THE DZONE GUIDE TO

BUILDING AND DEPLOYING APPLICATIONS ON THE CLOUD

2016 EDITION

RESEARCH PARTNER SPOTLIGHT

NGINX

KEY RESEARCH FINDINGS

704 IT Professionals responded to DZone's 2016 Cloud Development Survey; the demographics of this survey are as follows:

- 67% of these respondents use Java as their primary programming language at work.
- 76% have been IT professionals for over 10 years.
- 39% work at companies whose headquarters are located in Europe, 35% in the USA.
- 40% work at companies with more than 500 employees, 16% at companies with more than 10,000 employees.

AWS STILL ON TOP

When asked what cloud service providers their organization uses, an overwhelming amount of respondents to DZone's 2016 Cloud survey said they had used Amazon Web Services. 58% of respondents said their org used AWS in some way—more than twice the runner-up, Microsoft Azure (which 26% or respondents said their organization uses). Coming in third were Google Cloud Services, which had 17%.

When cross-tabbed with which issues the respondents expected or experienced with their cloud platform, respondents who used each of these top three service providers were more likely to respond that they "neither experienced nor expected to experience" than the average of all cloud service provider users (608 out of our 704 respondents). See Figure 1 for a breakdown of those providers compared to the average of all other providers.

Respondents expected or experienced greater scalability and higher availability from using cloud platforms over other benefits we asked about (such as better application performance and faster time to market), so it makes sense why orgs might want to stick to larger or more established cloud providers—they're seen as more likely to be able to have more data centers, more availability zones, more robust infrastructures to scale easily and guard against downtime.

Still, there's certainly some stick-to-itiveness involved in which cloud providers companies are using. Between last year's DZone Cloud survey and this year's, AWS usage gained just 1%, while Microsoft gained 2% and Google lost 2%. Providers such as Heroku, Rackspace, and OpenStack barely saw a change (under 1%). Digital Ocean saw the biggest boost, jumping from 5% to 8%.

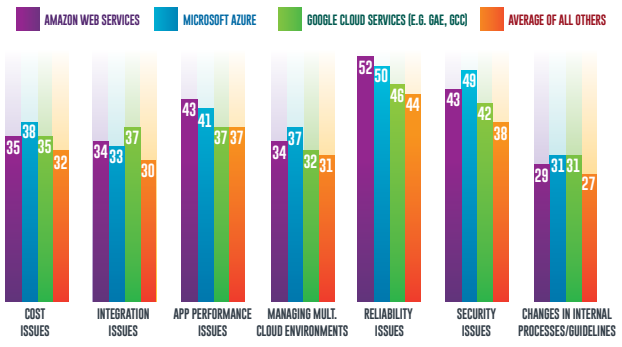
DOCKER IS HUGE. BUT YOU KNEW THAT

When we asked about which open-source cloud products devs used last year, we didn't include Docker. We found OpenStack to be pretty popular (30% in 2015), with Cloud Foundry and OpenShift as runners-up (12% each in 2015). But the majority of our 2015 respondents (55%) said they had not used an open-source cloud product for a business application.

This year, we decided to throw Docker into the mix and add it as an option to the same question. Sure, it's not quite the same as the PaaSes or IaaSes it was pitted against, so we're not going to try to compare the results against other open-source options (apples and oranges, and all that).

45% of our survey respondents this year answered that they had used Docker for a business application. And when asked about whether their org was using or looking into using containers (something we did ask about in 2015), about 66% said their org was either using or evaluating a container technology right now—which is up about 20% from last year's responses, pointing to a lot of recent container adoption that seems to match the buzz surrounding containers and Docker.

01. CLOUD PLATFORM ISSUES NEITHER EXPERIENCED NOR EXPECTED, BY CLOUD PROVIDER USED



02. DOCKER USAGE COMPARED TO TIME BETWEEN RELEASES

We also found that as developers' time between releases decreased, their container usage increased. While devs with all sorts of release schedules did say they were evaluating containers, those who release "on-demand" were 17% more likely to use containers currently than those who estimated "monthly" releases. Since containers can make decomposing apps easier, they also make it easier to make a change to one part of an app without touching anything in another container, so you have to worry less about patches turning into bugs.

LARGER ORG, MORE HYBRID CLOUD

There are plenty of factors that go into deciding whether to choose a public, private, or hybrid cloud platform: security, availability, scalability, cost, flexibility... the list goes on. For our last three Cloud surveys, we've asked respondents "Which cloud platform type best fits your company's needs." The answers haven't fluctuated much between those surveys. About half of respondents prefer hybrid, while the other half leans slightly toward private.

We found, however, that there is a direct correlation between the size of a respondent's company and which type of cloud platform that respondent thinks best fits that company's needs. Those in small companies (from 1–9 employees) were 15% less likely to choose hybrid cloud platforms than those in the largest companies (10,000+ employees), with a quite linear trend in between. The choice of public cloud platforms trended linearly in the other direction, with 36% of those in the smallest companies choosing public, versus 9% in the largest companies.

Of course, very small companies are generally less likely to be able to handle an all-private cloud infrastructure, just given the number of employees they would need in order to maintain it. Public cloud is—in the absence of dedicated enterprise IT staff—just easier. But that doesn't negate the advantages of having some of your cloud platform private. As cloud computing becomes more widespread, hybrid's popularity will certainly increase; at the same time, as public cloud platforms become easier (and cheaper), it is likely to close the gap with private amongst those who haven't jumped on the hybrid bandwagon.

PAAS LOVES THE LANGUAGES

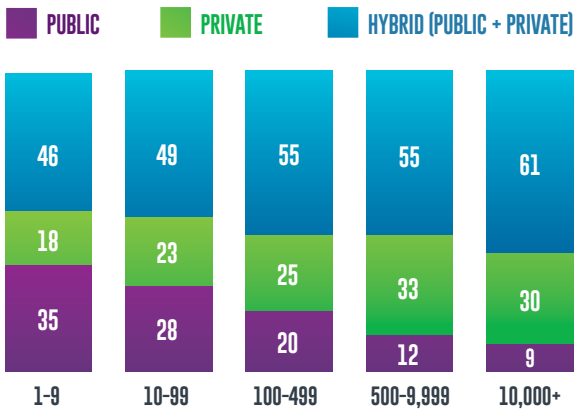
We developed a hypothesis about Platforms as a Service—the most popular cloud-based service in production amongst our respondents. It seemed that, because of how closely PaaSes are tied to the language being deployed, that those respondents whose organizations used fewer languages would be more likely to use a PaaS. Need to use more languages? Maybe you'd go with an IaaS—more work, but more flexibility.

Actually, when correlated against the number of programming languages each respondent said their organization uses(up to four), there was an upward linear trend for PaaS usage. Respondents who said their org only used one language were 49% likely to say their org used a PaaS. Those who said four languages were used at their organization were 68% likely. (Data above four languages for a single org was insufficient for analysis.)

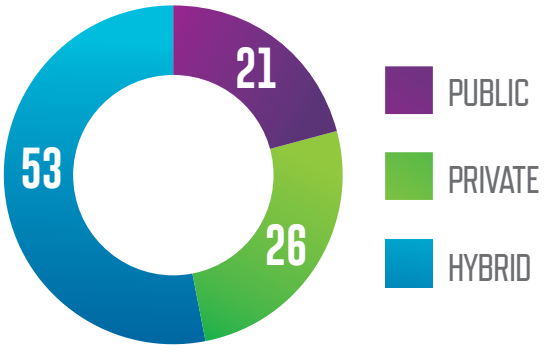
With multiple PaaSes, however, it seems an organization can more easily handle multiple languages. Yes, perhaps a single PaaS is tied to a particular language; but multiple PaaSes would allow an organization to utilize different languages for different uses, without having to build its own platform for each.

For Infrastructures as a Service—the second most popular cloud-based service in production—data was a little more scattered, but still trended upward, up to three languages used. Still, those who only selected one language for their organization were only 49% likely to say their organization uses an IaaS; two-language respondents were 57% likely; three languages showed 64%; and four languages hopped back down to 54%.

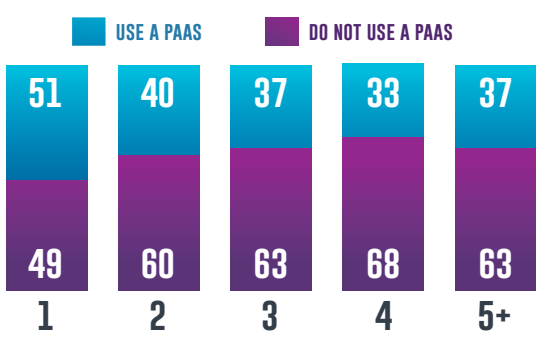
03. **PUBLIC, PRIVATE, OR HYBRID CLOUD PREFERENCES, BY COMPANY SIZE**



04. **PUBLIC, PRIVATE, OR HYBRID CLOUD PREFERENCES OVER ALL RESPONDENTS**

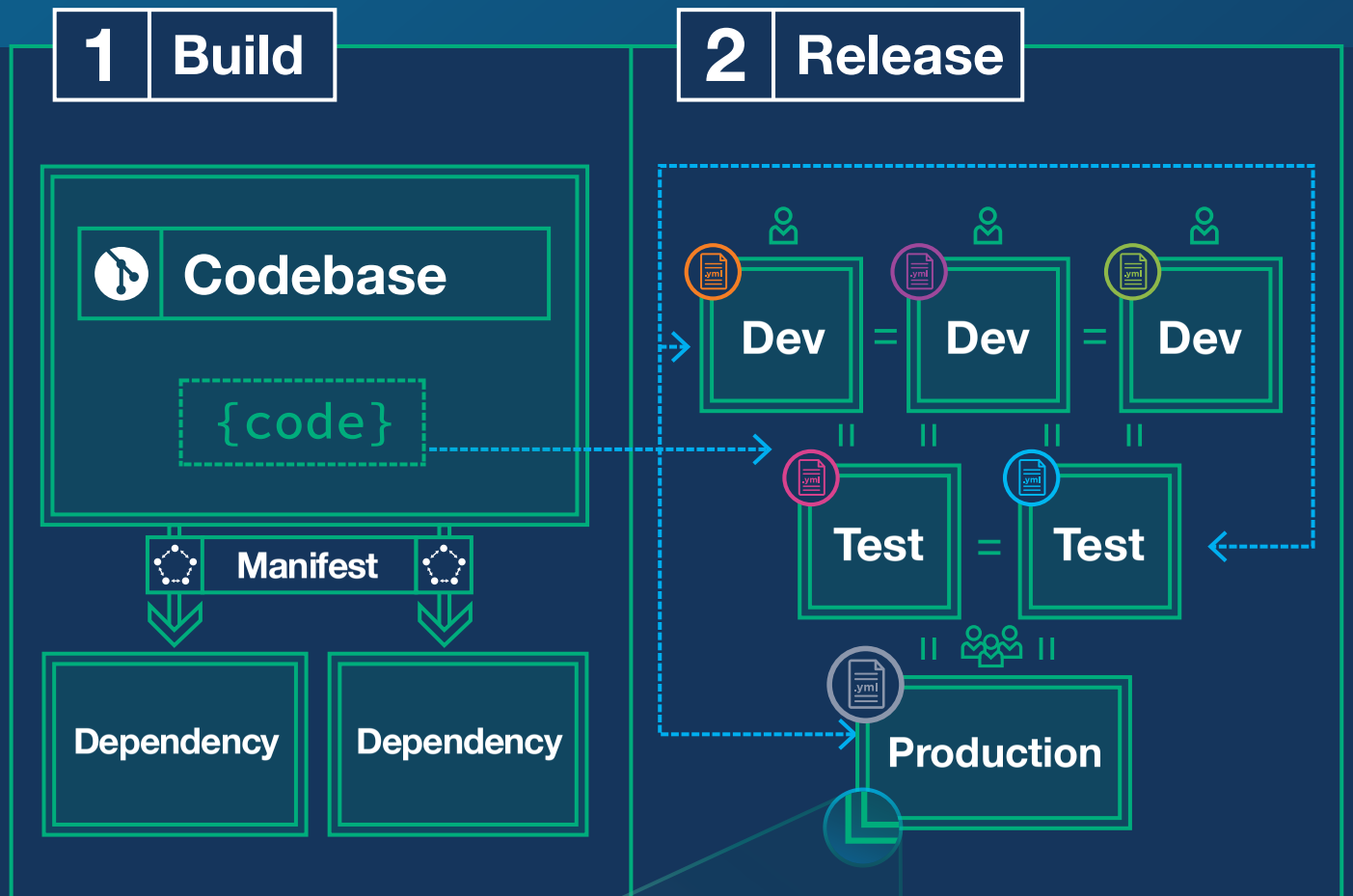


05. **PAAS USAGE, BY NUMBER OF LANGUAGES USED IN THE ORGANIZATION**



The 12-Factor App

Modern web applications run in heterogeneous environments, scale elastically, update frequently, and depend on independently deployed backing services. Modern application architectures and development practices must be designed accordingly. The PaaS-masters at Heroku summarized lessons learned from building hundreds of cloud-native applications into the twelve factors visualized below.



the 12 factors

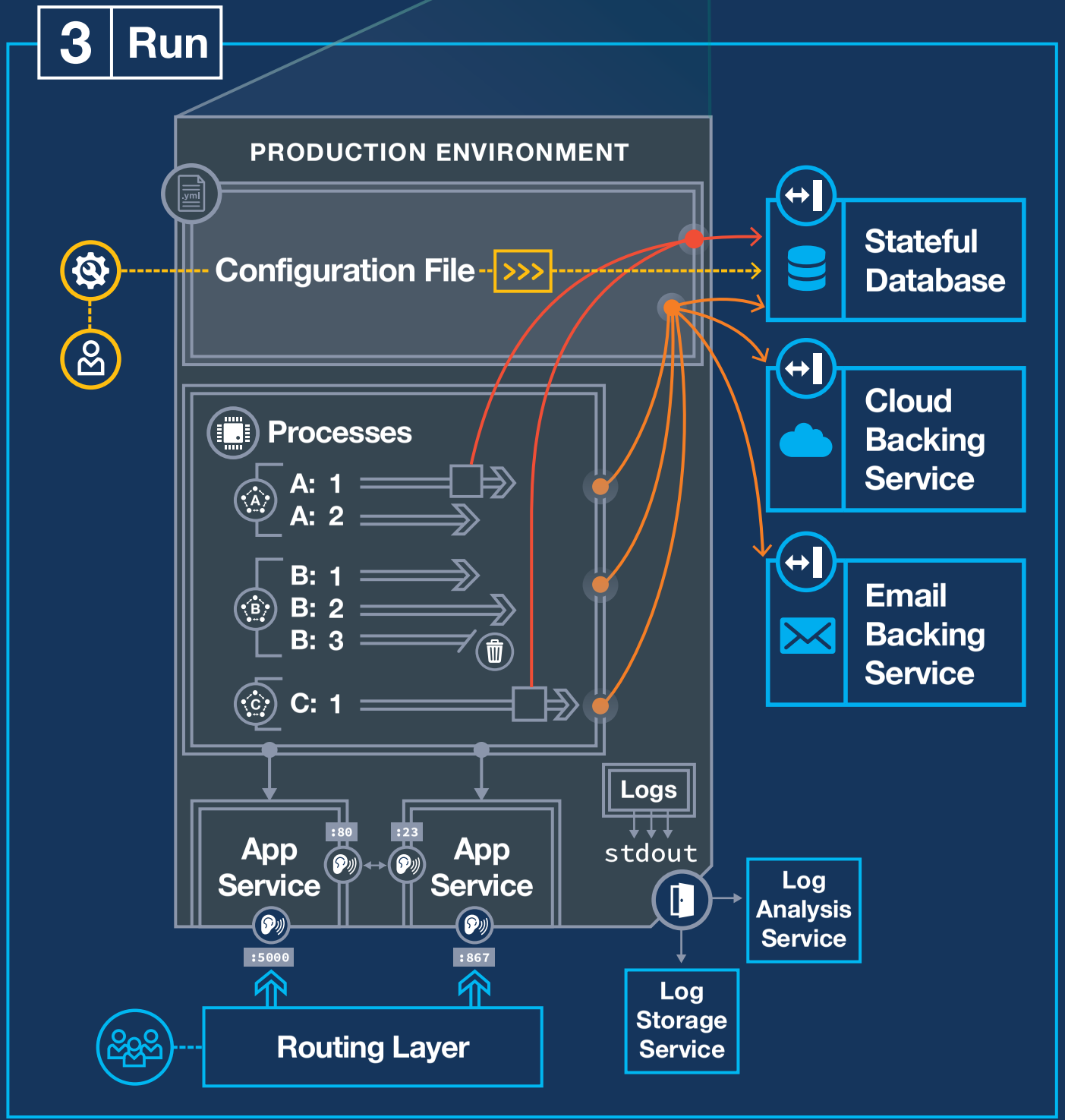
Codebase
One codebase, many deploys, strict version control

Dependencies
Explicitly declare and isolate dependencies

Configuration
Store config in each deploy environment, preferably using environmental variables

Backing Services
Treat backing services as resources (neutral as to local vs. third-party) located via config

Build, Release, Run



Strictly separate build, release, and run; never change code at runtime

Processes

Keep all processes stateless and share-nothing; store state (with other persistent data) in a stateful backing service

Port Binding

Bind every service to a port and listen on that port; don't rely on runtime server injection

Concurrency

Distinguish process types (e.g. web, background worker, utility) and scale each type independently

Disposability

Make processes start up quickly (<4s from launch to ready) and shut down safely (for web process: stop listening, finish current requests, exit; for worker process: return current job to work queue)

Dev/Prod Parity

Maximize dev/prod parity by minimizing gaps in time (between deploys: hours), personnel (authors=deployers), and environment (use adapters for backing services)

Logs

Log by writing all output streams to stdout; rout streams using non-app services

Admin Processes

Run one-off/admin processes (db migration, REPL , one-time scripts) in same environment as normal processes

THE STEP-BY-STEP GUIDE FOR

KEEPING YOUR CLOUD APPLICATIONS SECURE

While cloud adoption is accelerating rapidly, security and data privacy professionals are stretched to the limit keeping up. How do you ensure data in the cloud is secure? Which cloud applications can be trusted with business critical information? Cloud apps reside outside the perimeter, requiring a different breed of security solutions that follow the data, application, and user. **Let's explore a step-by-step cloud protection plan to help protect valuable corporate data from breach and exfiltration.**

STEP 01 DISCOVERY

Are employees using software applications your IT department doesn't know about? Yes they are!

- ☐ Use cloud app detection tools to identify what's being used (by whom and how often) and whether critical corporate data is involved.
- ☐ Consider a Cloud Access Security Broker (CASB); challenge the vendor to provide a Shadow IT Assessment to learn how big a shadow IT problem you are dealing with.

STEP 2 RISK ASSESSMENT

How do you know which apps present the biggest risk? Know which apps to sanction, monitor, or block (the good cloud/bad cloud challenge).

- ☐ Consider a rating system to identify cloud app risk attributes—this will help focus your protection in the right places: bit.ly/cloudcomprisks.
- ☐ Make sure the rating and reporting system can readily upload, anonymize, compress, and cache log data for shadow IT analysis and easily deliver automated risk assessment reports.

STEP 3 USER COACHING

Do all your employees know about common cybercrime tactics? Do your developers and other IT staff know the OWASP Top 10? The Cloud Controls Matrix?

- ☐ Reduce the Gilligan effect (remember Gilligan's Island? Gilligan unknowingly caused damage everywhere he went). There will always be a Gilligan, but security awareness training will lessen the effect.
- ☐ Employ constant reminders and more formalized quarterly training; this is a simple and low cost way to reduce risk of malware.

STEP 4 POLICY ENFORCEMENT

Security policy enforcement must be highly granular and real-time. These can be harder to achieve for cloud applications.

- ☐ Set policy controls based on user activity, using tools and business rules that are content and context aware based on user group, device, location, browser, and agent.
- ☐ Consider using a secure web gateway (on-premises, public cloud, or hybrid), plus an integrated CASB solution with advanced data loss prevention (DLP).

STEP 5 PRIVACY & GOVERNANCE

How are you addressing privacy and governance? Data in the cloud requires unique data-centric security policies.

- ☐ Use appropriate encryption – it's essential in any context. But for cloud security in particular, tokenization (substitution of secure for non-secure data, complementing a secure lookup table) can be especially practical.
- ☐ Make sure encryption doesn't impact application functionality (searching, sorting, etc.). If any of these are made significantly more difficult, users will figure out how to avoid it.

STEP 6 ENCRYPTED TRAFFIC MANAGEMENT

How are you maintaining privacy while selectively decrypting for security reasons?

- ☐ Consider addressing SSL and other forms of encrypted traffic to/from the cloud.
- ☐ Measure the impact of SSL/TLS on application performance and on payload visibility to internal security professionals.
- ☐ Tip: For industries whose traffic is more than 50% encrypted (like financial services and healthcare), policy-based traffic decryption may require a dedicated SSL visibility subsystem and/or specialized network architecture.

STEP 7 INCIDENT RESPONSE

How do you identify and quickly respond to malicious activity?

- ☐ Avoid the "car alarm syndrome!" Too many false alarms and alerts can be "needles hiding in needles" in the haystack.
- ☐ Consider beefing up incident response with a dedicated forensics function. Malicious software deliberately hides its tracks, and low-level complexity introduced by cloud deployment makes intuitive human interfaces especially key for incident response (e.g. security analytics, free-form search, and integration with 3rd party SIEM systems).

Even if you're not ready to make a big investment in a CASB solution, it's time to start managing cloud risk. To learn more about how to move to the cloud with confidence, view this webcast: dc.bluecoat.com/Forrester_Webinar

BY LISA ROM, SENIOR MANAGER OF AMERICAS MARKETING, BLUE COAT SYSTEMS

5 Tips for Optimal Cloud Performance

The cloud helps you get applications up quickly - but you still need to optimize your infrastructure if you want high-performance applications. NGINX has built the leading cloud-native application delivery software, so we've assembled these five tips to help you get the speed you need.

TIP 1. SIZE RESOURCES TO FIT

Sharing physical resources, such as RAM, with other people's VMs can cause unpredictable performance slowdowns. To reduce contention, use a dedicated host, or size your VM CPU, storage, and other requirements to use entire actual physical resources. Then use the following tips to make your application as hardware-independent and resilient as possible.

TIP 2. USE A REVERSE PROXY SERVER

Use a reverse proxy server to interface with the Internet and a separate application server to deliver your site's services, then optimize each function. This speeds up overall performance and provides benefits such as security controls and monitoring capabilities.

TIP 3. USE MULTIPLE APP SERVERS WITH LOAD BALANCING

Use your reverse proxy server as a load balancer for multiple application servers. The number of servers scales dynamically to match user demand, cutting costs. You can use NGINX load balancing alongside cloud capabilities such as Elastic Load Balancing on Amazon Web Services.

TIP 4. CACHE STATIC AND DYNAMIC CONTENT

Cache static files – GIFs, JPEGs, CSS files, and so on – on the reverse proxy server. Then cache application-generated files briefly, for instance for one second, to give your application server a big break.

TIP 5. MEASURE AND MONITOR

Measure performance and monitor uptime and responsiveness. Then, experiment with each aspect of your implementation to optimize it.

Read more of our blog posts to learn about cloud-native flawless cloud-native application delivery at scale.

WRITTEN BY FLOYD EARL SMITH
TECHNICAL MARKETING WRITER, NGINX

PARTNER SPOTLIGHT

NGINX Plus BY NGINX

NGINX Plus is the complete application delivery platform for the modern web.

CATEGORY
Application Delivery

NEW RELEASES
2-3 major releases per year with regular updates between

OPEN SOURCE?
Open Source & Commercial

CASE STUDY

MuleSoft offers a leading cloud-based integration platform that enables enterprises to easily and securely connect apps, data, and devices. MuleSoft's challenge? Meet customer performance and availability requirements, with little visibility into the customers' applications or upstream configurations. MuleSoft relies on NGINX Plus for load balancing, monitoring, and more. NGINX Plus enables MuleSoft to accurately forward requests to thousands of upstream workers with minimal latency. In addition, with NGINX Plus MuleSoft can easily ensure end-to-end SSL encryption of customer traffic. NGINX Plus gives MuleSoft flexibility in its cloud platform to innovate faster and provides them with the scale and performance they need.

STRENGTHS

- HTTP, TCP, and UDP load balancing
- Easy to deploy in any environment
- Advanced monitoring and management tools
- Rich security controls

NOTABLE CUSTOMERS

- Airbnb
- AppNexus
- Discovery Education
- Groupon
- Netflix
- Uber
- Wix
- Zendesk
- Blue Jeans

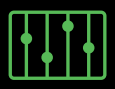
BLOG nginx.com/blog | **TWITTER** @nginx | **WEBSITE** nginx.com/cloud

NGINX

Migrating to the cloud?

Let us be your guide.

Advanced load balancing and caching



Custom health checks for high availability

Visibility into the performance of your infrastructure

Built and optimized to run on cloud and containers

Use independently or with load balancing services

Learn more at:
nginx.com/cloud