It’s time to
migrate from Acquia
to Platform.sh

Platform.sh vs Acquia

It's time to migrate from Acquia to Platform.sh

Platform.sh vs Acquia

Looking for a more modern, cost-effective platform to host and manage your Drupal sites than on Acquia? You're not alone. Your team deserves more than a traditional managed services provider approach. They deserve a cloud hosting solution that natively supports headless Drupal and gives them the flexibility to develop using modern Drupal tools to build new features faster and to deploy them anytime, anywhere.

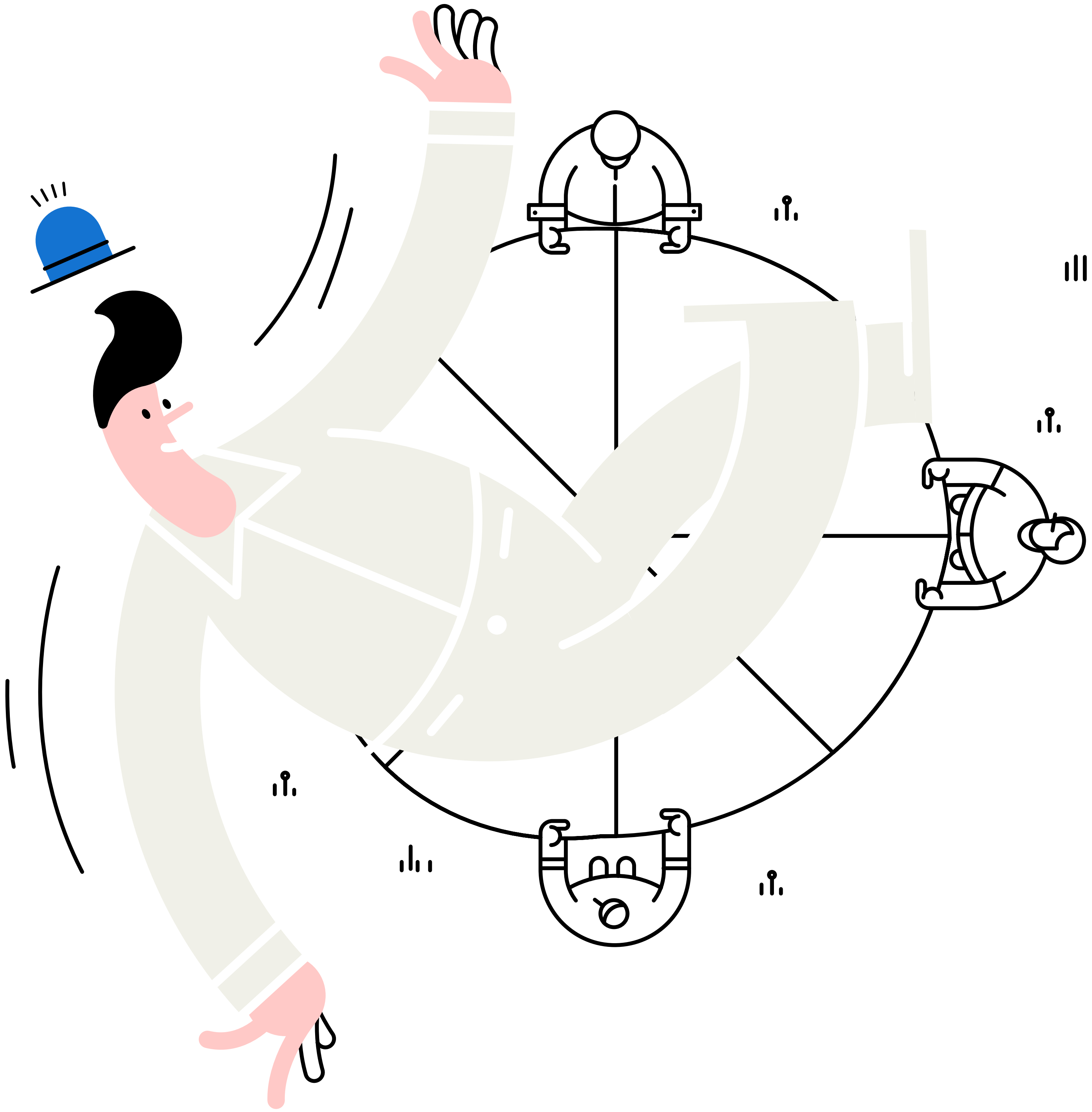
You probably have concerns about migrating to a new platform. How much is it going to cost? How many resources do you need to redirect from business-critical tasks to the migration? Does your team have the skill set to tackle the process? How long is it going to take? We get it.

> ***Today, more than 5,000 customers and 65,000 developers worldwide build and deploy their sites on Platform.sh.***

As the software vendor behind Drupal Commerce, Platform.s has deep roots in the Drupal community. And we say confidently that moving your sites from Acquia to Platform.sh can happen quickly and smoothly.

Why are people switching from Acquia to Platform.sh for modern Drupal applications?

We asked our customers for the reasons they switched. Here are their top-ten responses.



Platform.sh is built for Composer and Symfony, but doesn’t leave your legacy Drupal behind

1.

Modern Drupal 8 is built around two technologies that make developers’ lives easier.

Composer

Composer simplifies dependency management for PHP applications, much like PIP for Python or NPM for Node.js. Unlike Acquia, Platform.sh supports Composer natively.

Symfony

Symfony is the other technology at the heart of Drupal 8. As the cloud powering Symfony Cloud and Magento Commerce, our Symfony expertise runs deep.

All the services you need to run an ambitious digital experience with Drupal

2.

Drupal has never existed in a technological vacuum. With Headless Drupal or decoupled architectures becoming the norm, that's truer now than ever before. **Platform.sh supports multiple runtimes—like PHP and Node.js—in the same project, so you can manage and deploy your sites as a single app.**

Acquia supports Drupal, with limited Node.js support for additional cost, so you'll need to find a separate host for your application's dependencies, and manage the resulting complex deployment process yourself.

For your data, Acquia provides the basics: MySQL and Solr. That's great for a basic site, but Platform.sh provides far more.

> ***Platform.sh gives you simple access to MySQL and Solr, of course, and myriad other services, with no extra cost or complexity.***

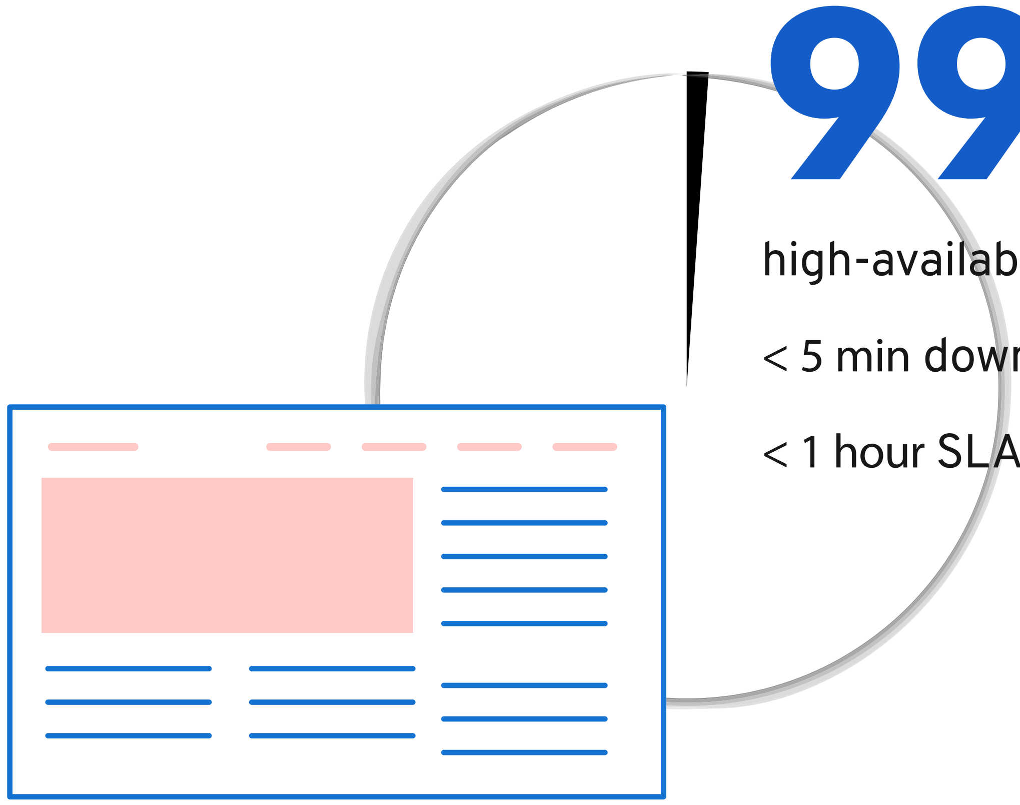
Snap in PostgreSQL, Kafka, Elasticsearch, RabbitMQ, Headless Chrome, Network Storage, worker processes, and more, with just a few lines of a YAML file.

Highly scalable, highly available

3.

Platform.sh can provide 99.99% high-availability guarantees for mission-critical sites. That's less than five minutes of downtime a month, and an SLA of less than an hour for urgent tickets. Acquia limits SLA for uptime to 99.95% (that's nearly 22 minutes per month), and only offers an SLA for Enterprise customers.

Platform.sh runs thousands of ecommerce sites and has extensive experience managing mission-critical applications. That's why our approach to high availability isn't based strictly on caching; we've designed triple-redundant architecture to ensure even transactional sites are fault-tolerant.



99%

high-availability guarantees

< 5 min downtime/month

< 1 hour SLA/urgent tickets

Platform.sh can scale up any site, as needed, when there's a surge of traffic—without downtime.

Drupal developers deserve tools for the way modern developers work

4.

Platform.sh has been based around Git from the very beginning. We think of it as our main API. And it drives collaboration across teams, between devs, and with other stakeholders. Do you need a new environment to work on a feature or to fix a bug?

It's as simple as:

```
git checkout -b new-idea
```

That's it.

One Git command, and Platform.sh clones your entire production app—data and all—in moments.

Acquia's approach is more linear, in a traditional dev/test/prod architecture. Though Acquia Cloud Enterprise customers can request more environments, that process involves support or account management intervention.

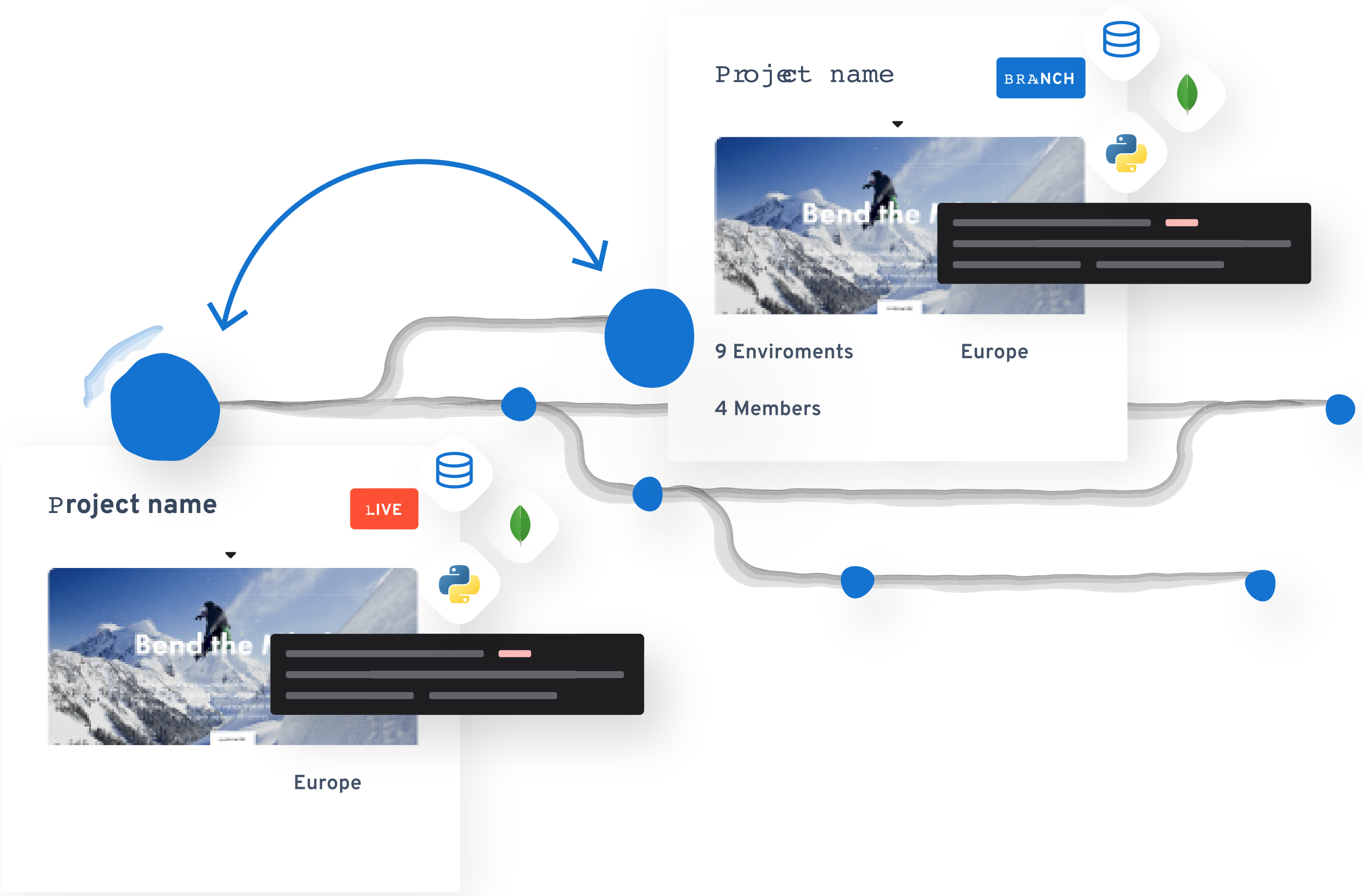
The Platform.sh workflow supports nested branches. This is great for teams because it means you can group work by feature, team, or release. And you can push changes live from any branch. Each environment also has configurable access control for devs and viewers, and auto-generated (and free) SSL certificates.

Acquia doesn't support teams doing parallel development.

Deploy anytime

5.

Because deploying to production is basically a Git merge, and your team is developing on a perfect clone—including all the components and data of your Headless Drupal site—**you can be assured that what worked in dev is going to work in production.**

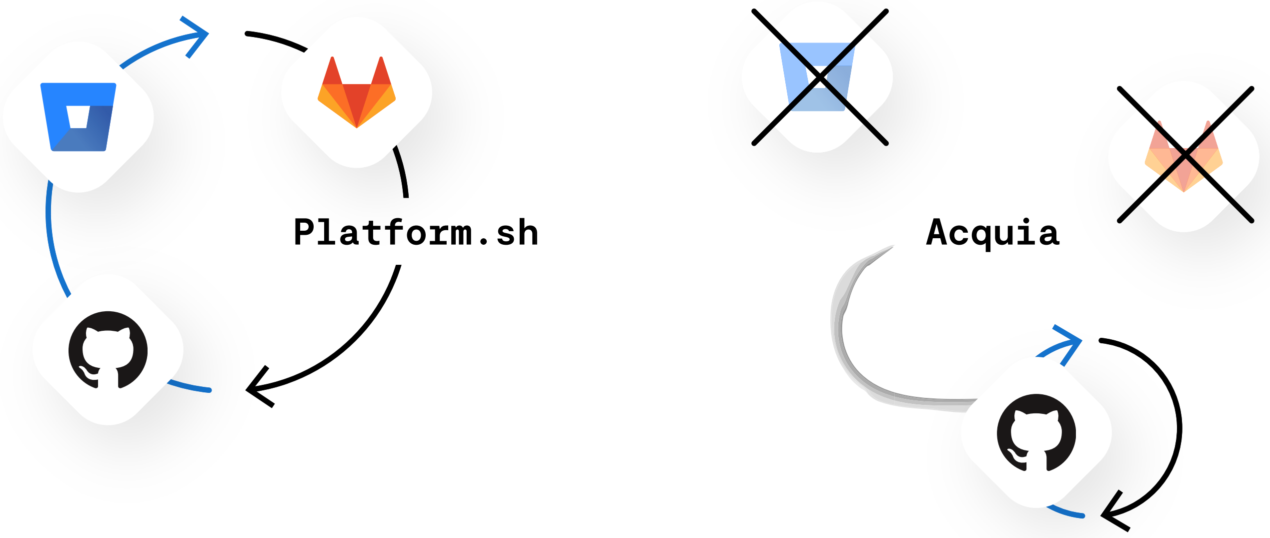


Real integrations with GitHub, Bitbucket, GitLab, and anything else

6.

Because Platform.sh has a robust API and supports custom webhooks, it's integrated with third-party tools you're probably already using, like GitLab, Bitbucket, and GitHub. Branches and pull requests created in those tools automatically generate (and clean up) environments in Platform.sh! That same API, coupled with our system of hooks and support for tools like Headless Chrome, gives you a robust continuous integration solution for Drupal and other technologies—again, at no extra cost.

Acquia provides some integration with GitHub via an extra-cost add-on product for Enterprise customers, but doesn't support GitLab or Bitbucket.



The Platform.sh API isn't limited to managing your current Drupal projects. With just a few commands, you can create new projects across regions around the world.

Platform.sh does Drupal best, at a lower cost

7.

The most cost-effective host for your Drupal applications, Platform.sh doesn't charge you for your success.

What does that mean?

	Monthly cost (Medium plans)	Uses
Platform.sh	$135	Low traffic, multiple apps
Acquia	$141	Personal Drupal sites and blogs

Acquia's pricing starts at $141 and is suitable for "Personal Drupal sites and blogs."

On the other hand, the Platform.sh Professional Medium tier can serve 500,000 pageviews per month, uncached, at a price of $135/month. If your app is cacheable, you can serve far more. That's a real business solution, not a personal blog.

Platform.sh also includes a global CDN with our Enterprise plans, starting at only $1,250/month, while CDN is an extra-cost service with Acquia. Do you need to run tests, compile Sass, use Node.js-based tools as part of your build? Platform.sh handles all of that natively, too, rather than charging you more.

Platform.sh does more than Drupal

8.

Across organizations, it’s rare for a single CMS to be adopted universally. There are many reasons for this: developer expertise, budget, and even internal politics. Supporting many languages and frameworks is good for us, and good for our customers. Real freedom comes from having a platform that can grow and shift with you as your needs evolve.

Platform.sh isn’t exclusive to Drupal by design. Instead, we let you run Drupal alongside your Node.js/Javascript app, Java, Python, and more at no extra cost. You can combine the front end and back end of your app in the same project, deployed automatically by us across our secure, containerized grid.

With Platform.sh, you can use the best solution for every site you build and manage: WordPress? Our PHP support is peerless. Static Site Generators (SSGs)? Java? Run CMS apps like Brightspot, or OSS stacks like Spring MVC and MongoDB.

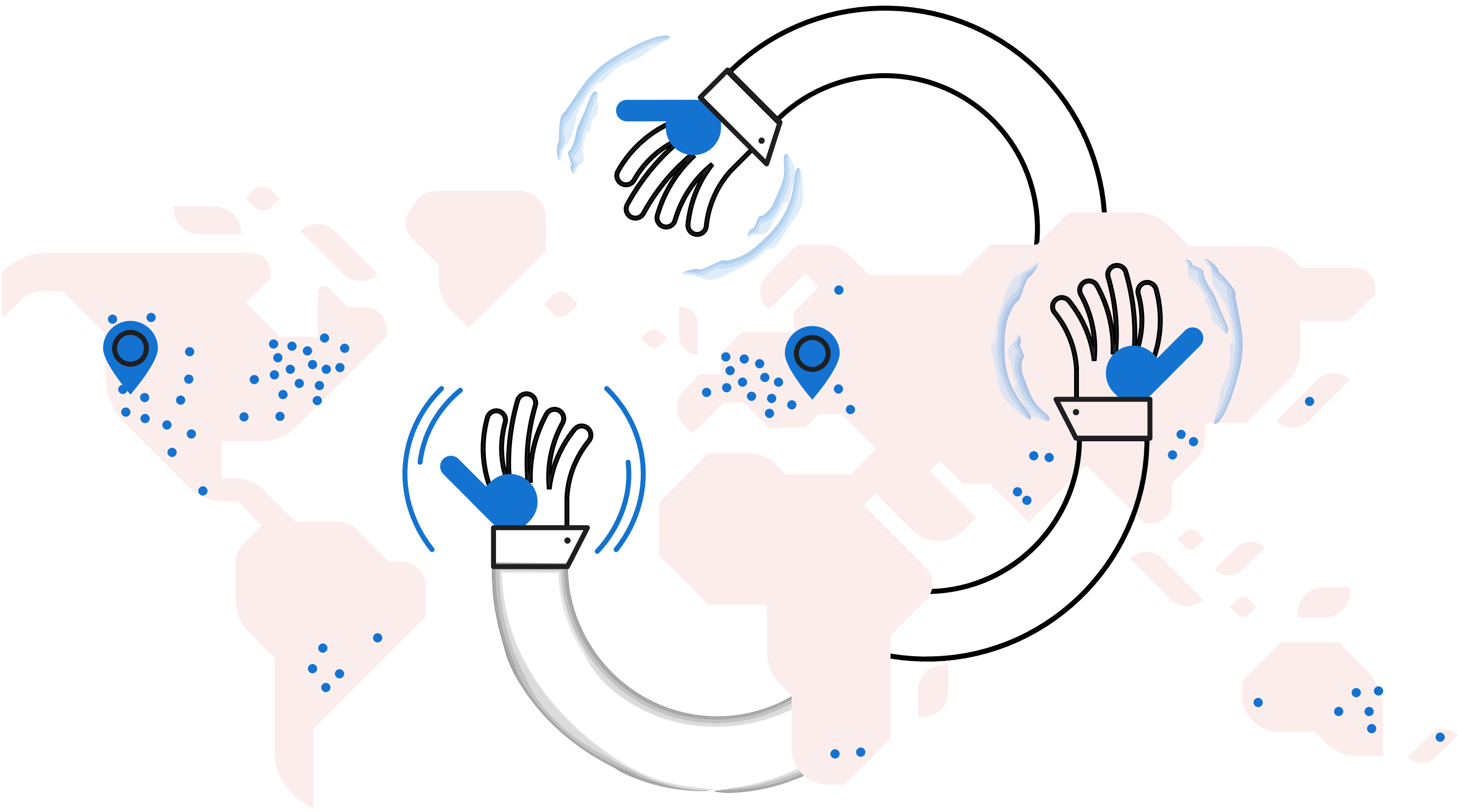
	8	JS	W	php	H	jekyll	G	JS
Platform.sh	✓	✓	✓	✓	✓	✓	✓	✓
Acquia	✓	−	✗	✗	✗	✗	✗	✗

Acquia supports Drupal, with limited Node.js app support as an extra-cost feature for Acquia Cloud Enterprise.

Platform.sh has Drupal expertise when you need it

9.

Selecting the right platform for scalability is important, but sometimes your team will run into challenges. For that, we have a global team of support engineers spanning North America, South America, Europe, Africa, Oceania, and Asia.



Need hands-on support to keep your Drupal projects patched and secure?

Platform.sh does that, too. Just ask us about Application Support.

Choice is good. That’s why Platform.sh is multicloud.

10.

Choosing which Infrastructure-as-a-Service (IaaS) provider to use can be a gamble, and once your die is cast, it’s hard to switch. The same can be true of your hosting or PaaS provider. That’s why most hosting providers like Acquia work with only a single IaaS partner (in Acquia’s case, that’s Amazon Web Services [AWS]).

Instead, Platform.sh makes it easy to use the IaaS provider that works best for your business, whether your choice is based on data locality for compliance, getting your application as close to your end users as possible, or simply preference. We can run your applications in the US, Canada, Australia, Ireland, Germany, and France.

Platform.sh works with four IaaS partners—Amazon Web Services, Google Cloud Platform, Microsoft Azure, and Orange—to give you the most choices and global reach. And you can run the same code across all four of these platforms without making any changes.

Need a custom location? Let’s talk.

We take security and compliance seriously. For some countries and companies, that means hosting locally, too. Not only are we SOC 2 certified, we’re compliant with the CASL in Canada, and Germany’s BDSG and strict data locality and access rules.

How to migrate your Drupal 8 site to Platform.sh

1. Copy down your Acquia Git repository details

Connect to your Acquia Cloud account, select the site you want to migrate, and retrieve the Git and SSH connection details.

2. Export your data and configurations

Select your MySQL database[s] from the Acquia console, and download it to your computer. Make a copy of your scheduled cron jobs and variables, which are easily configured in Platform.sh.

3. Prepare your Platform.sh project

Once you've created a new project on Platform.sh to host your site, you'll need to add in a few files to define your crons, variables, and mount points. You can download these directly from our GitHub templates, making setup simple. Did you know you can even create new Platform.sh projects from the command line?

4. Time to deploy

Simply commit your changes to your Platform.sh project. We'll take care of running

Composer, compiling Sass, and containerizing your Drupal, Solr, and MySQL apps, then deploying to our grid. We'll automatically generate an SSL certificate for your site, too.

5. Sync your database and media files

Using the Platform.sh CLI, you can import your MySQL database into your project. You can even transfer your media files directly from Acquia to Platform.sh.

6. Test your site, and go live

Platform.sh has continuous integration and continuous deployment built in, so if you've got unit or integration tests, Platform.sh can run those for you. Once you're ready to go, just add your domain to the project, and direct your DNS to Platform.sh.

Three ways to migrate

Many customers are up and running on Platform.sh in a matter of minutes or hours (depending on how much data needs to be migrated). **There are three options to get started:**

- » Use our how-to guide to Acquia migration.
- » Work with our onboarding team to migrate your sites.
- » Have our team perform the migration for you.

Chat with our team now.

Or try out Platform.sh

for free.

Power. Simplicity. Freedom.

The best tools are the ones that fade into the background and let you get on with your work. Find out how we can help your organization ***focus, faster.***