

NGINX



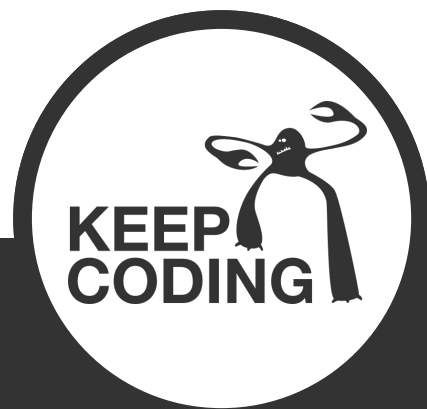
El todo en uno: servidor web, proxy inverso y balanceador de carga

■ Nginx

- Creado en 2002 por Igor Sysoev
- Nace como respuesta al problema C10K: manejar 10.000 conexiones simultáneas
- Muy bajo consumo de recursos (frente a Apache)
- Fácil de configurar (frente a Apache)
- No procesa archivos dinámicos (como PHP o Python), delega esta tarea en servidores de aplicación y actúa como proxy inverso

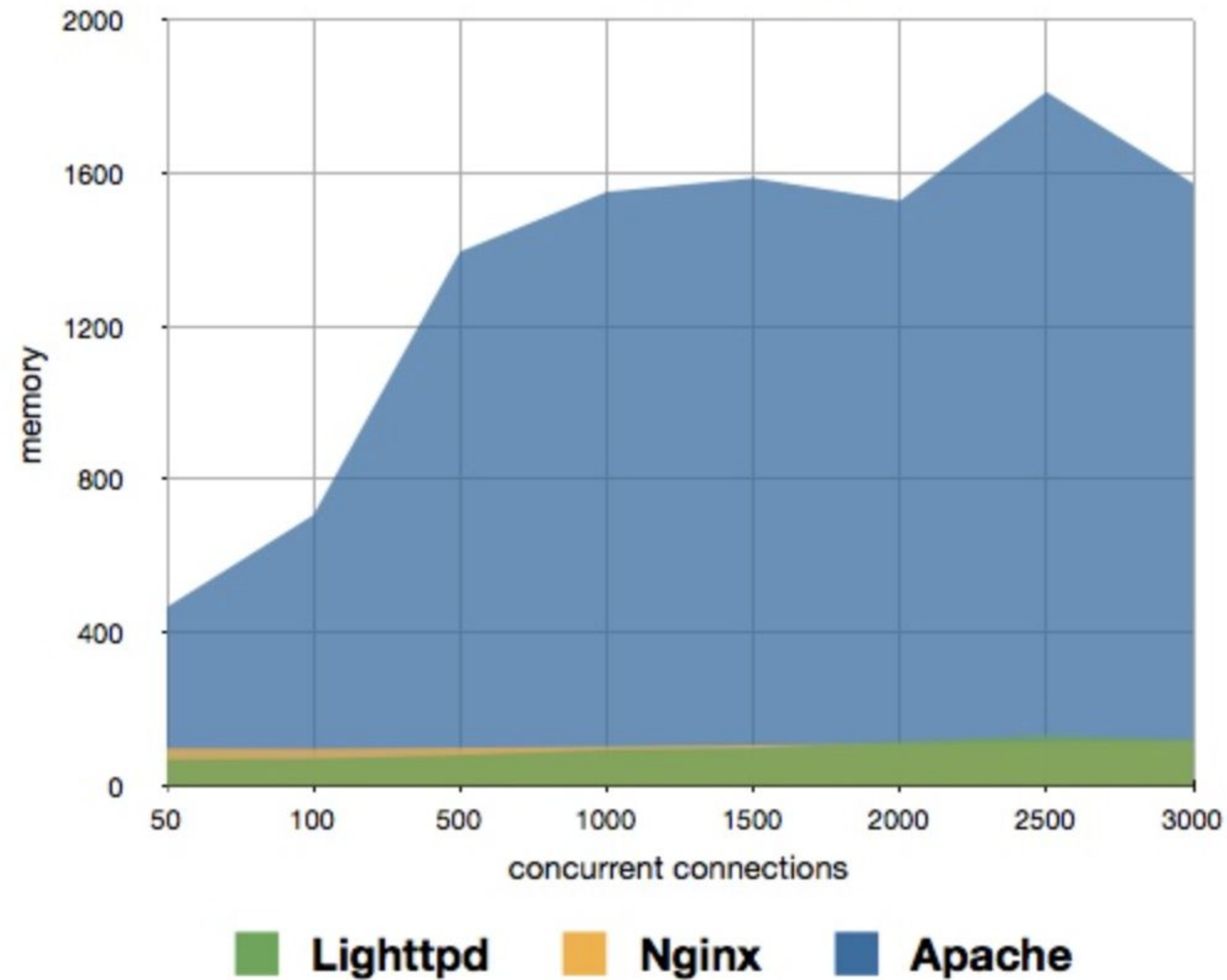


Igor

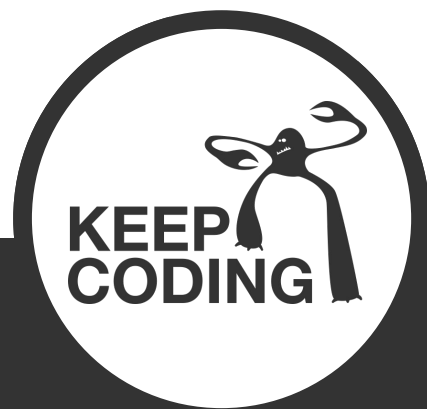
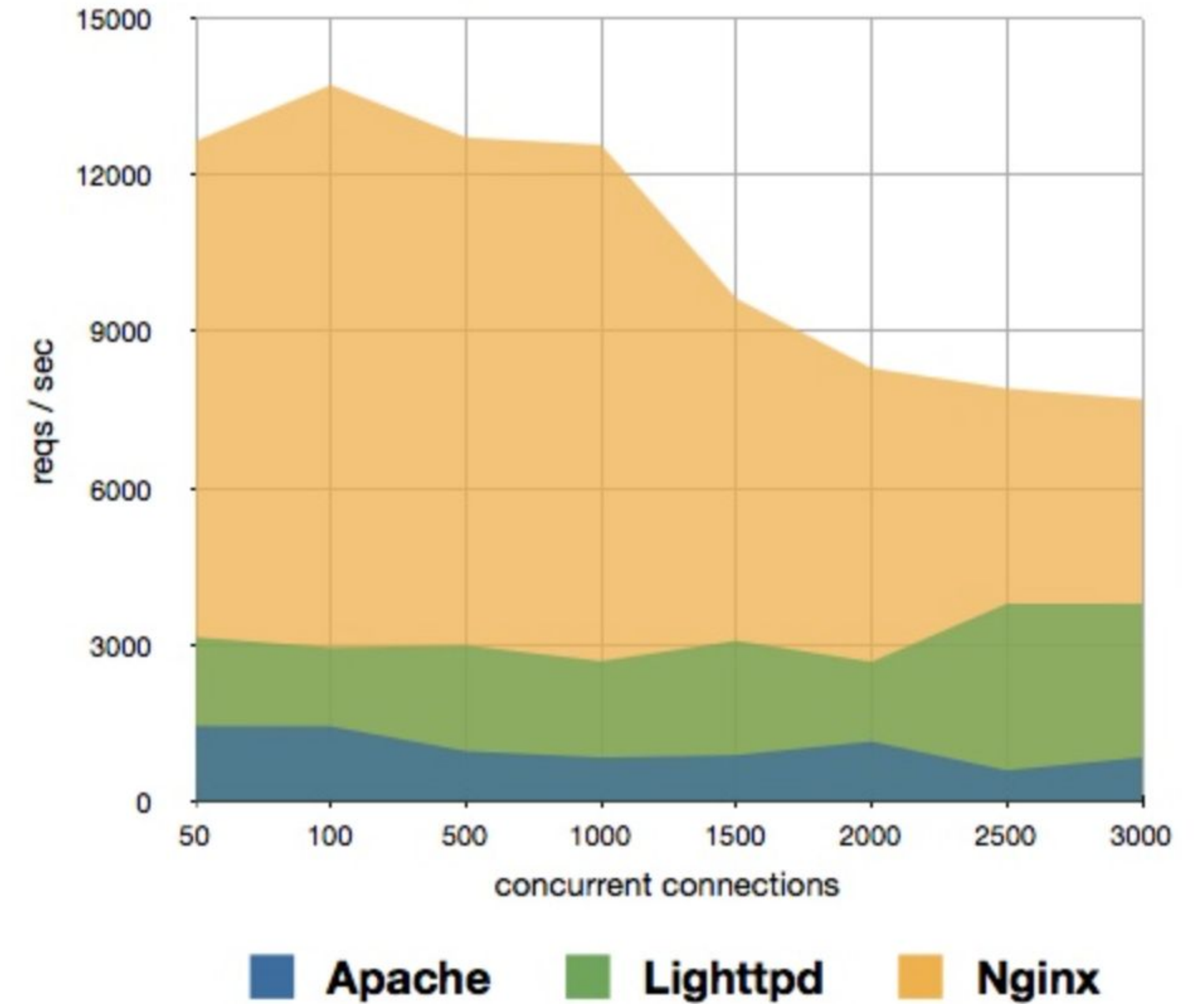


■ Nginx vs Apache

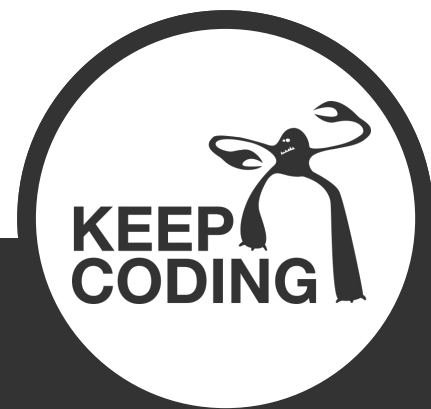
Memory Usage



Requests Per Second

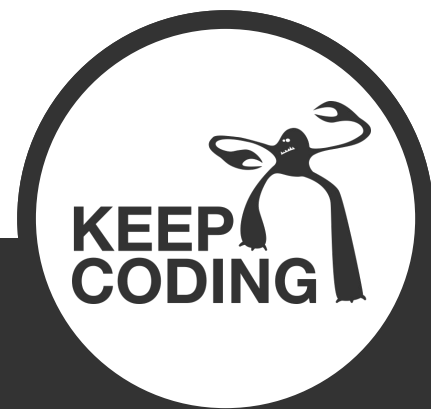


■ Instalación



■ Instalación

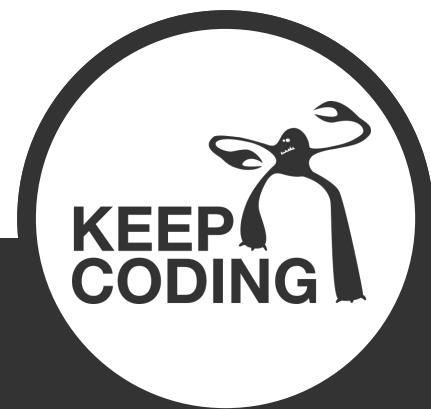
```
sudo apt-get install nginx
```



■ Comprobar si se está ejecutando

`sudo service nginx status`

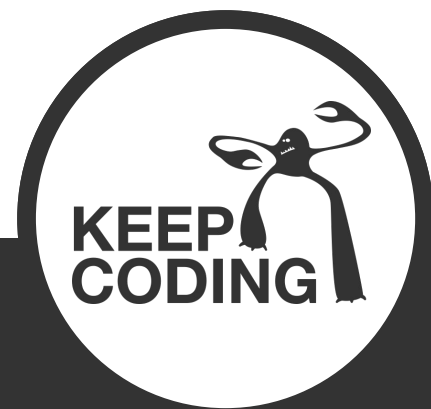
`sudo systemctl status nginx`



■ Parar nginx

```
sudo service nginx stop
```

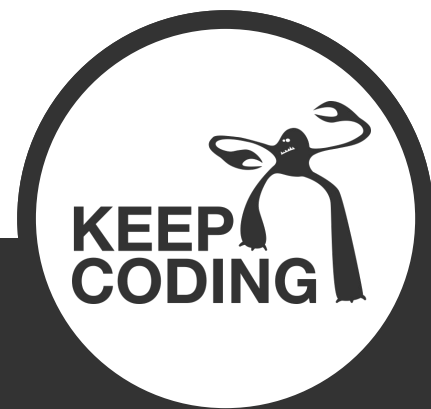
```
sudo systemctl stop nginx
```



■ Arrancar nginx

```
sudo service nginx start
```

```
sudo systemctl start nginx
```



■ Reiniciar nginx

```
sudo service nginx restart
```

```
sudo systemctl restart nginx
```

Útil cuando modificamos su configuración

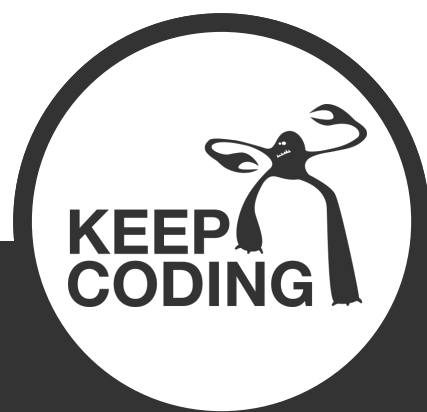


■ Recargar configuración sin reiniciar

```
sudo service nginx reload
```

```
sudo systemctl reload nginx
```

Útil cuando modificamos su configuración. Preferible a restart.



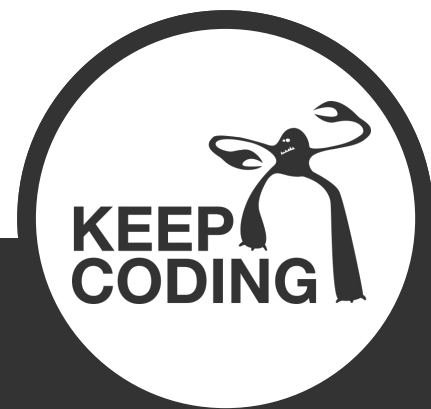
■ Probar configuración

`sudo nginx -t`

Útil cuando modificamos su configuración

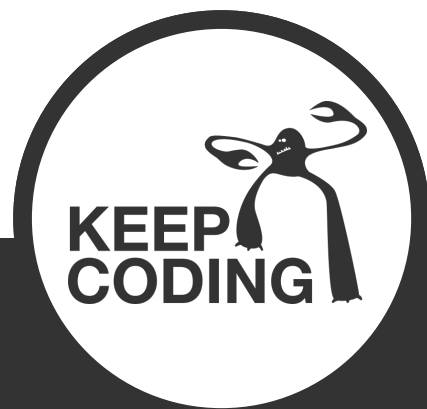


■ Configuración



■ Configuración

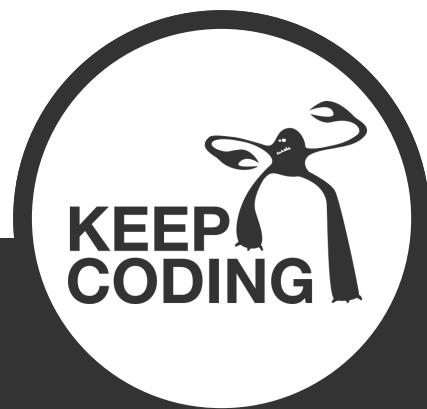
- El fichero de configuración está en: `/etc/nginx/nginx.conf`
- En `/etc/nginx/conf.d/` podemos incluir configuración personalizada que sobrescriba parámetros por defecto (si no queremos tocar `/etc/nginx/nginx.conf`)



■ nginx.conf

```
user www-data; # usuario que ejecuta el proceso nginx
worker_processes 4; # tantos como cores tenga el servidor
pid /run/nginx.pid;

events {
    worker_connections 768; # max conexiones por worker
    # multi_accept on; # aceptar más de una conexión nueva a la vez
}
```



■ nginx.conf

http {

sendfile on; # http://nginx.org/en/docs/http/nginx_http_core_module.html

tcp_nopush on; 🙌

tcp_nodelay on; 🙌

keepalive_timeout 65; # timeout para cerrar conexiones keepalive

types_hash_max_size 2048; # <http://nginx.org/en/docs/hash.html>

include /etc/nginx/mime.types; # incluye conf. aburrida de mimetypes

default_type application/octet-stream; # mimetype por defecto

access_log /var/log/nginx/access.log; # log de accesos

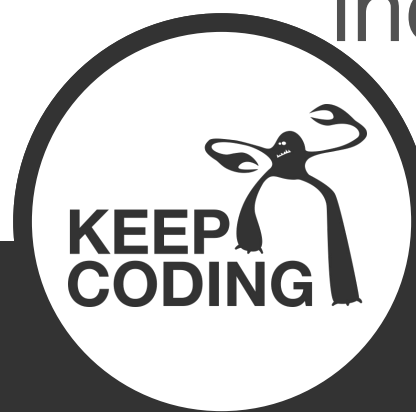
error_log /var/log/nginx/error.log; # log de errores

gzip on; # habilita la compresión gzip

gzip_disable "msie6"; # deshabilita la compresión gzip para IE6 💩

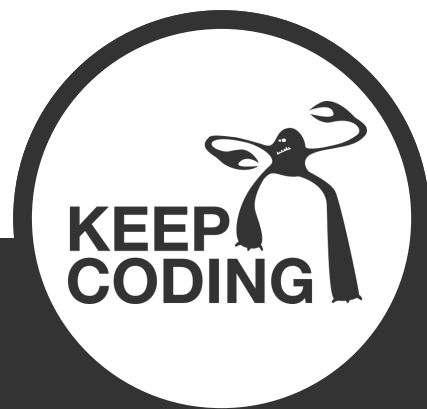
include /etc/nginx/conf.d/*.conf; # incluye otros archivos de config.

include /etc/nginx/sites-enabled/*; # incluye config. de sitios web



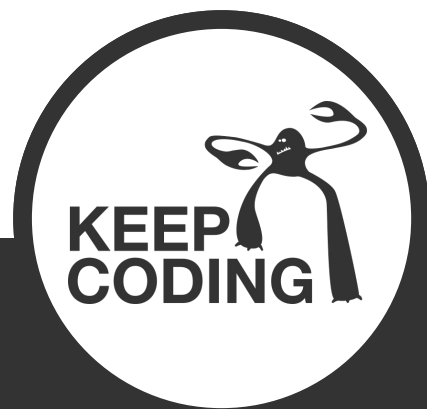
■ Configuración de nuestros sitios web

- Con una sola instalación, podemos dar servicio a varios dominios o subdominios
- En `/etc/nginx/sites-available/` podemos incluir configuración de otros sitios que queramos configurar
- Una vez configurados, debemos hacer un **acceso directo** (`ln -s`) el archivo a `/etc/nginx/sites-enabled/` para que nginx los sirva

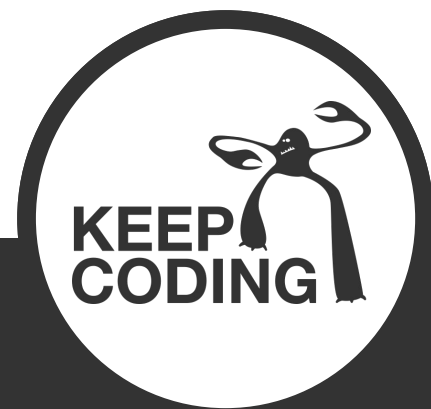


■ Fichero de configuración de sitios

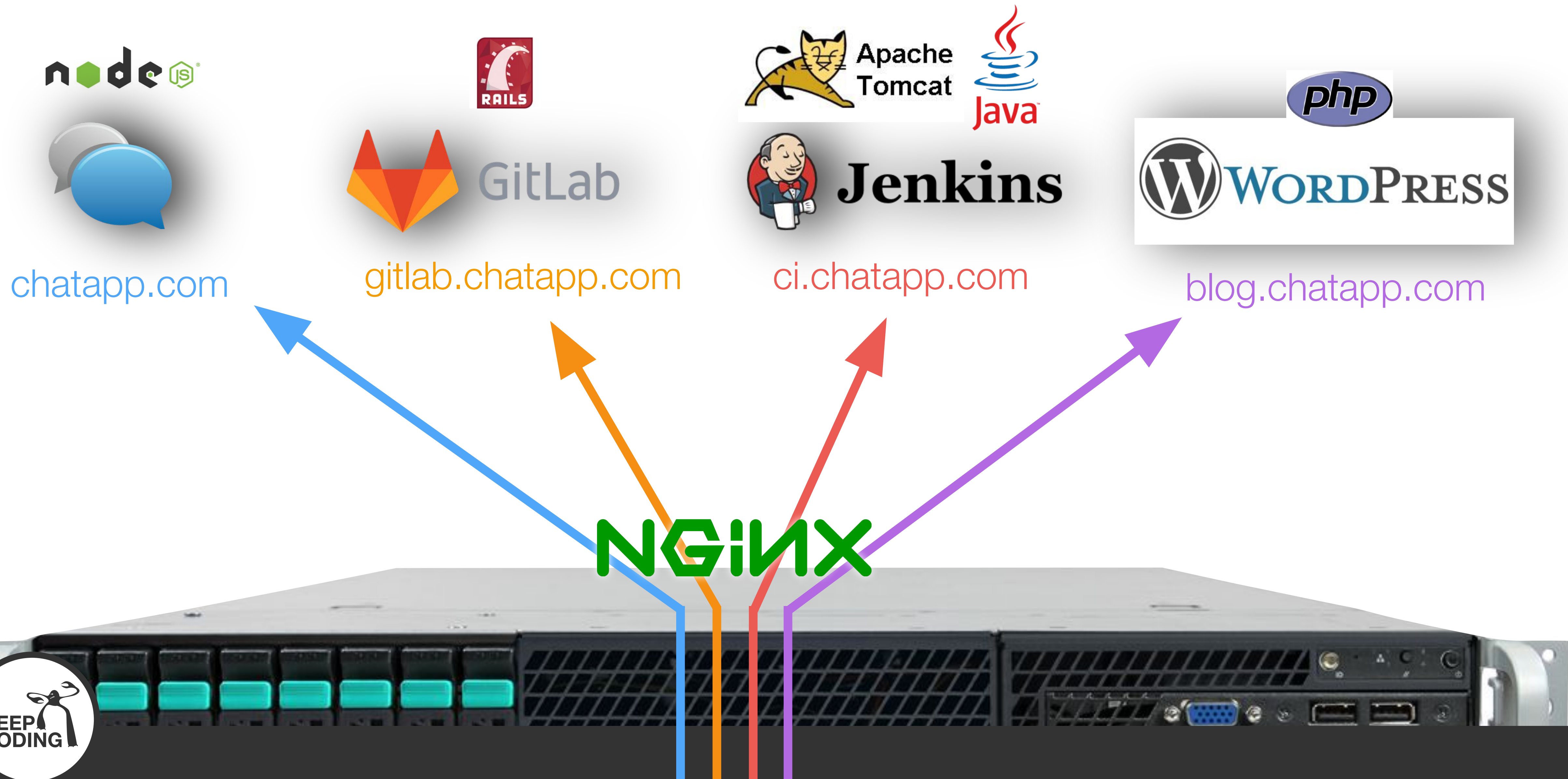
```
server {  
    # puerto en el que escucha ipv4, default_server (server por defecto, único)  
    listen 80 default_server;  
    # puerto en el que escucha ipv6 (sólo puede haber un ipv6only=on)  
    listen [::]:80 default_server ipv6only=on;  
    # ruta de donde servir los archivos estáticos  
    root /usr/share/nginx/html;  
    # archivos considerados como índices  
    index index.html index.htm;  
    # nombre del servidor (dominio o dominios)  
    server_name localhost;  
    # cuando la ruta empiece por / buscar localmente la ruta tal cual  
    # o terminada por /, si no encuentra nada, devolver un 404  
    location / {  
        try_files $uri $uri/ =404;  
    }  
}
```



■ Nginx como proxy inverso



■ Nginx como proxy inverso



■ Fichero de configuración de sitios

```
server {  
  
    # sirviendo archivos estáticos desde nginx  
    location ~ ^/(css/|img/|js/|sounds/) {  
        root /var/www/app/public; # ruta base al directorio de estáticos  
        access_log off; # no dejar log de acceso a estáticos (no aportan info)  
        expires max; # máximo de expiración (cache)  
    }  
  
    # cualquier cosa que no sea estático, se lo pasamos a node  
    location / {  
        proxy_set_header Host $http_host;  
        proxy_pass http://127.0.0.1:3000; # pasamos a node la petición  
        proxy_redirect off; # no redirigimos  
    }  
}
```

