstack project.

Respondents could select more than one answer.

*this option was not offered in the survey last year



N (2021) = 592 developers working with the Jamstack
 N (2020) = 339 developers working with the Jamstack

One-half of respondents working with the Jamstack have used e-commerce in their Jamstack projects. E-commerce technology is most frequently used in Asia (60% of respondents there) and also in the Trade & production (54%) and Marketing & media (56%) industries.

40% of developers have utilized WebSockets, 33% have worked with Serverless functions, and 30% have used GraphQL. The least needed technologies in Jamstack projects are containers (17%) and identity providers (only 14%).



“

Over the past couple of years, two things have changed how we develop websites and apps. JavaScript won the web, becoming the default programming language for front-end developers. Performance is no longer an afterthought. Sites must be built with Core Web Vitals in mind, as these metrics have a significant impact on Google search ranking.

Jamstack is not only gaining traction in the enterprise, it's also becoming the default. This approach allows developers to choose the JavaScript framework that best fits their requirements. It offers the blazing fast app delivery, reduced infrastructure costs, increased developer productivity, minimized security vulnerabilities, and 100% site up-time.

Uniform.dev, in combination with our technology partners like Kontent by Kentico, is delighted to offer headless personalization and testing that is made for Jamstack. Uniform powers the fastest possible personalized experiences, delivered on your edge of choice.”

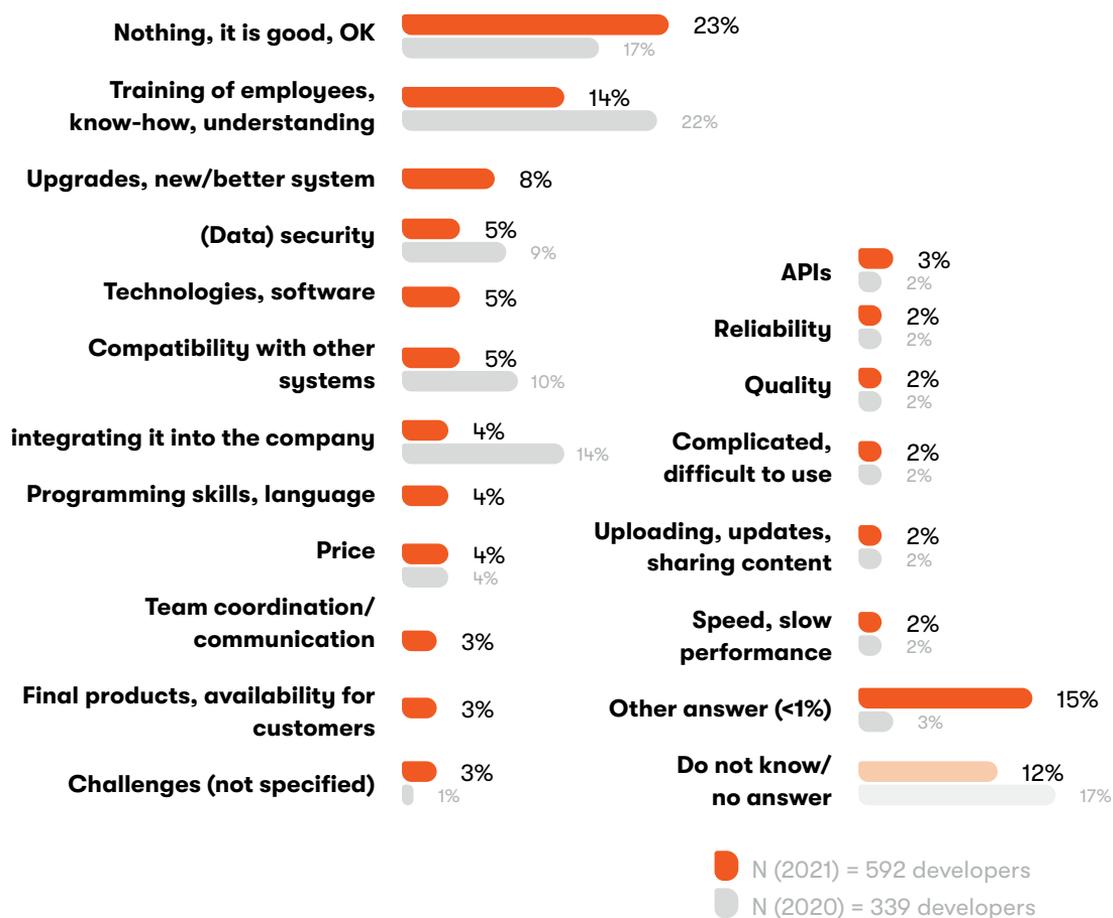


Lars Peterson.
Co-Founder, Uniform



6 Considering your experience, where do the biggest challenges in adopting the Jamstack architecture lie? Try to think about all three aspects—JavaScript, APIs, Markup.

Respondents could select more than one answer.



Considering their past experience and all three aspects of the Jamstack, respondents perceive understanding, knowing how to work with the Jamstack, and training new employees as the greatest challenges when entering the Jamstack world.

Almost one-quarter of respondents claimed that there are no serious challenges in adopting the Jamstack architecture.

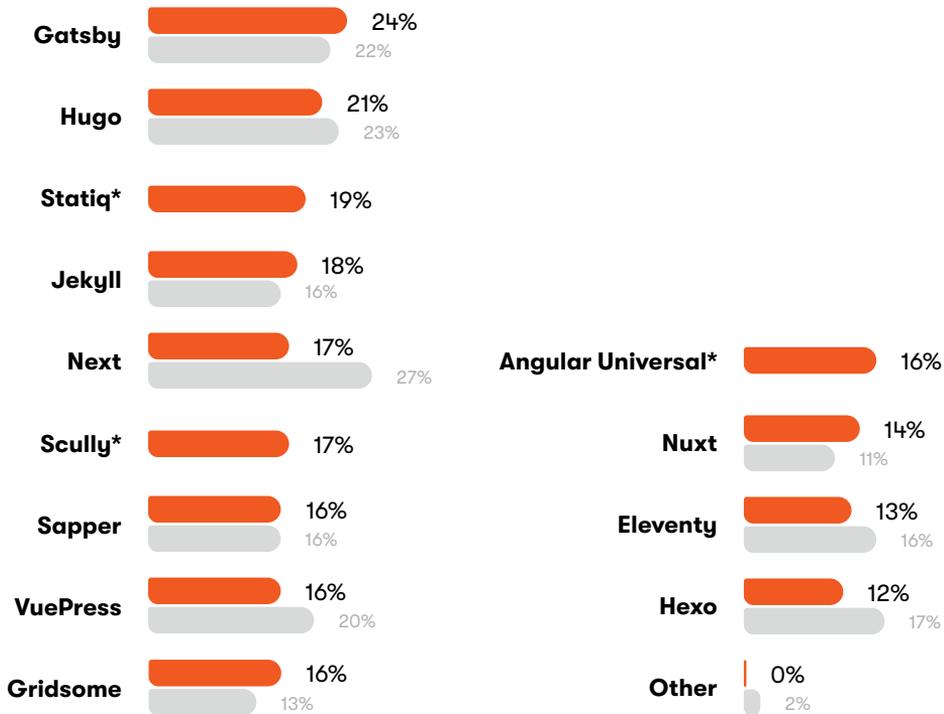


7 Which static site generators do you like to develop with?



Respondents could select up to three answers.

*this option was not offered in the survey last year



■ N (2021) = 592 developers working with the Jamstack
■ N (2020) = 339 developers working with the Jamstack

Gatsby remains the most popular static site generator (24%). Its friendly learning curve, thanks to GraphQL and quick and simple deployment, makes it a great fit, especially for smaller businesses.

Statiq is the #3 generator (19%), which shows that the Jamstack is not only about JavaScript—there is a lot of .NET developers interested in this architecture too.

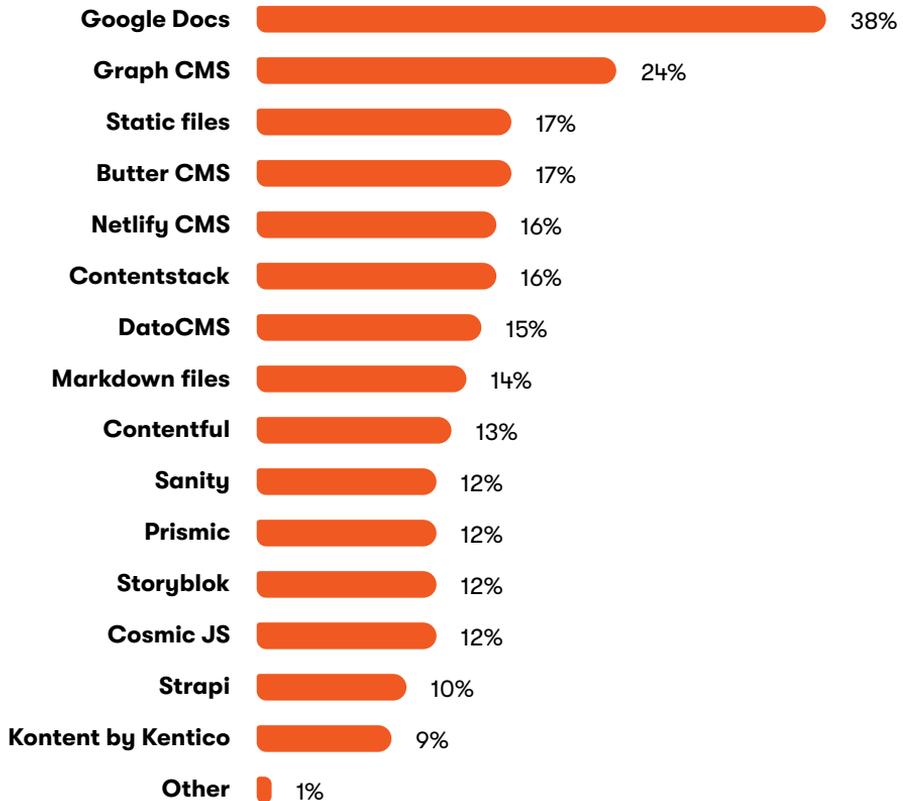
Next (17%) and VuePress (16%) have lost popularity over the last year.



8 Where do you prefer to keep your content?



Respondents could select up to three answers.



N = 592 developers working with the Jamstack

In 2020, 43% of respondents claimed they stored their content in Google Docs. As could be expected, more than one-third of developers (38%) who work with the Jamstack today still prefer to use Google Docs.

A lot of developers also like to use Graph CMS (24%) for their projects, particularly those in Asia, in midsize and large companies, and in the IT industry.

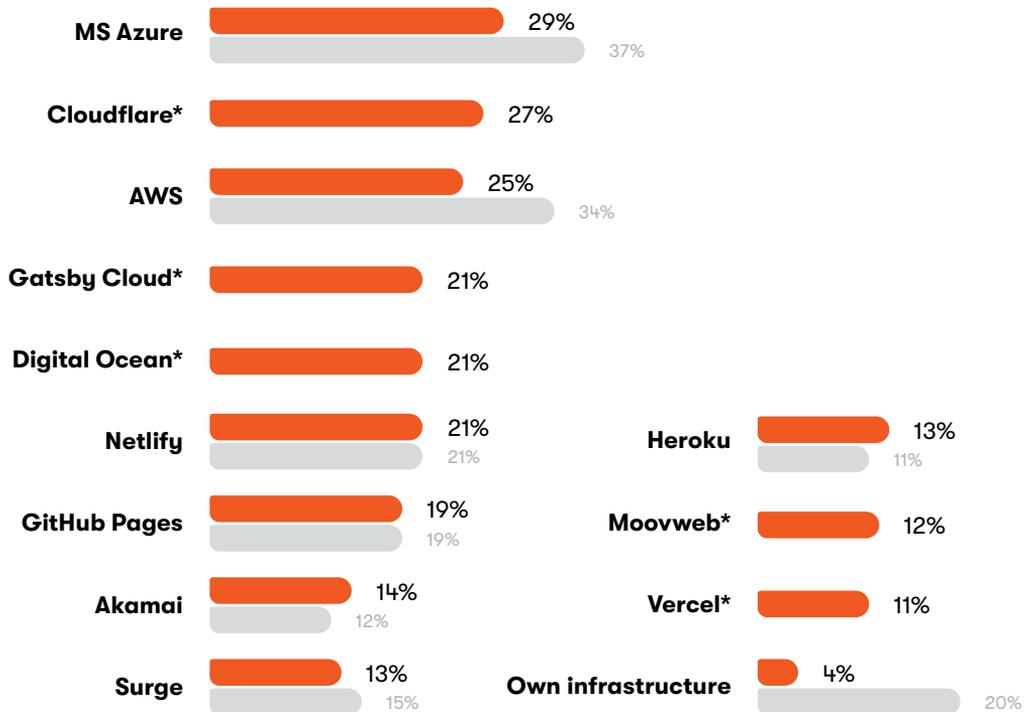


9 Where do you typically deploy and host your projects?



Respondents could select up to three answers.

*this option was not offered in the survey last year



■ N (2021) = 592 developers working with the Jamstack
■ N (2020) = 339 developers working with the Jamstack

The majority of respondents typically deploy and host their projects on MS Azure (29%), Cloudflare (27%), and AWS (25%). MS Azure and AWS were the most popular options in 2020 too.

Only 4% say that they use their own infrastructure for project deployment and hosting. Compared to 20% in 2020, this indicates a trend in abandoning the self-hosting strategy and a shift toward the use of external tools for project deployment and hosting.

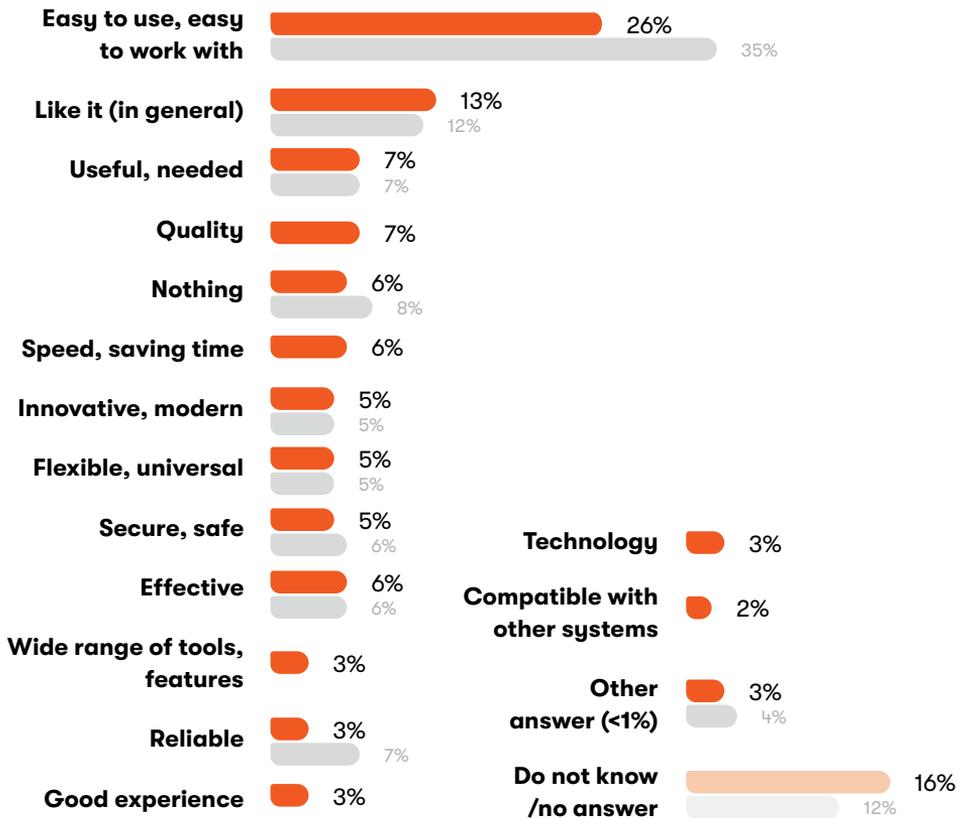


Think about the tools you've worked with—Static Site Generators, Headless CMSs, and Deployment Providers. Consider things like developer experience, learning curve, productivity, reliability, collaboration features.

10 What do you love about today's Jamstack tooling?



Respondents could provide more than one answer.



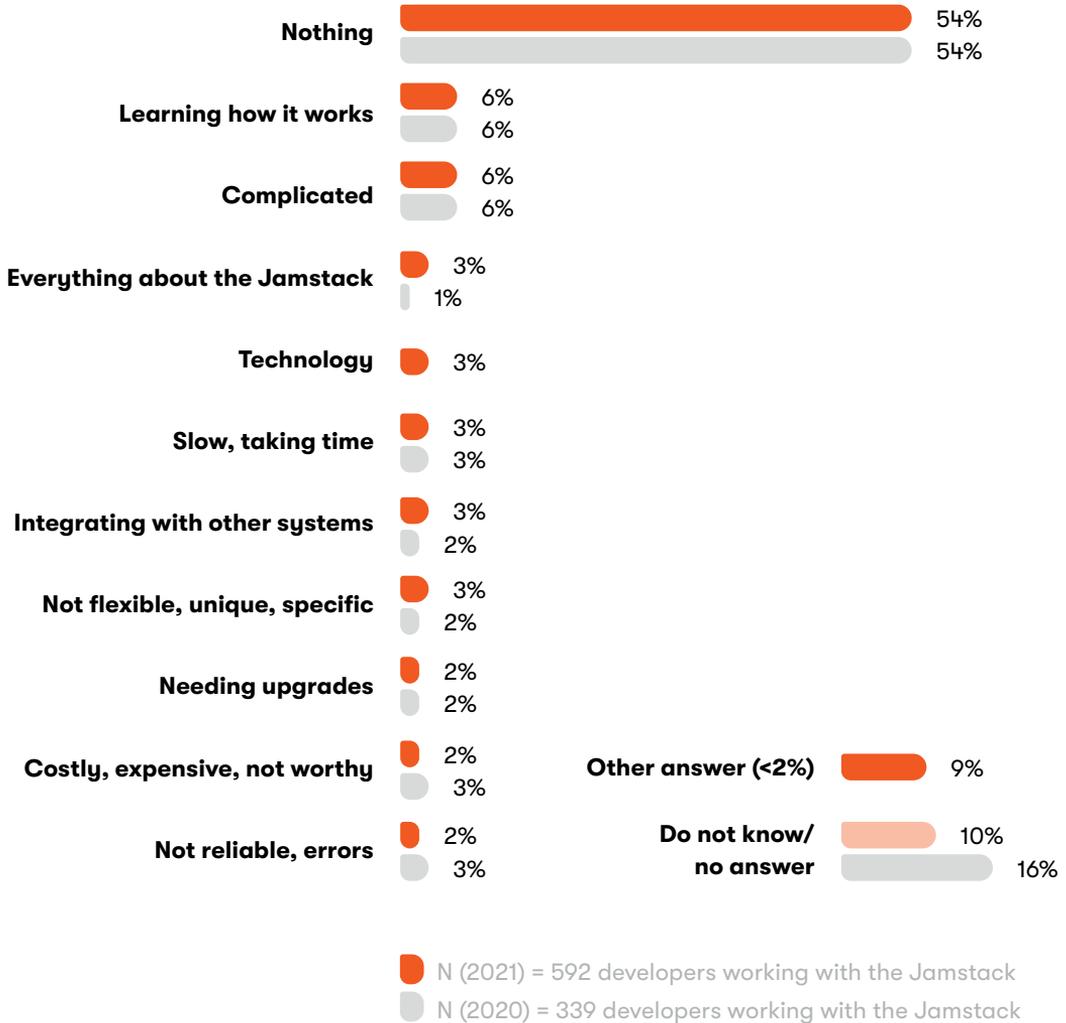
■ N (2021) = 592 developers working with the Jamstack
■ N (2020) = 339 developers working with the Jamstack

26% of respondents appreciate that the Jamstack is easy to use, and 13% like it in general.

11 What do you hate about today's Jamstack tooling?



Respondents could provide more than one answer.



The most problematic aspect of Jamstack tooling seems to be learning how it works (6% of respondents); some think that this modern way of building websites and apps is complicated (also 6%).

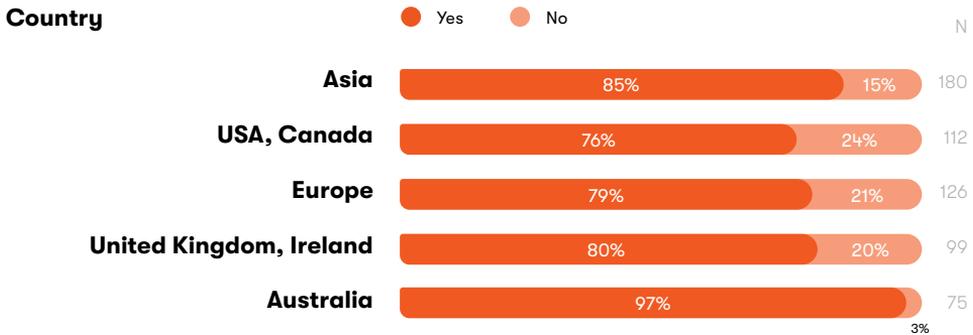
12 Are you interested in initiatives like Stackbit and Web Spotlight that provide a real-time editing experience for content editors on Jamstack sites?



Respondents could select only one answer.



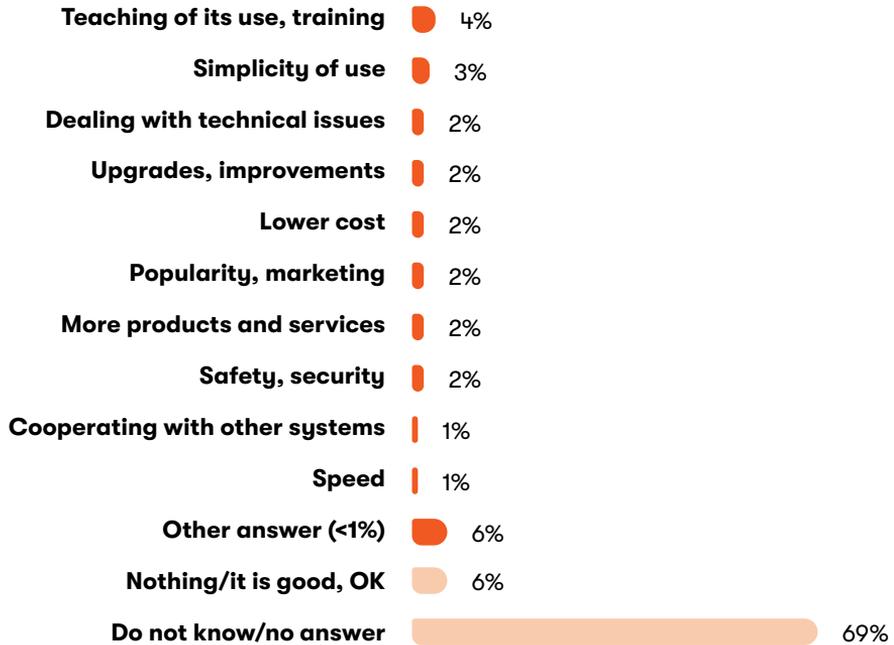
83% of developers working with the Jamstack are interested in initiatives that provide a real-time editing experience for content editors on Jamstack sites, such as Stackbit and Web Spotlight.



13 Name one thing that could help increase the Jamstack adoption.



Respondents could provide more than one answer.



N = 769 developers

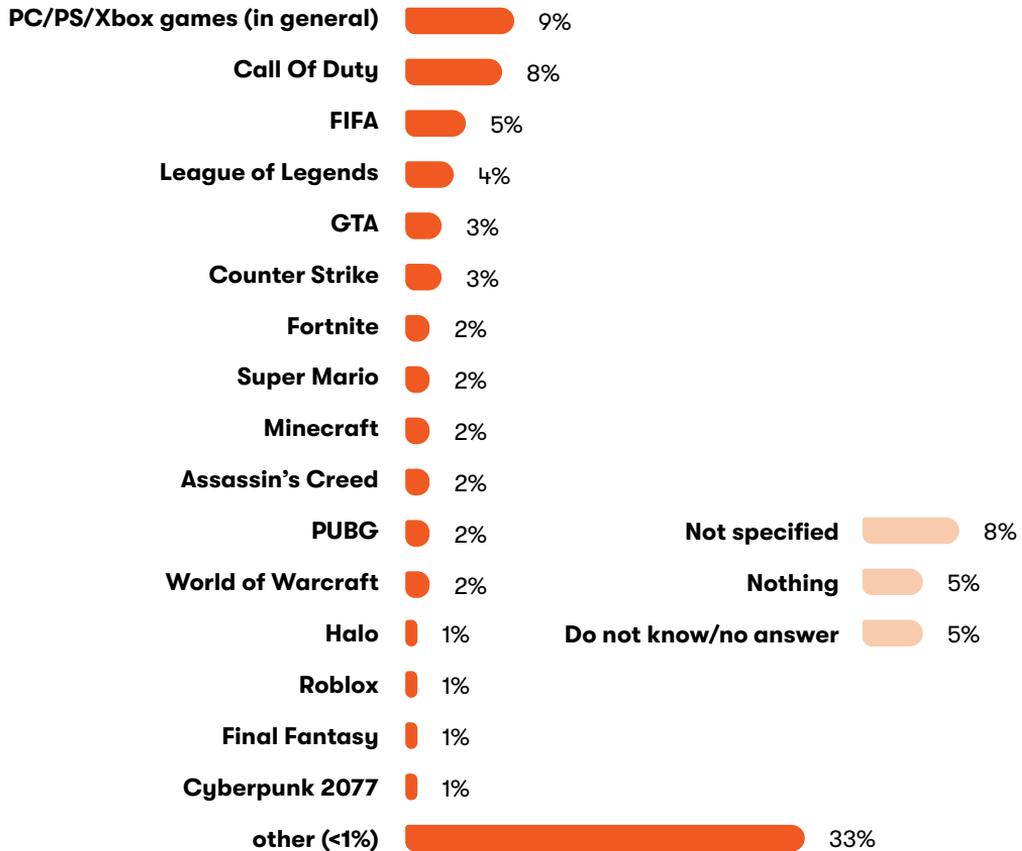
The majority of respondents do not know what could increase the Jamstack adoption or provided no answer to this question. However, it seems that training and explaining how the Jamstack works could bring more developers on board.



14 What is your favorite PC/PS/XBOX game?



Respondents could provide only one answer.



N = 769 developers

The survey shows that 9% of developers like PC/PS/Xbox/Nintendo games in general. Specific games that are popular among developers include Call of Duty (8%), FIFA (5%), and League of Legends (4%). Nevertheless, as you can see in the chart, the gaming market is highly fragmented.

8% mention only a game genre (most frequently adventure games), and 5% do not have a favorite game.



Conclusion.

Our survey among 769 developers in Europe, North America, Asia, and Australia shows that the Jamstack is still relevant in 2021. Here are some of the highlights from this report:

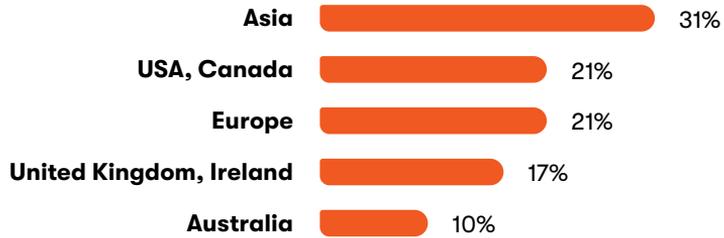
- 64% of developers are familiar with the Jamstack, and 59% have worked with it for more than one year.
- Top three reasons for choosing the Jamstack are **security** (45% of respondents), cost-effectiveness (42%), and reliability/stability (40%) together with productivity (40%).
- Most complex Jamstack projects serve hundreds to thousands of unique users and two to three channels, involve up to five services and ten engineers, and have a 5–10% level of dynamicity.
- Developers most often use **e-commerce** technologies (49%), WebSockets (40%), and serverless functions in their Jamstack projects.
- The most popular static site generators in 2021 are **Gatsby** (24%), Hugo (21%), and also Statiq (19%).
- Developers store their content in **Google Docs** (38%) and host their projects on **MS Azure** (29%), Cloudflare (27%), and AWS (25%).
- 83% of respondents are interested in initiatives providing a real-time editing experience on Jamstack sites.

Just like last year, developers said that they love how easy to use the Jamstack is and that there's nothing to hate about this popular and modern architecture. We're very curious to see how the Jamstack ecosystem will evolve in the future, and we hope you enjoyed reading this report!

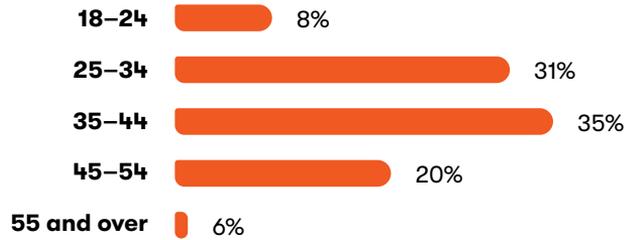


Overview of our respondents.

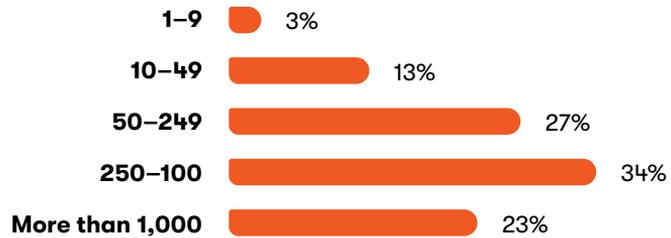
Country



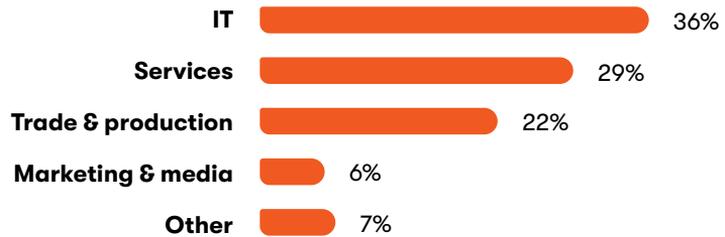
Age



Company size (employees)

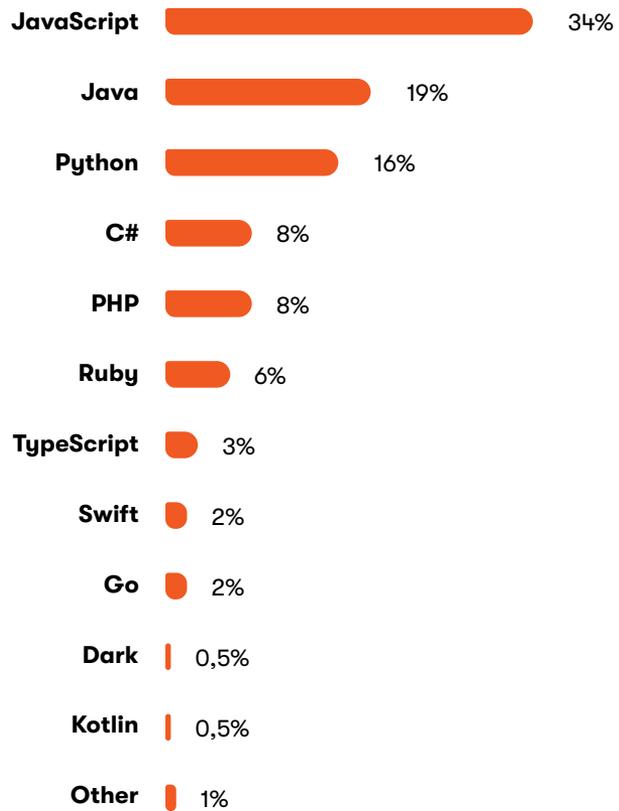


Industry

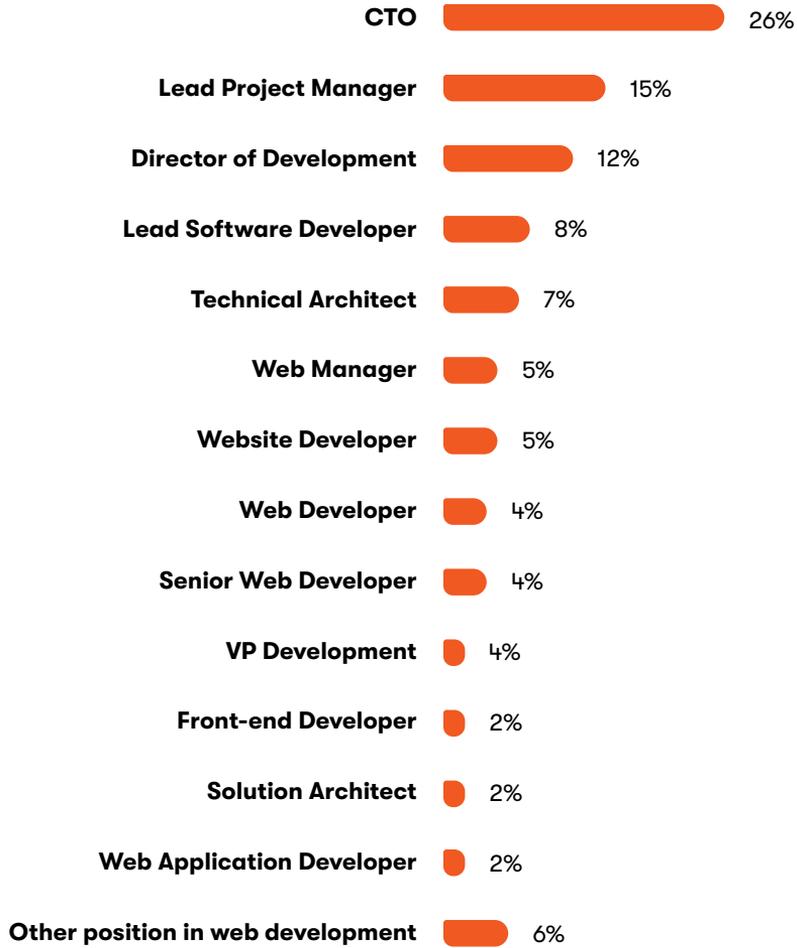


N = 769 developers

Primary programming language



Job title





The State of Jamstack 2021 Report.

[Talk to us](#) 

[START YOUR FREE
30-DAY TRIAL.](#)