Web cache

NGINX / VARNISH

Découverte

Par définition:

Permet de stocker temporairement des documents web

Pourquoi faire:

Optimiser:

la bande passante

la charge du serveur

les latences

Service Similaire:

Le CDN

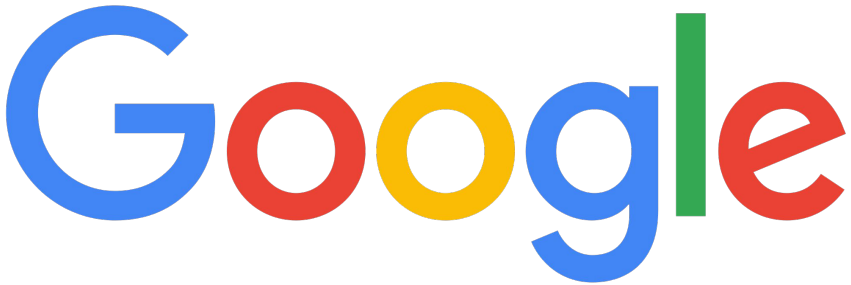
(Content Delivery Network)

https://www.ovh.com/fr/images/videos/cdn.xml

Le web cache, c'est vieux comme l'internet

Vous l'utilisez tous les jours dans vos navigateurs web

Un simple logo:



14kb

2.3 Million de recherche par seconde

Sans web cache:

14kb * 2.3 million par seconde

=

32.2 GB / seconde Ou 278 Tb / jour

Le web cache de votre navigateur garde le logo localement afin de ne pas le recharger à chaque accès

On appelle cela le cache internet côté client
Ou “Internet Temporary Files”

Le web cache côté serveur utilise le même concept

NGINX et VARNISH

Sont des applicatifs serveurs permettant la gestion de web cache

NB: NGINX n'est pas spécifique pour le webcaching, mais supporte les fonctionnalités exposées.

Voyons comment tout cela fonctionne

Le concept

Site pour un événement musical

Site Wordpress

MySQL + Apache + php5

Images, html, css et javascript

100 visiteurs par jour au lancement

Jusqu'ici, tout va bien

Mais à l'approche de l'événement

Le trafic augmentera

Avec ses heures de pointes

Un site web fonctionnant sous Wordpress

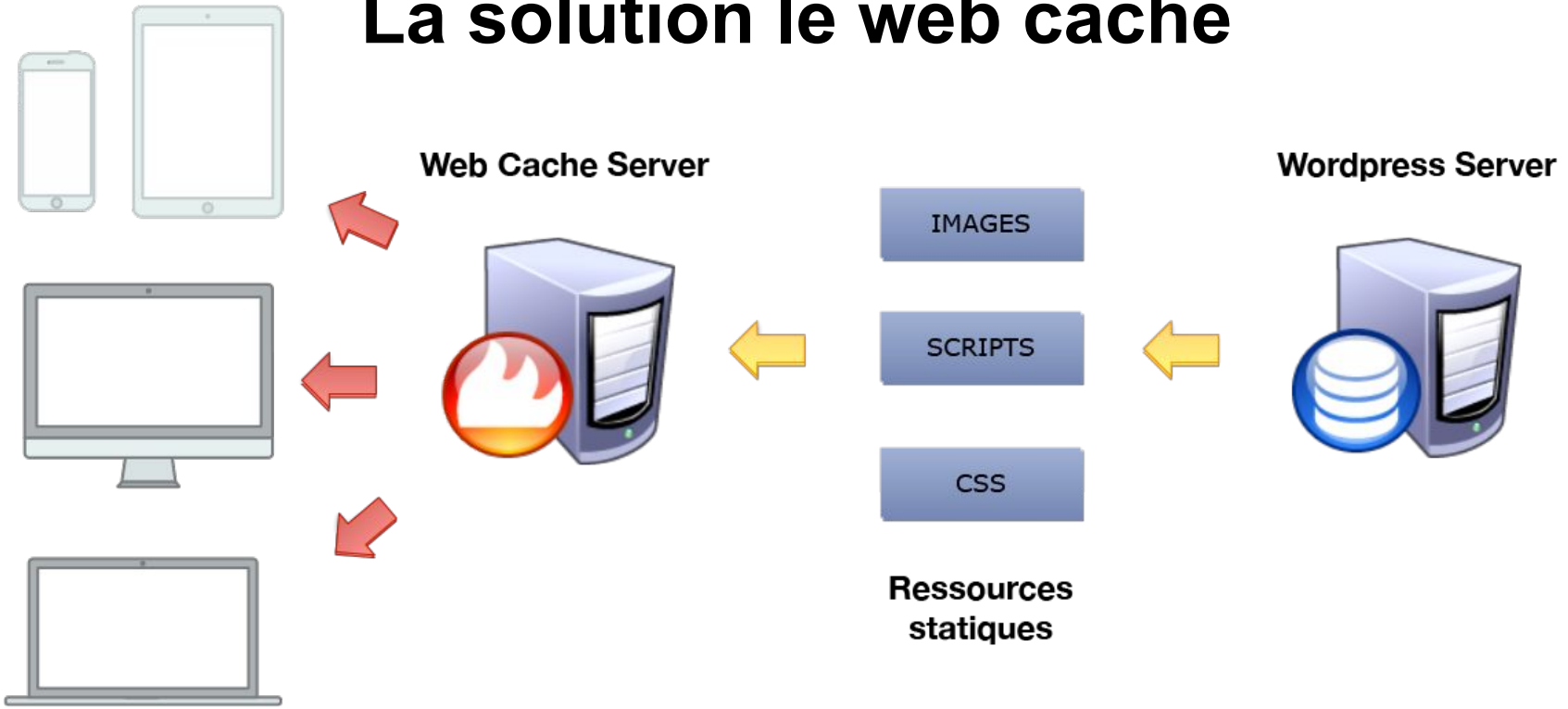
100 requêtes par page vue

4 secondes de chargement

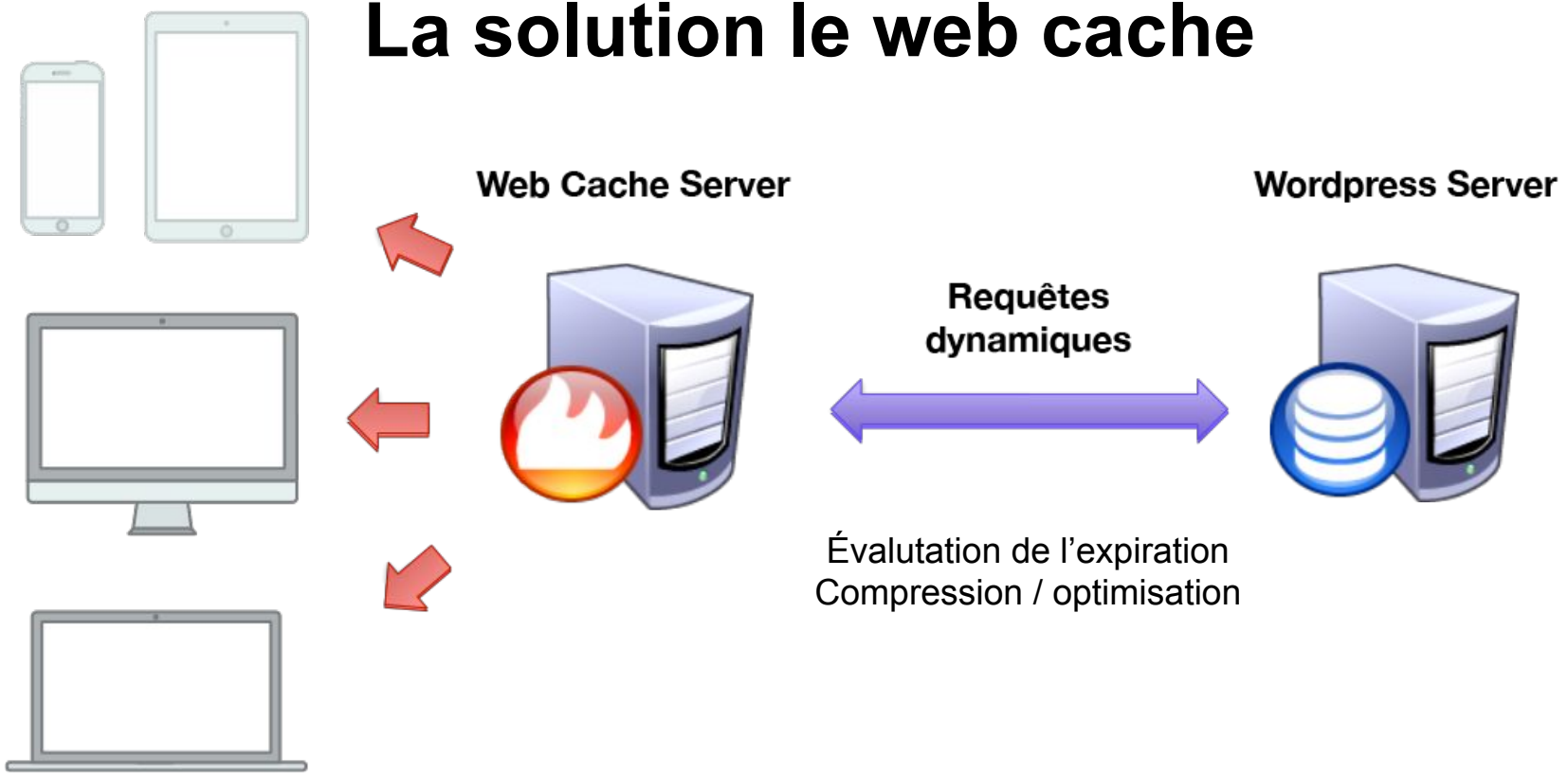
TOUT ÇA POUR UN SIMPLE UTILISATEUR

Quelques dizaines d'utilisateurs simultanés pourront bloquer votre site

La solution le web cache



La solution le web cache



La solution le web cache

Cache serveur

Ce qu'automatise le web cache

Minification et compressions (GZ) des scripts et HTML

Optimisation (PNG) et Compressions des images (JPEG)

Gestion avancée des headers HTTP (cache control / expire …)

Routage simple, pas de traitement de ressource

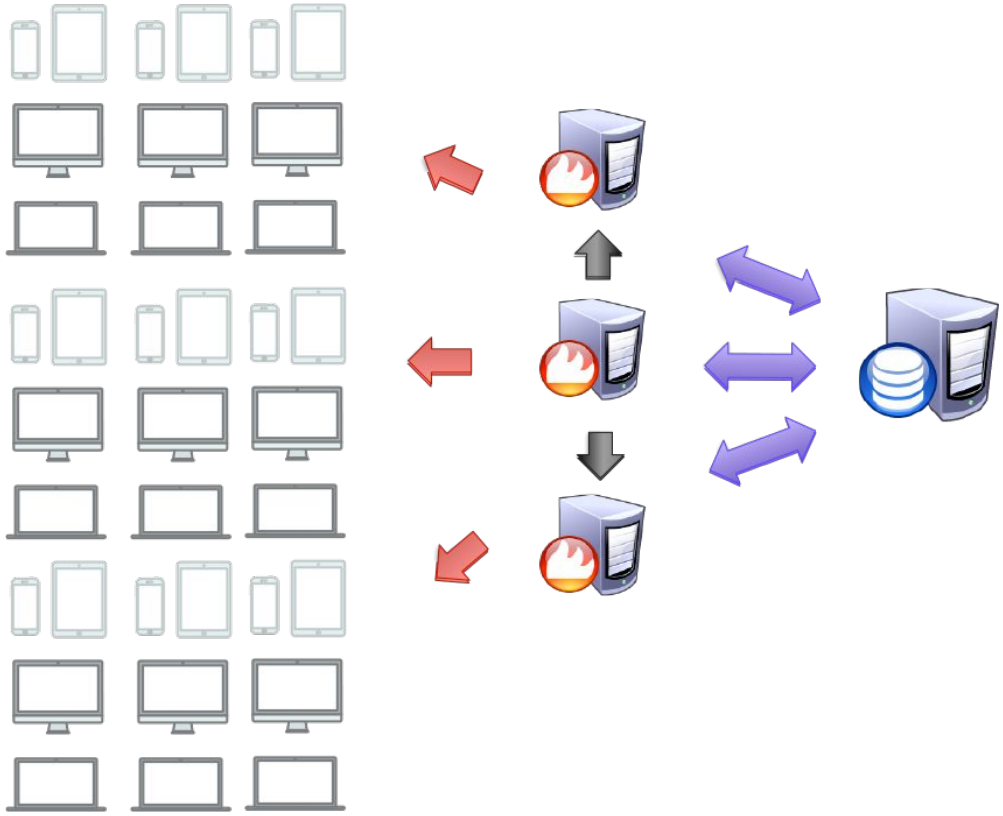
Filtres précis via les URLs

N'affecte pas les régies pub, vidéos youtube ou ressources tierces

La solution le web cache

Niveau 2

La répartition de charge (Load Balancing)



Load Balancing

Répartition de la charge sur plusieurs serveurs

Répartition intelligente des ressources Master / slave

Vérification de la santé des noeuds (Health Polling)

Le Load Balancing

Routing Hardware

Couteux
Talon d'Achille
Ingérable en cloud

Round Robin DNS

Flexible
Robuste
Adapté au cloud ET aux infras classiques

Le round Robin DNS

¿ KeSaKo ?

Le round Robin DNS

C'est très simple,
Mais attention à ce que vous faites

DNS Normal

www.monsite.com -> 80.231.67.88

Le round Robin DNS

www.monsite.com -> 80.231.67.88 ⇦ **Cache serveur # 1**
-> 80.231.67.92 ⇦ **Cache serveur # 2**
-> 92.123.87.93 ⇦ **Cache serveur # 3**
-> 92.123.87.94 ⇦ **Cache serveur # 4**
-> ...

Le round Robin DNS

3 petits conseils:

N'utilisez pas des adresses IP mais des CNAME

Reduisez le TTL à quelques minutes

Utilisez des services DNS tiers

Revenons à l'essentiel

Quand utiliser quoi !

CDN | CONTENT | DELIVERY | NETWORK | NGINX | VARNISH | DNS | ROUND ROBIN | COMPRESSION | MINIFIER | STATIQUE | DYNAMIQUE | IMAGES | SCRIPTS

Revenons à l'essentiel

Si votre site à moins de 1000 Utilisateurs jours

N'utilisez pas de web cache
D'autres solutions vous permettront d'optimiser votre site.

Revenons à l'essentiel

CDN

Votre site utilise beaucoup d'images
L'ajout d'image n'est pas ouvert au public
Vous n'utilisez pas de CMS lourd (Usine à gaz)

Revenons à l'essentiel

NGINX

Votre site utilise des requêtes lourdes
Vous diffusez de la video en streaming
Vos images et vos scripts sont optimisés avant la mise en ligne

Revenons à l'essentiel

VARNISH

Votre site propose des images mise en ligne par vos visiteurs
Vous utilisez de nombreuses librairies que vous ne controlez pas
Vous utilisez un CMS lourd (Usine à gaz)

Quelques astuces

Avec quelques astuces vous pouvez faciliter
l'intégration de web cache lorsque que cela sera necessaire.

Voir même vous affranchir d'en utiliser un.

Quelques astuces

Pensez à utiliser un sous-domaine spécifiques pour vos ressources statiques.

Exemple:

http://www.monsite.com/img/image.jpg

http://images.monsite.com/img/image.jpg

Quelques astuces

Utilisez et respectez une arborescence structurée

img/
css/
js/

Quelques astuces

Minifiez vos scripts !!!

Quelques astuces

Créer des médias adaptatifs

Quelques astuces

Identifiez vos points faibles.
Des outils comme YSlow ou PageSpeed
apportent des informations pertinentes

Un service gratuit en ligne d'analyse:

www.GTMetrix.com

Les coûts

Très relatifs.

Vous n'avez pas besoin de déployez constamment.

Avec le cloud vous pouvez activez vos serveurs
Aux heures ou jours de pointe

Les coûts

Le déploiement peut être

Progressif

OU

Regressif

Installation

VARNISH

apt-get install varnish

Configuration

VARNISH

Les options de démarrage du serveur sont dans le fichier:

/etc/varnish/default

Changez le port en fonction de votre configuration
Et changez le fichier default.vcl par le votre
Pour éviter qu'il soit écraser lors d'une prochaine mise à jour.

File: **/etc/varnish/user.vcl**

```
backend default {
    .host = "127.0.0.1";  # IP address of your backend (Apache, nginx, etc.)
    .port = "8080";       # Port your backend is listening on
    .probe = {
        .url = "/";
        .timeout = 34ms;
        .interval = 1s;
        .window = 10;
        .threshold = 8;
    }
}

sub vcl_recv
{
   # Do not cache example.com, the admin area,
   # logged-in users or POST requests
   if (req.http.host ~ "example.com" ||
        req.url ~ "^/admin" ||
        req.http.Cookie ~ "logged_in" ||
        req.request == "POST")
    {
        return (pass);
    }

    # Don't allow cookies to affect cachability
    unset req.http.Cookie;

    # Set Grace Time to one hour
    set req.grace = 1h;
}

sub vcl_fetch
{
    # Set the TTL for cache object to five minutes
    set beresp.ttl = 5m;

    # Set Grace Time to one hour
    set beresp.grace = 1h;
}
```

Source:
linode.com

Configuration
VARNISH

Enfin redémarrez le service:

service varnish restart

Installation

NGINX

apt-get install nginx

Configuration

NGINX

Le fichier de configuration se trouve sous:

/etc/nginx/nginx.conf

Configuration

NGINX

File excerpt: **/etc/nginx/nginx.conf**

```
1  user www-data;
2  worker_processes 4;
3  pid /run/nginx.pid;
4
5  events {
6          worker_connections 768;
7          # multi_accept on;
8  }
```

Source:
linode.com

Configuration

NGINX

```
proxy_cache_path /path/to/cache levels=1:2 keys_zone=my_cache:10m max_size=10g inactive=60m
                 use_temp_path=off;

server {
...
    location / {
        proxy_cache my_cache;
        proxy_pass http://my_upstream;
    }
}
```

Source:
NGINX.com

Configuration

NGINX

Redémarrer les services:

service nginx restart

Services

CDN

Amazon CloudFront

OVH CDN

Merci !!

Quelques liens:

Site officiel NGINX
Site officiel Varnish
Configuration NGINX linode.com
Configuration NGINX en web cache
Configuration Varnish Linode.com
Varnish Docker Image (Non officiel)
NGINX docker image Officiel


That's all Folks!