netlify

Developer Productivity

Unlocking Developer Productivity for the Enterprise with Netlify

Unlocking Developer Productivity for the Enterprise with Netlify

When something works incredibly well, it feels like magic. That's the type of experience you want to build for your users. To do just that, you have to take a peek behind the curtain and examine the technology and developer tools that power the experiences your customers know and love.

There's a massive shift happening in the enterprise space. More and more companies are moving towards fast, flexible, developer-friendly tools and away from legacy architecture and proprietary processes. There's a good reason for that.

To stay in lockstep with both your customer's preferences and the pace of the market, you have to be able to build quickly and efficiently. That's where developer productivity is absolutely essential.

Netlify's mission is to boost developer productivity so your developers can focus on what they're best at — writing code. By eliminating the traditional barriers to true developer productivity, Netlify empowers developers to do their best work and make an outsized impact at work.

Using Netlify's comprehensive development environment, enterprise developers have all the tools they need to build, deploy, maintain, and improve websites, all in one place.

So, how do you unlock that magical customer experience? You start by building the ideal experience for your developers. We'll show you how to do just that, and cover how you can hurdle any roadblocks along the way.

The Barriers to Building An Ideal Developer Experience

Enterprises operate at a distinct level of scale, sophistication, and security. In an enterprise environment, you might not be able to adopt every new tool that comes to market, or every API that's released. Some of those developer tools can't support the needs of the enterprise.

Netlify thinks a bit differently about enterprise development. The entirety of the Netlify platform is built with stalwart security, intuitive developer workflows, and infinitely scalable architecture.

Some of the tools enterprises rely on today might have been intuitive when they were initially released. But, today, they're not doing that enterprise any favors. Let's explore the challenges you might encounter while trying to build with legacy tools in a modern-web world.

Speeding up deployment cycles

Your developers want to focus on code. It's as simple as that. But, there are so many things on the periphery of your developer's focus that call their attention away from the task at hand.

They might have to create a ticket for an upcoming sprint, or roll back to site to do some testing. They could be spending an inordinate amount of time wrestling with a new virtual environment they're trying to set up.

Anytime your developer has to close a text editor or pull themselves out of a focused flow, their productivity suffers. Netlify is built to help streamline a developer's workflow so they can concentrate on what they're best at and what's most impactful to your enterprise.

Speaking (and coding) in your developer's preferred language

Finding the right developer for the job is incredibly important. If, for example, you're building a new chat app, you might want to find someone who specializes in just that. You'd likely review their portfolio or take a look at any projects they've built that are related to the type of application you're planning on building.

When you use ubiquitous programming languages and developer tools, you widen the pool of candidates you can choose from. With a deeper pool of candidates, you can be more selective in picking the best developer for the job. You can find the expert you think will be right both for your upcoming projects, and your team of developers.

If you're using legacy software, you might be narrowing your candidate pool before even starting your search. Developers today are far less likely to learn legacy programming languages like Delphi, Object Pascal, or Perl that are commonly used in legacy software systems.

In a 2019 Stack Overflow Developer Survey, 67% of respondents listed Javascript as their primary programming language, while 63.5% know HTML and CSS. These are the core ingredients of the Jamstack.

Simply put, the Jamstack lets you control your site's dynamic functions using JavaScript, extend server-side capabilities by integrating your favorite APIs, and control the display of your content using Markup. This architecture gives you control and endless opportunity for extensibility with APIs, all without sacrificing speed or security.

Now your users don't have to wait on a server to deliver the entirety of your site's content. Using Netlify, your site is pre-built and waiting for your user to render it in their browser. This leads to much faster sites, gives you much more control over the experience you want to build for your users.

If you're using legacy software, you don't have that same degree of granular control. And, your options for hiring a developer to maintain that software are increasingly limited.

Your choice of available talent declines the deeper you dive into the niche of legacy programming languages. But, if you're using a more modern stack, the world is your oyster when it comes to talent recruitment.

Scaling your web presence

In a typical enterprise web stack, your developer has to manage a series of APIs, browser-based logic, a CDN, a load balancer, web servers, and application servers or databases just to get a production-ready application off the ground.

When your company plans on launching a new product or service, more load balancers and more servers are added to the stack. While this might accommodate a spike in traffic from happy customers interested in your product launch, it also adds technical debt for your developers.

Instead of adding more pieces of infrastructure to support your scale, you can instead use more flexible, robust tools like Netlify's CDN, ADN, and serverless functions.

These tools give you the freedom to scale on a dime, anywhere in the world without worrying about installing new infrastructure.

These principles of crafting enterprise developer productivity might sound abstract. So, let's see them in practice.

How Netlify Helped Canada's Largest Online Retailer 20x Their Productivity

Loblaw is Canada's largest online retailer with over 25,000 employees and 300 developers. Last year alone, the company posted $48 billion in revenue. The bedrock of their success is customer experience, a lion's share of which takes place online.

Over 7 million people engage with Loblaw's digital platforms every single day, while over 85% of Canadians shop at Loblaw stores every week. As a top choice for Canadians looking to buy goods online or in person, Loblaw has to ensure that every aspect of their business is streamlined and scalable, including their website.

Back in 2017, Loblaw was finding difficulty speeding up their time to design, develop, and deploy new sites and products.

Their deployment cycle followed an assembly line process in which one team completes their work and then passes the project down to the next stakeholder. (This might sound familiar to some.)

This assembly line style of project management wasn't working efficiently. Projects commonly sat in one team's domain until stakeholder feedback was resolved. This delay in getting feedback from stakeholders was slowing down Loblaw's time to market, potentially causing them to miss out on opportunities to grow customer loyalty.

At first, Loblaw tried to alleviate this problem by hiring more engineers to build products, but this just added more stops along the proverbial assembly line before a product could get to market.

Using Netlify, the Loblaw team was able to drastically increase their development speed and bring more products to market faster.

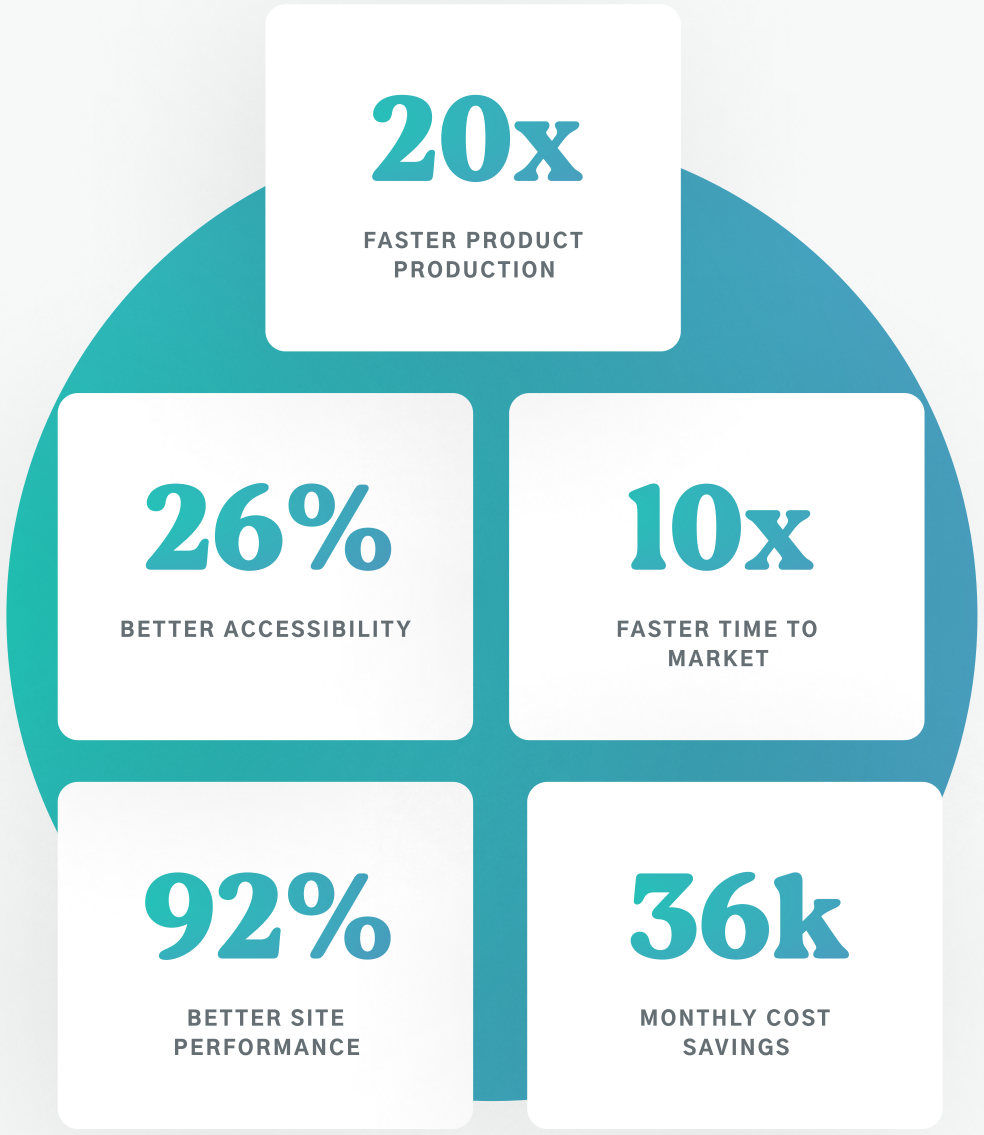
Before switching to Netlify, Loblaw shipped one new product per year on average. After switching to Netlify, Loblaw's team of engineers were able to launch 11 products in just four months.

By unlocking the power of developer productivity, Loblaw is able to get more products to market faster.

Loblaw realized that developer efficiency isn't measured by the number of developers working on one particular project or problem, but rather by the impact of the tools, architecture, and workflows they're using to solve a problem and bring a product to market.

Netlify allows the Loblaw team to avoid lengthy feedback cycles by ditching the assembly line-style deployment flow and instead using continuous deployment and integration processes.

We'll show you how Netlify can empower you to do the same, building and hosting web properties more efficiently.



20x
FASTER PRODUCT PRODUCTION

26%
BETTER ACCESSIBILITY

10x
FASTER TIME TO MARKET

92%
BETTER SITE PERFORMANCE

36k
MONTHLY COST SAVINGS

By unlocking the power of developer productivity, Loblaw is able to get more products to market faster.

The Netlify Difference

The principles of continuous deployment and continuous integration are essential to the modern developer's workflow.

By constantly rolling out updates in a continuous development cycle, developers can make changes iteratively and reduce the risk of impacting site functionality, while maximizing site performance.

When developers are closer to their code on a day to day basis, they have a granular sense of their infrastructure and the component parts therein, allowing them to ship updates frequently and with confidence.

The speed you gain from continuous deployment allows you to stay in lock step with your customer, pushing products to market faster and delivering on customer feedback with a quickness. This is absolutely critical to sustaining customer loyalty.

So, what does this workflow look like in action? Let's dive a little deeper to find out.

Inside the Netlify Workflow

At a high level, Netlify allows developers to deploy their code to a git repository with a simple git push. This triggers Netlify's CI / CD processes as Netlify automatically builds the site based on the code it received from your developer and pushes it to an enterprise-grade, global Content Delivery Network built for our favorite modern web stack — the Jamstack.

Let's take a look under the hood of Netlify's CI / CD workflow to see what features make all the difference when it comes to developer productivity.

Git to Global

What if you had to build your desk from scratch every time you arrived at work? Then you had to break it down before you left. Would you feel productive or motivated to work? The odds are that in-between looking for that alan wrench you need to tighten the last leg of your desk and starting work, you would be looking for ways to get rid of this cartoonishly unnecessary process standing between you the work that counts.

If you're using the wrong webstack, getting up and running in a production environment can feel a bit like building a desk before you can get down to business.

Netlify cuts through the inefficient steps developers suffer through just to start their work, and removes the barriers to production environment productivity that you'd typically encounter in a legacy web stack.

Using Netlify, you can spin up a new environment in a few minutes. All you have to do is create a new git branch in the repository or your choice from GitHub, to GitLab, to Bitbucket. Then name that branch, and select your Netlify Build and Deploy settings to tell Netlify how you want your site built and where you want that site deployed.

Netlify will deploy your new site to a branch of your choosing, allowing your developers to go above and beyond working in a single feature branch. Now, developers can create a new branch and generate an easily previewable URL automatically for the various independent tickets they're working on.

Set Up New Environments Easily

So, you've got your new site deployed to a branch. You've got your preview link. You're clicking around to make sure everything is looking great and performing as expected.

But, this page you've deployed doesn't exist in a vacuum, it exists within the context of your company and your web architecture. To truly QA and test a site, you have to view it within that larger context of other branches.

Netlify makes it a breeze to add additional branches to your recently deployed branch so you can get a full picture of your site's functionality with a few clicks.

In Netlify's Build and Deploy settings, you can select other branches from your git repository by simply typing in the name of the branch and clicking deploy. It's that simple. From there, your changes will trigger a deployment to Netlify and your site will be updated in an instant, allowing you to see the most recent changes.

Using CI and CD to Cut Down on Deploy Time

Remember that assembly line we referenced earlier? Let's imagine you're back on the proverbial development assembly line, waiting for a release to be handed down from team to team.

In an enterprise engineering team, you might have to coordinate DevOps in several different regions and offices, ensure everyone has QA'd your code, and hope everyone has availability on their calendar to meet and review the rollout plan.

Netlify allows teams to work more efficiently and asynchronously using CI and CD practices that aren't reliant on cumbersome IT environment setup and deployment dependencies. This allows developers to focus on the core application in front of them, rather than the infrastructure within the enterprise.

When a developer is done working on a critical part of your website, they can easily share their progress using Netlify's Deploy Preview tool. With a Deploy Preview URL, any developer on your team can review the full history of the website, with an interactive history of changes. Now, code reviews are more engaging and interactive when you don't have to wrestle with inefficient tools.

After your development team has reviewed a new site, they can merge and deploy it with a single click.

You can view your past deployments and a granular history of your assets uploaded, redirect rules, full build logs and more. If you need to roll back a deploy or review the changes from a specific build, you can do so easily just by clicking a unique deploy URL within the logs.

Once you deploy a site to your customers, Netlify solves any cache problems before they start. Netlify invalidates the cache across all CDN engine nodes at deployment, ensuring that no users see an older version of the site.

This type of asynchronous collaboration helps developers share their work faster, tighten review cycles, and push to production with confidence.

Finding What's Right for Your Customer Using Split Testing

With the right set of tools, you can deliver on customer preference without your customer having to tell you what their preference is explicitly.

Split testing is a tremendously valuable tool you can use to understand which changes your customers prefer, and which they might not.

Let's say you're launching a new feature. Instead of pushing it live to your entire customer base and hoping they like it, you can use a more data-oriented process.

Using Netlify's Split Testing tools, you can rollout that feature to a small percentage of your audience, and gradually increase the percentage of your audience that sees that feature. You can also A/B test features against one another to decide which you should implement.

The Features That Make All The Difference To Your Developers

We've covered some of the tools that can help streamline your team's performance and workflow. But, what about the individual developer? Netlify thinks of the little touches, and bespoke tools that make all the difference in a developer's typical workflow.

Netlify Dev

Netlify Dev is a hosted service that continually runs your dev command, just like you do locally, watching for changes. The result is an instant preview you can share with your entire team, with live updates as code and content change. You can also create new sites, set up continuous deployment, and push new deploys, right from the command line.

This is a huge boost to developer productivity. Developers can now replicate their changes in a production environment right from their local machines. This is especially important when writing cloud functions, and determining the needed dependencies before going to production.

Netlify Functions

All you have to do to set up backend actions, or a dynamic action Netlify is set a folder and drop in your functions. It's that intuitive.

It can be overwhelming using other serverless functions where you're responsible for managing service discovery, configuring API gateways, and coordinating deployments between our app and serverless functions.

Netlify automates and streamlines your serverless functions in a version-controlled environment so they're built and deployed along with the rest of your site. This not only eliminates overhead, it brings together the power of Deploy Previews and rollbacks to your serverless functions.

Netlify Build PlugIns

Netlify's pre-built plugins let you add new features to extend the functionality of your build process, run plugins at different stages of your build, and automate workflows and tasks that kick off events.

Instead of spending valuable time building solutions to solve common tasks like generating a search index for your site, you could simply search for a plug in and click one button to install it on the site of your choosing.

Netlify Identity

Netlify Identity service brings a full suite of authentication functionality, backed by the GoTrue API. This allows you to manage and authenticate users on your site or app, without requiring them to be users of Netlify or another service.

Let's say you wanted to gate a piece of content like a white paper or e-book, or ensure that only content administrators have access to certain sites within your suite of web properties. Netlify Identity lets you manage access seamlessly and control specific roles and permissions when it comes to site access and editing.

Netlify Forms

Netlify gives you the power to build forms without rallying backend engineers, or waiting on several people to come together to build the form you need fast. You can set up forms quickly on Netlify to gather responses from users or add a registration flow to a webinar or event site.

Using a Zapier plug-in, you can ferry the information collected in your Netlify-powered forms to your CRM to trigger custom processes like greeting a new event registrant with a personalized email, or confirming their spot in an upcoming webinar.

Your Next Step In Building a More Productive Developer Experience

Now, you have the tools you need to build a magical experience for not only your customers, but your developers, too.

With a more developer-friendly web infrastructure, enterprises can operate with the speed, scale, and security they need to stay ahead of their competition.

Every day, leading enterprises rely on Netlify to increase developer productivity and craft ideal customer experiences.

We hope you'll join us in doing just that alongside the 1 million developers already using Netlify to continuously build better, faster sites all over the world.

Contact Us →

netlify