

NGINX

MORE INFORMATION AT [NGINX.COM](https://nginx.com)

Who are we?

Floyd Smith

Technical Marketing Writer

Formerly:

- HSBC, Apple, AltaVista
- Author, including *WordPress in Depth*

Kunal Pariani

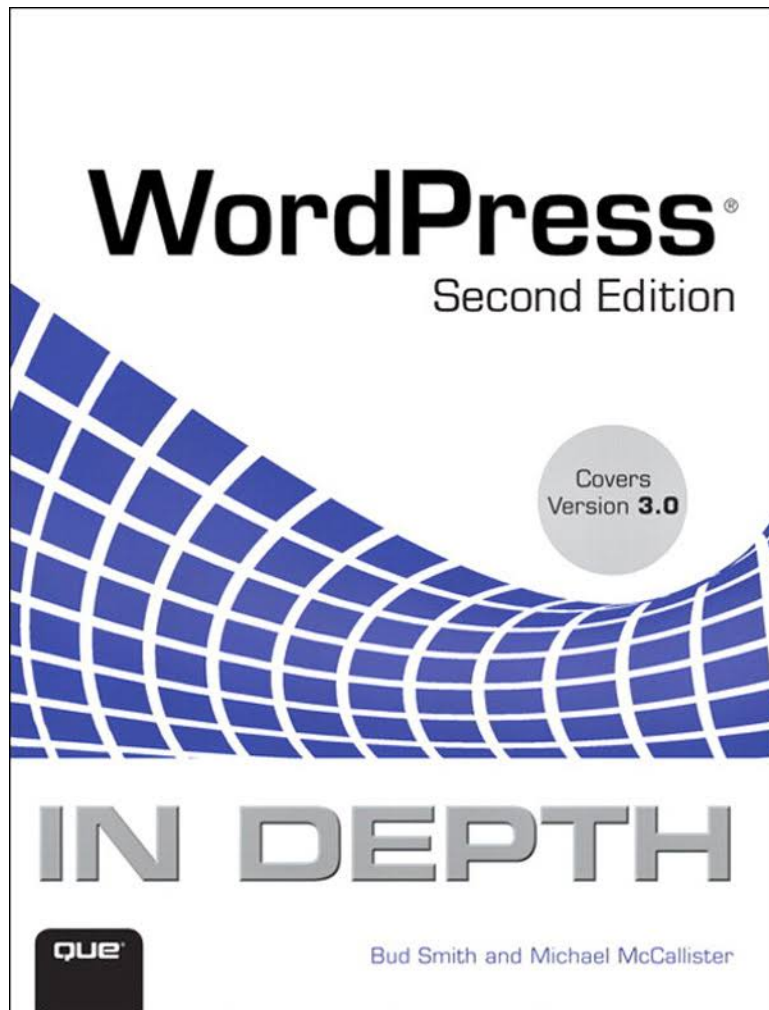
Pre-sales engineer

Faisal Memon

Product Marketer

Patrick Nommensen

Digital Marketing Manager



nginx.conf

September 7th-9th

Austin, Texas – first time out of the Bay!

www.nginx.com/nginxconf

Code: MEETUP

50% off 2-day and All-Access passes

Final day is **training**



OVERVIEW

GETTING THERE

SPONSOR

CONTACT

PROGRAM

REGISTER NOW



16

Austin
TX

September 7-9

Have a question about NGINX Plus?



Good/bad/good news

Good news: I do have some WordPress tips, and can point you to more.

Bad news: I'll take a roundabout route to get there.

Good news: By the end you'll have a very good idea of your options.

Resources:

1. Our NGINX/WordPress [tips article](#)
2. Fresh start with [microservices](#)
3. "NGINX WordPress" is a surprisingly fruitful [search](#)

What will we not do?

W.C. Fields > Quotes > Quotable Quote



“If you can't dazzle them with brilliance, baffle them with .

— W.C. Fields

Read more quotes from [W.C. Fields](#)

Some questions for you all...

- Your sites
- Your nightmares
- Your NGINX knowledge and use
- Open source NGINX and NGINX Plus

**And now,
a little bit about NGINX...**



". .. when I started NGINX, I focused on a very specific problem – how to handle more customers per a single server."

- Igor Sysoev, NGINX creator and founder

MORE INFORMATION AT [NGINX.COM](https://nginx.com)

**"I wanted people to use it,
so I made it open source."**



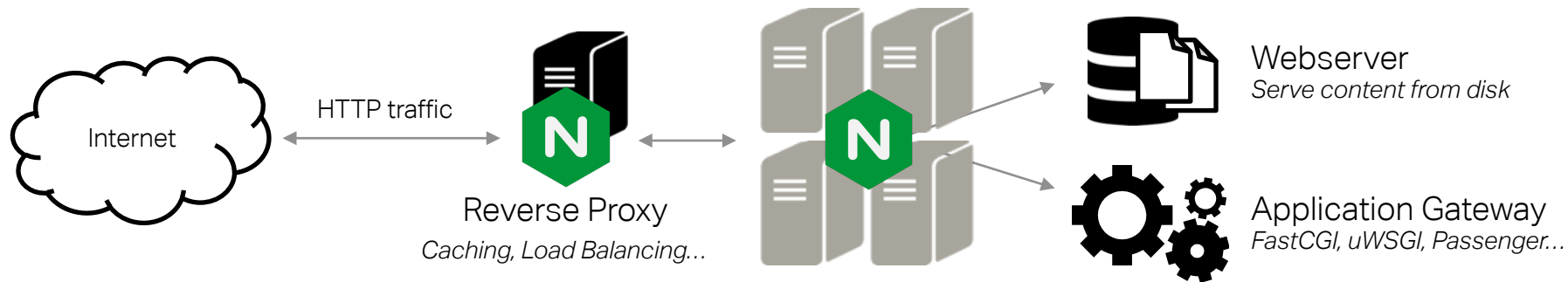
MORE INFORMATION AT [NGINX.COM](https://nginx.com)



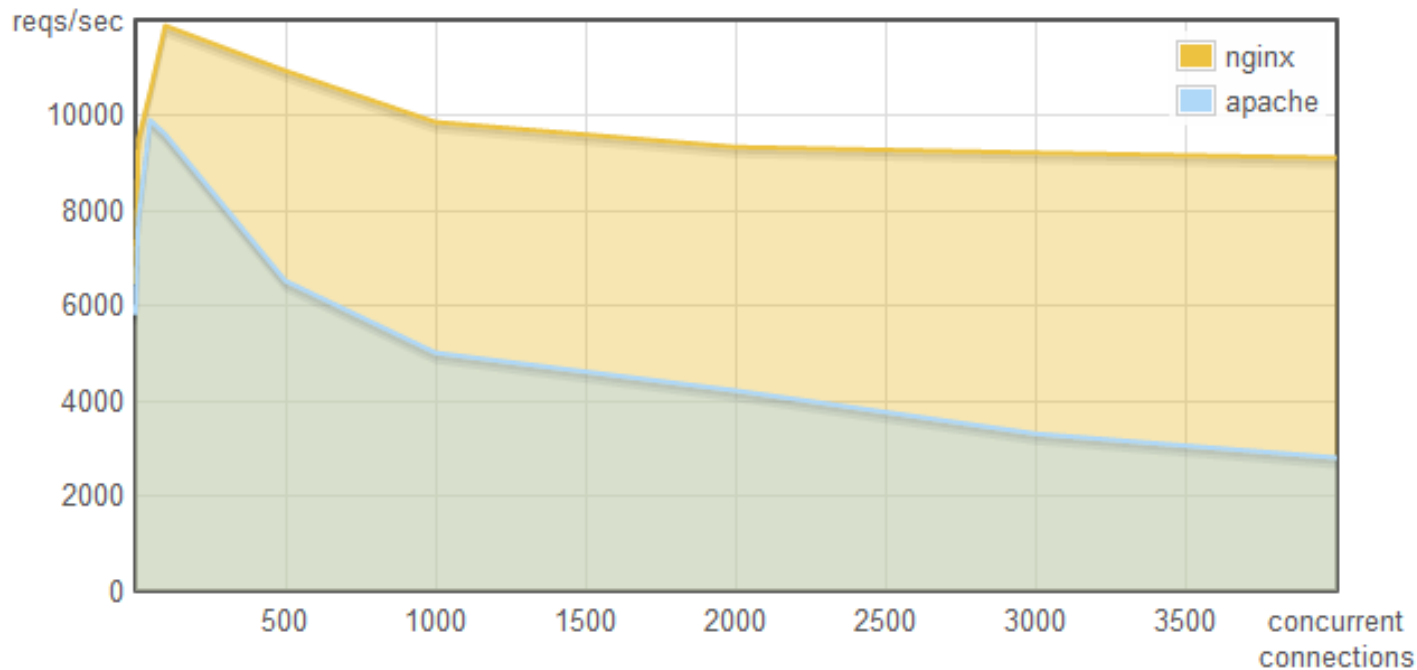
Igor Sysoev, NGINX creator and founder

- **First OSS release in 2004**
- **Company founded in 2011**
- **VC-backed by industry leaders**
- **500+ Customers**
- **75+ Employees**

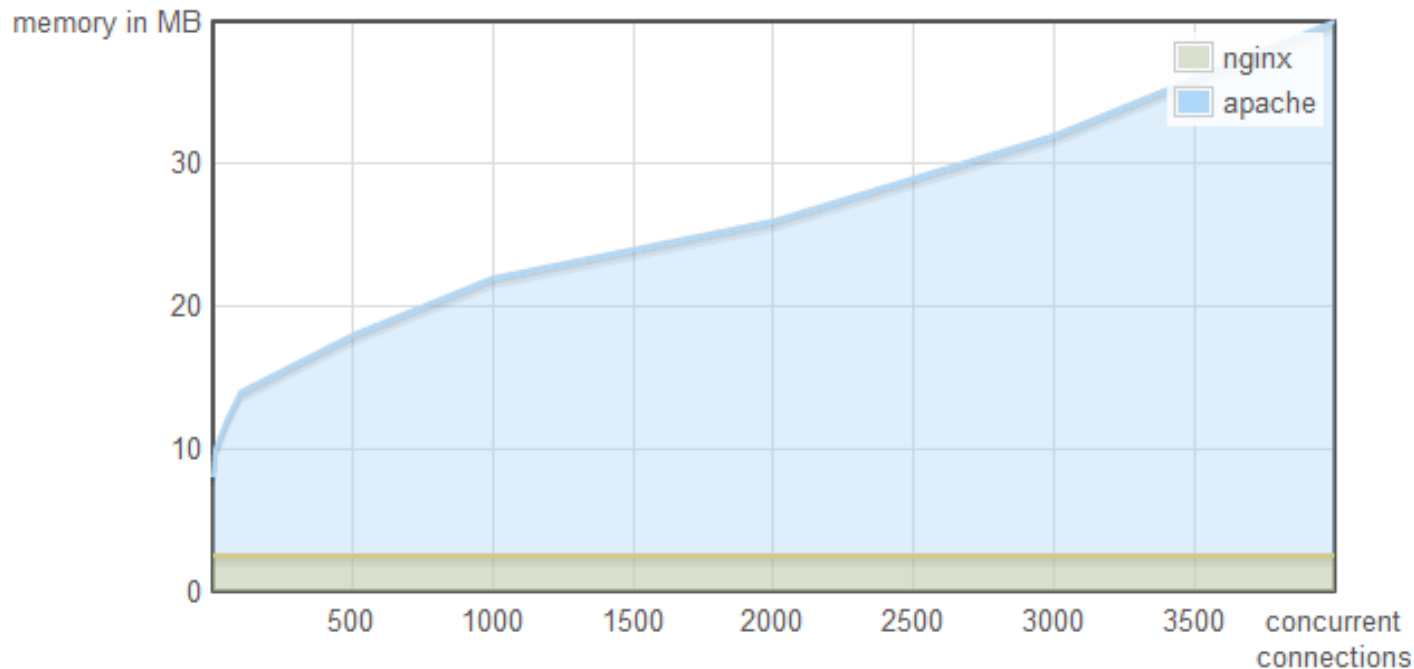
What is NGINX?



High Concurrency for Web Scale



Low Memory Use Drives Performance



Source: Webfaction Blog: <http://blog.webfaction.com/2008/12/a-little-holiday-present-10000-reqssec-with-nginx-2/>

A light gray background featuring a complex network of interconnected lines and dots, resembling a web or a molecular structure, with some dots highlighted in a slightly darker shade of gray.

**NGINX has become
the heart of the modern web**

160 million

**total sites
running on NGINX**

51%

of the Top 10,000
most visited websites

36%

of all sites on
Amazon Web Services



GitHub

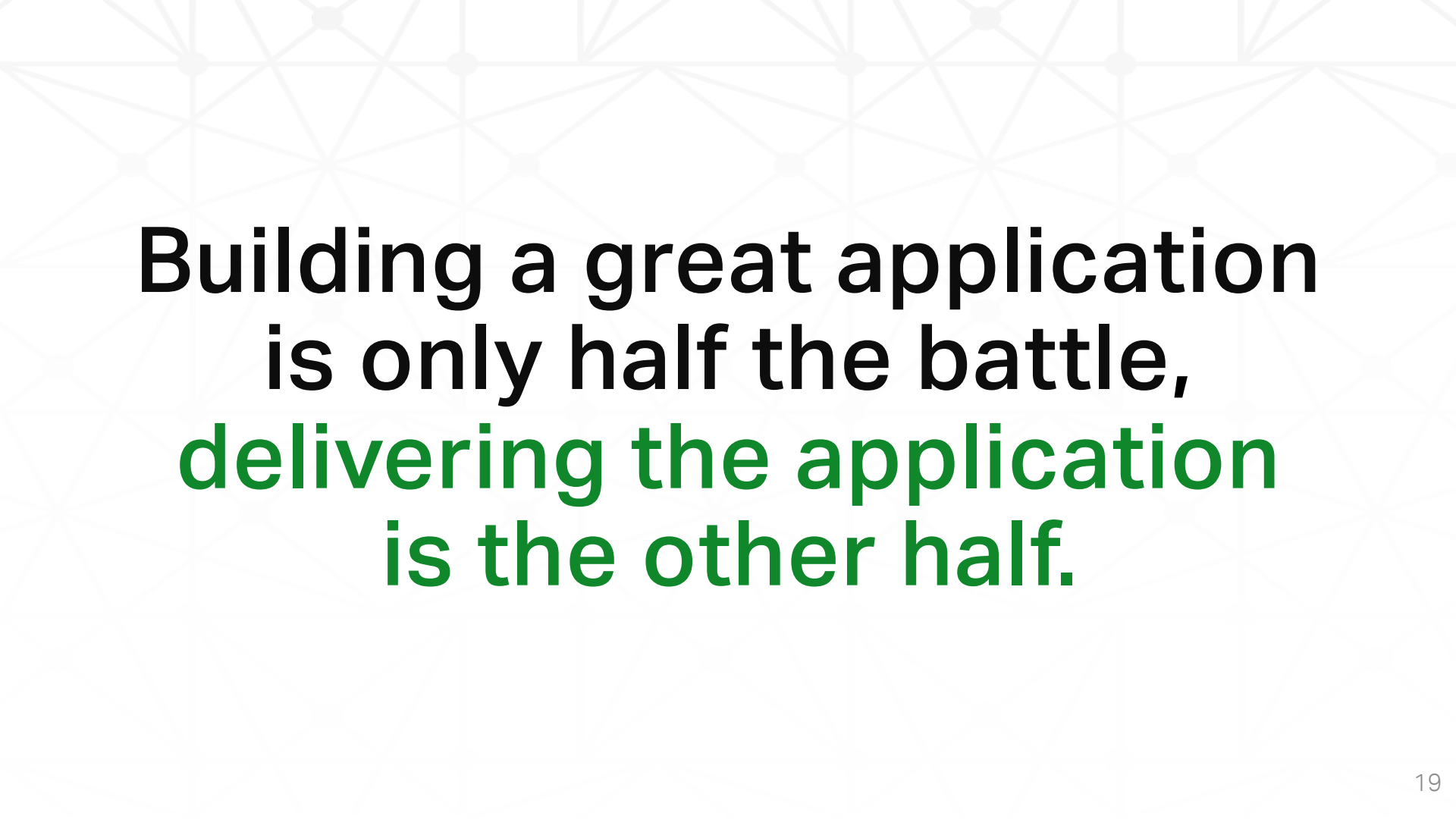


hulu



Pinterest



A light gray background featuring a network of interconnected nodes and lines, forming a complex geometric pattern of triangles and polygons.

**Building a great application
is only half the battle,
delivering the application
is the other half.**

The System is down at the moment.

We're working to resolve the issue as soon as possible. Please try again later.

Please include the reference ID below if you wish to contact us at 1-800-318-2596 for support.

Error from: https://www.healthcare.gov/marketplace/global/en_US/registration%23signUpStepOne

Reference ID: 0.cdd74f17.1380634949.2f9c301c



181 DAYS LEFT TO
ENROLL

OCT
1 Open
Enrollment
Began

JAN
1 Coverage
Can Begin

MAR
31 Open
Enrollment
Closes



#1: Load Balancing and Caching

- ✓ Application Load Balancing
 - Round Robin? Least time? Least Connections?
- ✓ Automated Failover
- ✓ Session Persistence and Slow Restart
- ✓ Advanced Cache Controls
- ✓ Content switching and request routing

#2: Security & Access Control

- SSL everywhere
- Authentication
- Whitelist, Blacklists, GEOIp
- Set up smart access limits:
 - Request limiting
 - Bandwidth controls
 - Limit connections



#3: Monitoring & Management

- ✓ Enable developers to control app delivery!
 - Manage configurations with every deployment
- ✓ On-the-fly load-balancing configuration
 - React in real time!
- ✓ No down time with live binary updates
- ✓ Live activity monitoring – via GUI, API, JSON
 - Enable remote logging (with syslog)

#4: Advanced Statistics

- Extended HTTP status ✓
- JSON Response ✓
- Plug-ins for Dynatrace, New Relic, DataDog, etc.

NGINX Plus feature

TCP Upstreams

mysql_backends 

Server		Connections			
Name	W	Total	Conn/s	Active	Limit
127.0.0.1:9306	1	24	0	0	∞
127.0.0.1:9307	2	1637535	0	0	∞
127.0.0.1:9308	3	0	0	0	40
b 127.0.0.1:9309	1	0	0	0	∞

[Dashboard](#)[Server zones](#)[Upstreams](#)[TCP Zones](#)[TCP Upstreams](#)[Caches](#)

Version **1.9.4**

Address **206.251.255.64**

PID **65836**

Uptime **1d 6h 37m**
.....

Connections Accepted: 2265884

Current	Accepted/s	Active	Idle	Dropped
56	26	22	34	0

Requests Total: 4165611

Current	Req/s
22	36

Server zones



Total Problems

3 / 0

Traffic

In: 2.98 KiB/s
Out: 93.3 KiB/s

Upstreams



Total Alerts

4 / 2

Servers

All: 7 / Up: 5
Failed: 2

TCP Zones



Conn total: 110077
Conn current: 0
Conn/s: 1

Traffic

In: 105 B/s
Out: 5.57 KiB/s

TCP Upstreams



Total Problems

1 / 0

Servers

All: 4 / Up: 3
Failed: 0

Caches



Total Warnings

1 / 1

Caches states

☀ Warm: 1
❄ Cold: 0

WordPress is Huge

About [a quarter](#) of all sites are now built on open-source WordPress software, including sites for eBay, Mozilla, RackSpace, TechCrunch, CNN, MTV, the New York Times, the Wall Street Journal.

WordPress.com, the most popular site for user-created blogs, also runs on WordPress open source software.

[NGINX powers WordPress.com.](#)

Easy, Hard, and Optimal Solutions

The **easy solution** is to totally re-imagine the situation and not change your core code and configuration at all. Costs some \$ but no trouble.

The **hard solution** is to change your webserver and reconfigure a bunch of stuff, without spending a dime.

The **optimal solution** is, all of the above.

The LAMP Bottleneck

Two main causes of bottlenecks are:

- **The Apache web server** – Apache consumes substantial resources for each and every connection. Memory can be exhausted and performance slows because data has to be paged back and forth to disk.
- **The PHP/MySQL interaction** – Together, an application server running PHP and a MySQL database server can serve a maximum number of requests per second. When the number of requests exceeds the maximum, users have to wait. Exceeding the maximum can cause significant performance problems.

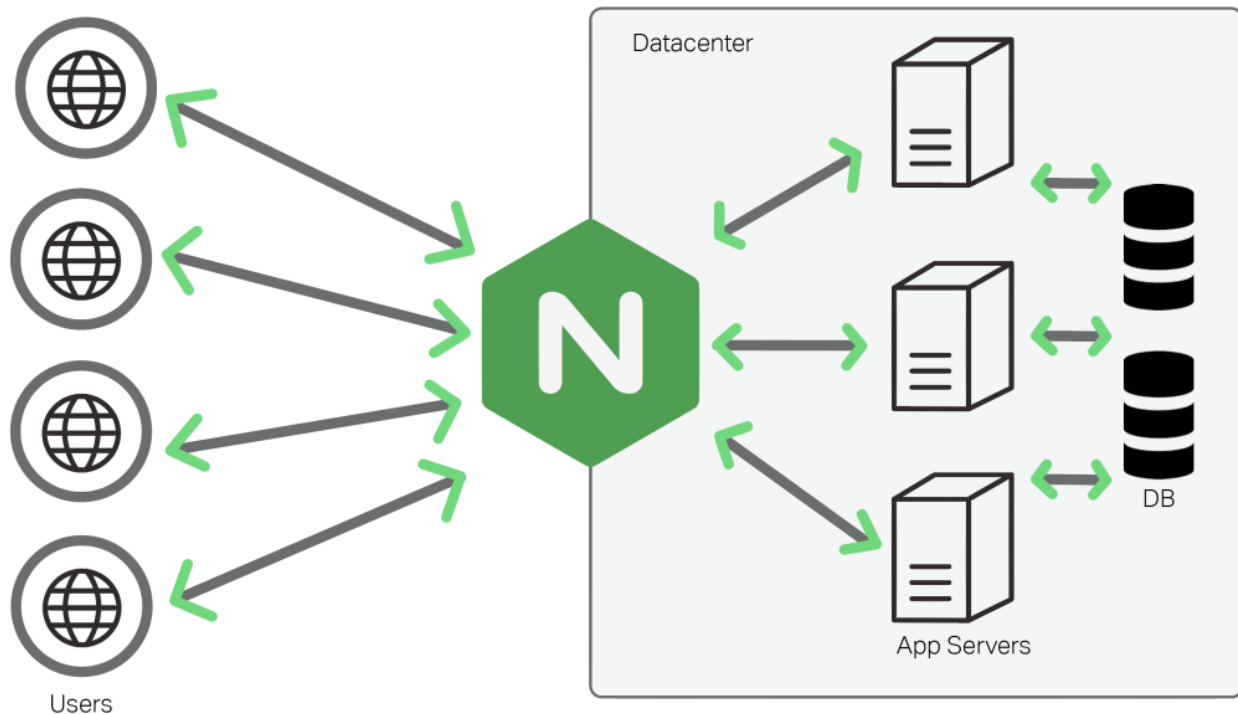


MORE INFORMATION AT [NGINX.COM](https://nginx.com)

Reverse Proxy Server Plus Caching

Three tricks in one:

- **Reverse Proxy Server** – Move user I/O to an external server.
- **Cache Static Files** – Move JPEGs, CSS files etc. to the reverse proxy server.
- **Cache Dynamic Files** – Get current content for non-logged-in users once per second. Then for 10/50/100 requests per second, serve the (briefly) cached file.



MORE INFORMATION AT [NGINX.COM](https://nginx.com)

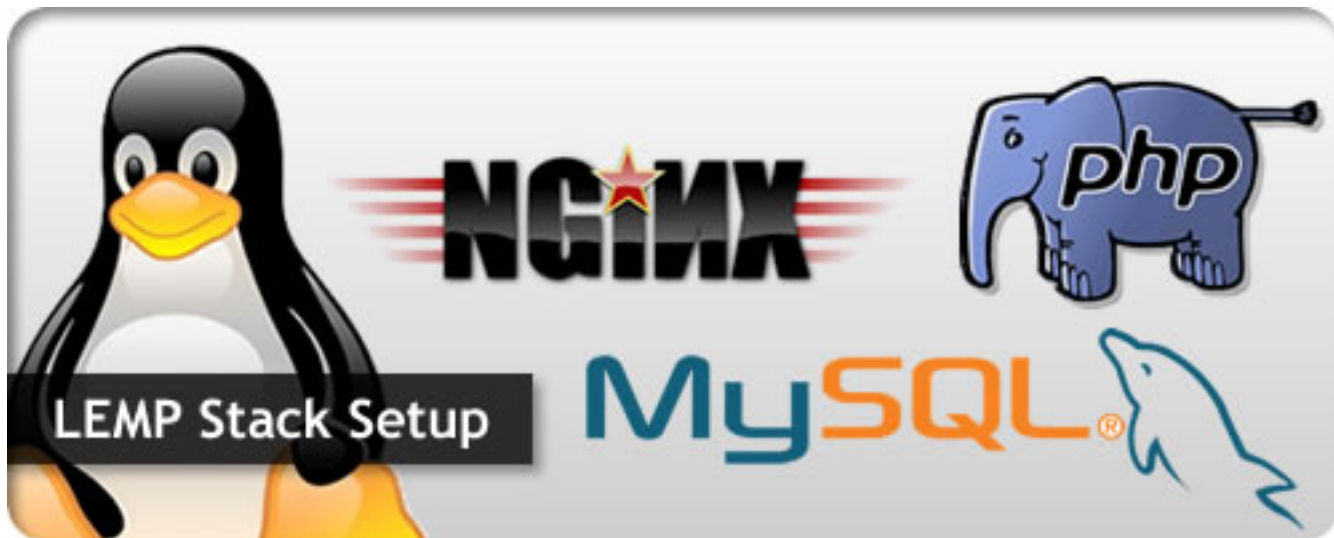
What's inside NGINX Plus?



The LEMP Solution

Two steps to LEMP:

- **Replace LAMP (Linux-Apache-MySQL-PHP) with LEMP (Linux-(e)NGINX-MySQL-PHP)**
 - NGINX is faster and more stable.
- **Make Needed Updates:**
 1. Permalinks instead of .htaccess
 2. Caching with FastCGI Cache etc.
 3. Implement security precautions
 4. Configure WordPress Multisite w/proper file structure



1. Permalink Support for NGINX

- **Why .htaccess is bad**
- **NGINX config**

```
server {  
    server_name example.com www.example.com;  
    root /var/www/example.com/htdocs;  
    index index.php;  
  
    access_log /var/log/nginx/  
example.com.access.log;  
    error_log /var/log/nginx/  
example.com.error.log;  
    location / {  
        try_files $uri $uri/ /index.php?  
$args;  
    }  
}
```

2. Configure NGINX for FastCGI (or ...)

- **FastCGI is most popular and most flexible**
- **W3 Total Cache, WP Super Cache, and others work well too**

```
fastcgi_cache_path /var/run/nginx-cache levels=1:2
                    keys_zone=WORDPRESS:100m inactive=60m;
fastcgi_cache_key "$scheme$request_method$host$request_uri";
```

```
server {
    server_name example.com www.example.com;
    root /var/www/example.com/htdocs;
    index index.php;
    access_log /var/log/nginx/example.com.access.log;
    error_log /var/log/nginx/example.com.error.log;
    set $skip_cache 0;
    # POST requests and URLs with a query string should always go to PHP...
```

MORE INFORMATION AT [NGINX.COM](https://nginx.com)

3. Secure NGINX

- **Best to do this at reverse proxy server *and* at webserver**

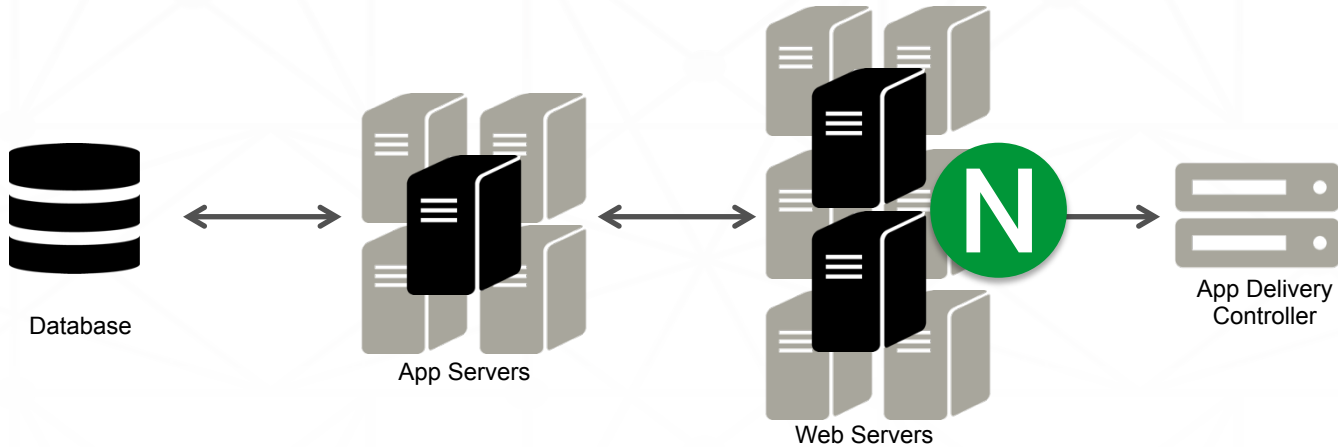
```
# Restrict access to WordPress dashboard
```

```
location /wp-admin {  
    deny 192.192.9.9;  
    allow 192.192.1.0/24;  
    allow 10.1.1.0/16;  
    deny all;
```

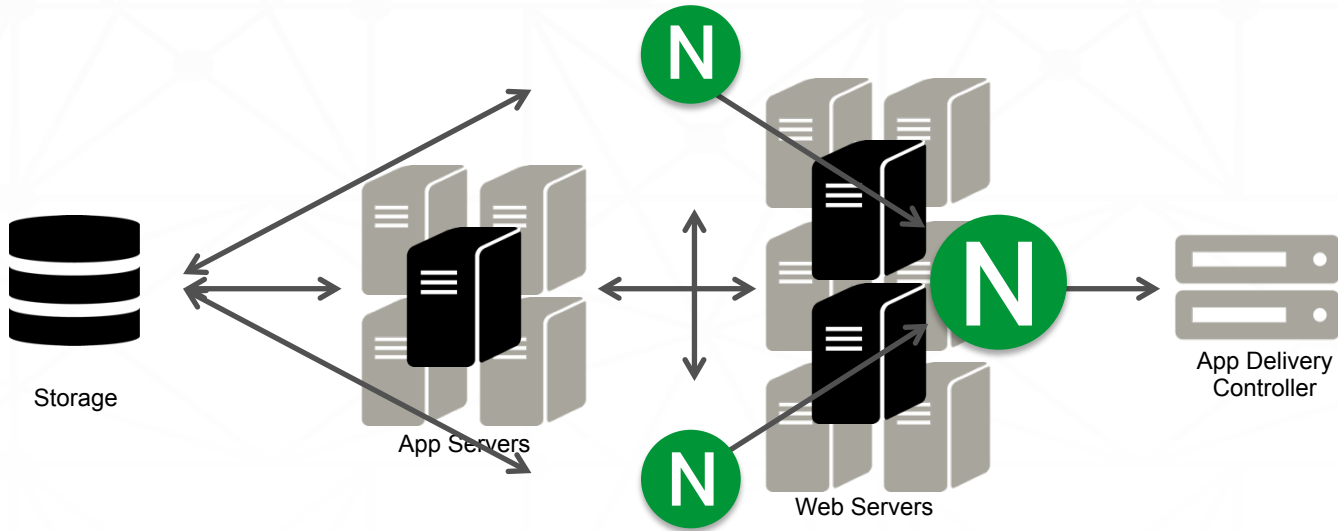
```
# Deny access to uploads that aren't images, videos, music, etc.
```

```
location ~* ^/wp-content/uploads/.*.(html|htm|shtml|php|js|swf)$ {  
    deny all;  
}
```

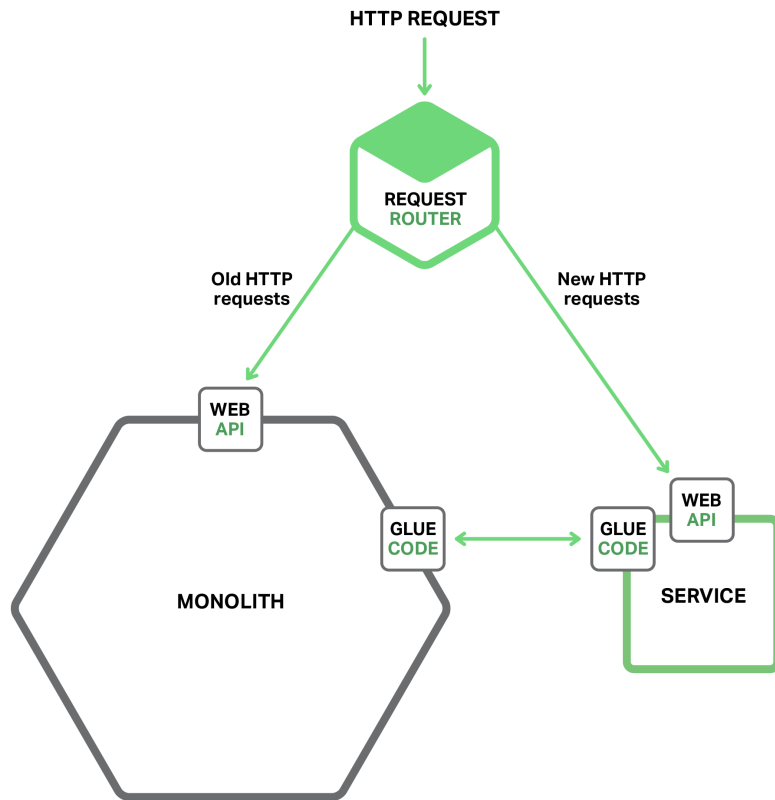
You can add us to your current stack today



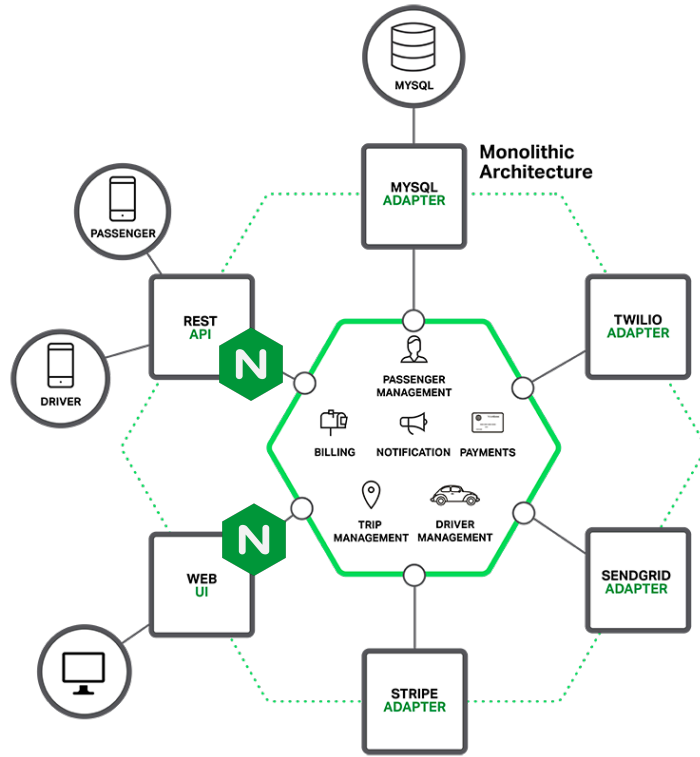
You can add us to your current stack today



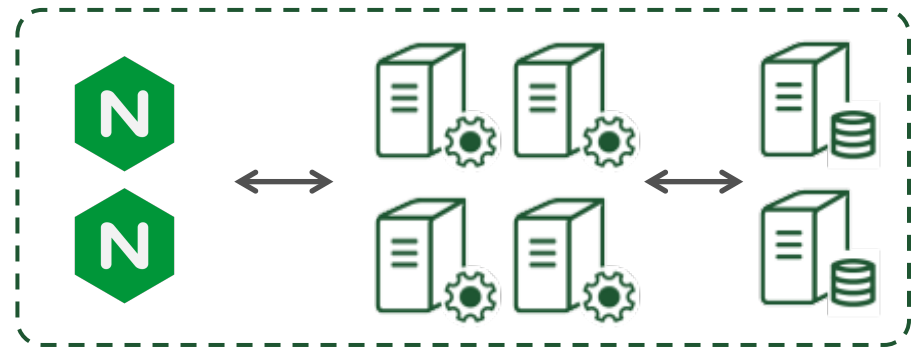
Or... Eff it, let's just do microservices



Moving from monolith to microservices



Architecture



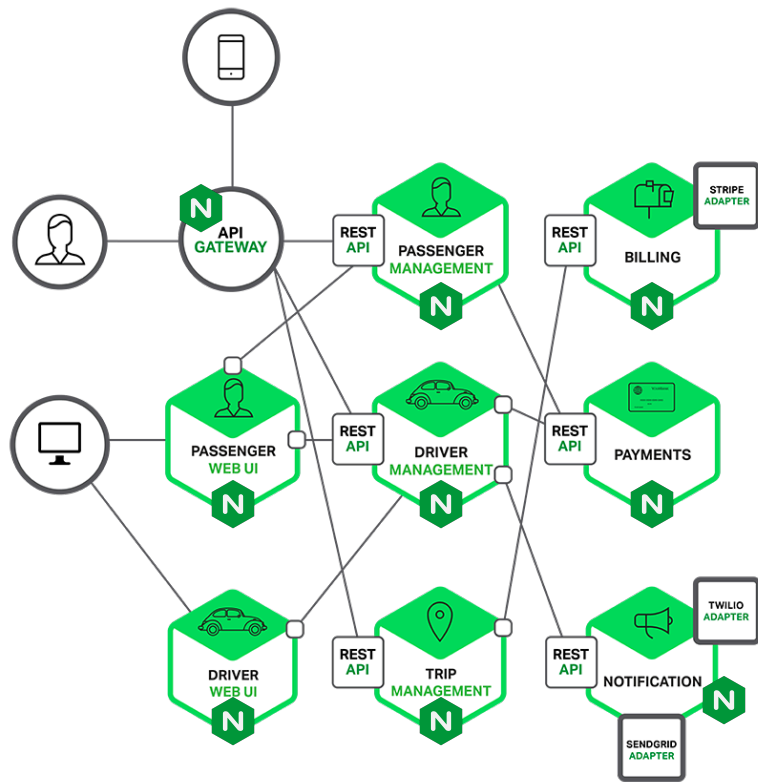
Application
Delivery

Monolithic
Application

Clustered
Database

Deployment
Platform

Application delivery for microservices



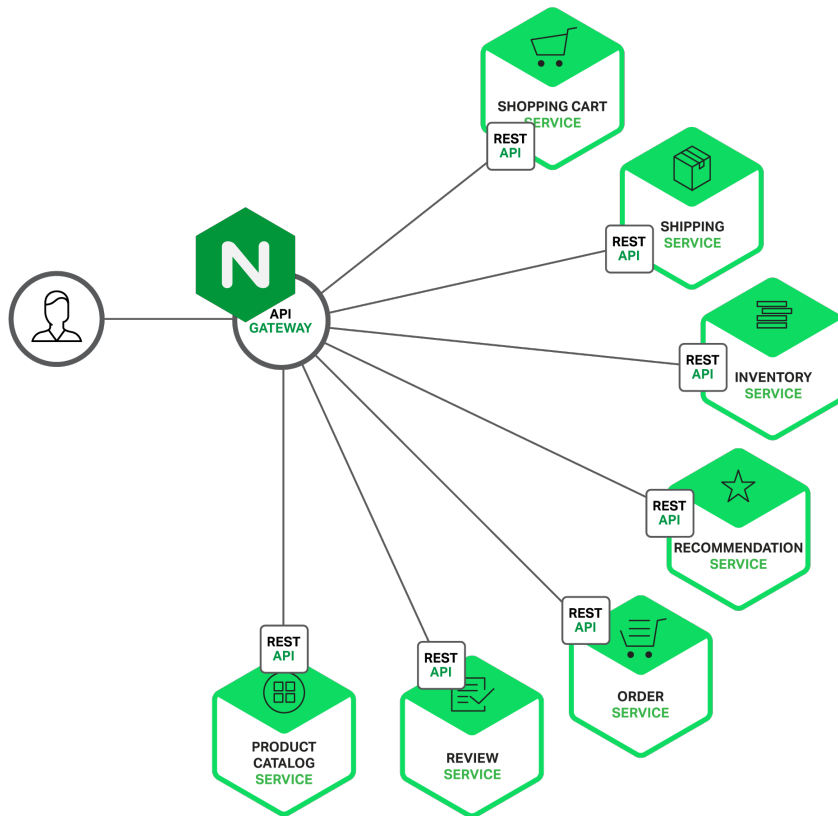
NGINX is in each microservice, ensuring they are:

- Connected
- Served
- Authenticated
- Secured
- Cached
- Load Balanced
- Scaled

You can also use NGINX in your API Gateway

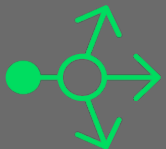
NGINX Plus provides:

- Integration
- Security
- Request Routing
- Optimization
- Scalability

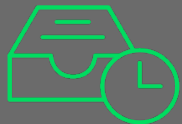




Flawless application delivery made easy



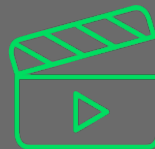
Load Balancer



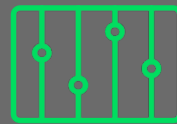
Content Cache



Web Server



Streaming Media



Monitoring &
Management

Thank You

nginx.com | [@nginx](https://twitter.com/nginx)

