

Temps de Chargement et SEO : Ce qu'il faut savoir



Par Aymeric Bouillat

Domaine :	Recherche	Référencement
Niveau :	Pour tous	Avancé

Le temps de chargement des pages impacte le trafic sur votre site et son taux de conversion. D'une façon directe ou indirecte, l'impact sur le crawl de vos informations par les robots de Google joue également un rôle non négligeable dans la qualité des données que vous fournissez à vos visiteurs. Dans cette série d'articles, nous allons tenter de vous donner les clés pour améliorer ce critère de plus en plus indispensable, notamment dans le cadre de la consultation de vos sites sur des mobiles. Exploration...

Les chiffres l'ont déjà démontré par le passé, à travers différentes annonces ou études : le temps de chargement des pages Web a un impact direct sur le trafic d'un site, et sur son taux de conversion quand celui-ci a des objectifs de vente. Les chiffres d'Amazon qui ont beaucoup circulé à ce sujet illustrent bien ce « fléau » : 100 ms de temps de chargement supplémentaires feraient perdre au géant Amazon 1% de ses ventes, et les études qui véhiculent la même idée depuis cette annonce ne sont pas difficiles à trouver sur le Web.

Performance : le combat des grands acteurs

Facebook a d'ailleurs signé des accords cette année avec plusieurs éditeurs (exemple : NY Times, National Geographic ou encore NBC News) qui vont en ce sens, pour accélérer l'accès au contenu. Afin d'éviter les allers-retours de ses utilisateurs entre son réseau social et des sites externes pouvant mettre plusieurs secondes à se charger, l'offre « Instant articles »

permet à certains éditeurs d'héberger leurs contenus sur Facebook, tout en étant rémunérés grâce à la possibilité d'inclure leurs publicités dans les articles qu'ils auront publié via ce service.

Certes, il y a derrière cet accord une volonté de conserver les internautes sur leur plateforme, comme le fait déjà Google avec le Knowledge Graph, mais aussi un réel besoin d'améliorer l'expérience utilisateur pour favoriser la consommation de contenu qui pourrait être bridée par des temps de chargements externes trop longs. C'est d'ailleurs le principal argument de Facebook au sujet de la mise en place de ce service, pour pallier les temps d'attente interminables qui peuvent être provoqués en situation de mobilité, en autre.

Mais avant Facebook, Google avait déjà fait du temps de chargement une priorité (Page speed insights, module pagespeed pour les serveurs Apache, Vitesse du site sur Analytics, etc.) pour une meilleure expérience utilisateur,

notion donc intimement liée au métier de référenceur puisque l'UX est l'un des maîtres-mots de Google. Les performances d'un site sont donc importantes pour les grands acteurs, rappelons également que cela fait partie des critères de classement de Google, bien que les informations communiquées par Google soient assez floues sur le sujet.

Un critère de classement direct ?

Un brevet Google, nommé « *Using resource load times in ranking search results* », déposé en 2010, décrit la façon dont des ressources peuvent être classées en cas d'égalité, grâce au temps de chargement. Si Google évalue de façon identique la pertinence de deux résultats différents, il pourra alors les départager pour proposer en priorité la page qui sera la plus rapide, afin d'offrir une meilleure expérience utilisateur.

(source :

<http://www.seobythesea.com/2014/02/googles-patent-site-speed-ranking-signal/>).

On peut lire dans l'introduction de ce brevet : « *This specification describes technologies relating to using resource load times in ranking search results* ».

Nous avons donc un signal qui peut faire prévaloir un résultat sur un autre en cas de similitude en terme de ranking, mais combien de fois cela se produit-il dans les pages de résultats ? Il ne s'agit pas d'un facteur de ranking mais plutôt d'une fonction permettant de faire pencher la balance en cas de similitudes entre deux pages par rapport aux autres critères de classement.

La vitesse de chargement des pages d'un site (taille du fichier couplé au temps de réponse) est donc un critère pouvant être utilisé par Google et son moteur de recherche, mais si l'on met de côté son utilisation directe dans l'algorithme, il a également un impact sur la visibilité des pages de votre site, de façon indirecte.

Un impact indirect sur vos positions

Le temps de chargement d'une page peut jouer un rôle dans l'algorithme de Google, mais il peut également avoir une incidence indirecte sur ses positions SEO : nous allons voir les différents impacts que peuvent avoir de mauvaises performances sur la capacité d'un site à se positionner.

Vitesse du crawl

Cela a été constaté sur plusieurs sites clients : une baisse significative du temps de téléchargement des pages explorées par Googlebot provoque une hausse des pages explorées. Une image comme celle de la figure 1 (page suivante) sera toujours plus parlante.

A l'inverse, une augmentation des temps de téléchargement des pages par Googlebot provoquera une baisse du nombre de pages explorées par jour (figure 2).

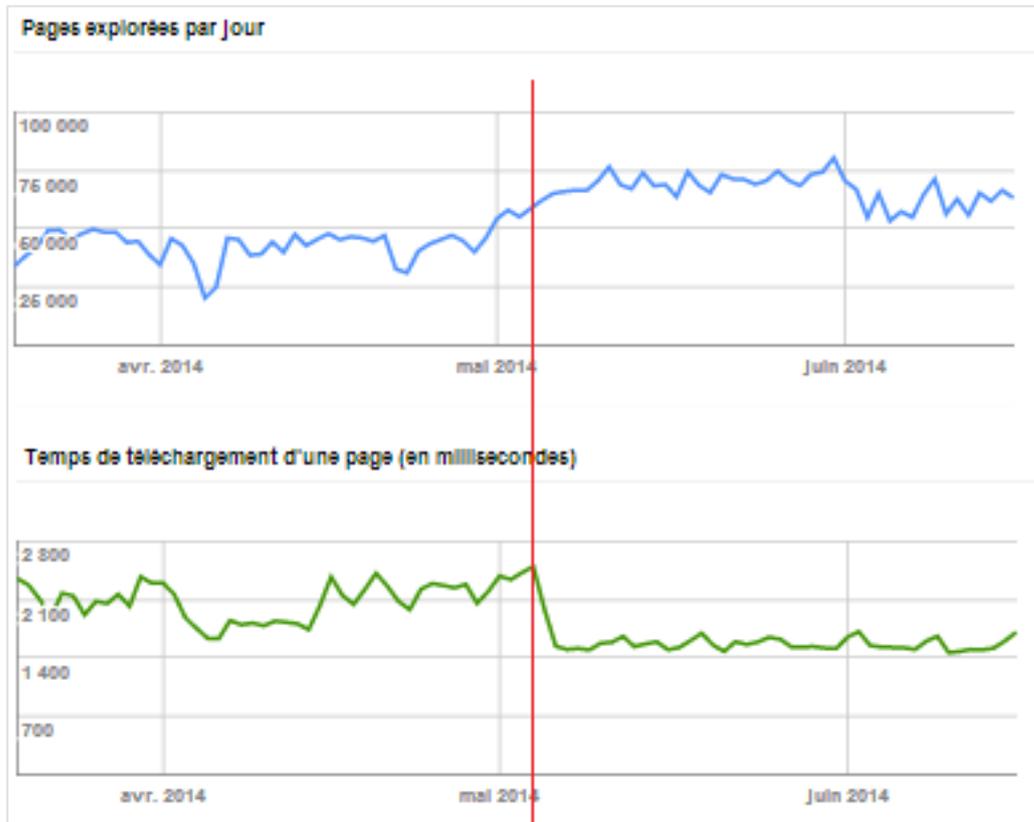


Fig.1. Impact de la baisse du temps de chargement des pages sur le crawl de Google.

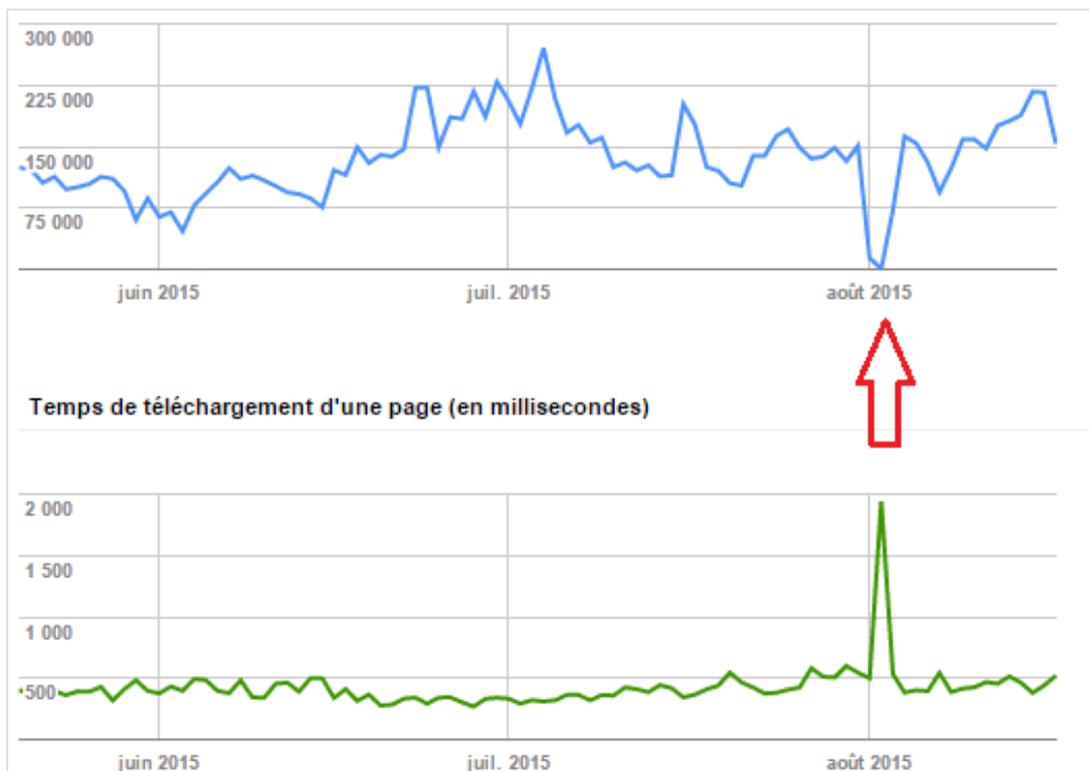


Fig.2. Impact de la hausse du temps de chargement des pages sur le crawl de Google.

Le crawler de Google alloue un temps CPU journalier à chaque site web qu'il visite, temps défini en fonction de différents critères (taille, autorité par exemple). Plus les temps de chargement seront longs (temps de réponse, poids et bande passante), plus le nombre d'éléments crawlés par jour sera faible, ce qui aura un impact sur votre visibilité dans les pages de résultats. Ces variations sont d'autant plus importantes en fonction du volume de pages moyen crawlé par jour, et se produisent par palier en fonction de ces données, d'après nos constatations.

Indexation des nouveaux contenus

Dans le cas d'un site d'actualités, vos contenus risquent de se retrouver indexés plus tard que ceux de vos concurrents, si les faibles performances de votre site contrarient le crawl de Googlebot. C'est le trafic lié à ces pages - qui se positionneront en retard ou moins bien que d'autres sites - dont vous ne bénéficierez pas.

Maillage interne et calcul du PR

Pour un site e-commerce avec un gros volume de pages, la mise en place d'une stratégie de maillage interne poussée peut mettre du temps à impacter les positions des pages travaillées. Chaque page doit être re-crawlée, analysée, évaluée pour permettre le calcul du PageRank des URL en fonction de la nouvelle répartition de vos liens, destinée à faire remonter les pages les plus pertinentes.

Les résultats de votre nouvelle répartition des liens au sein de vos pages pouvant mettre plusieurs semaines à avoir un effet visible dans

les pages de résultats, mieux vaut un taux de crawl élevé pour gagner du temps sur vos concurrents : pendant que vos URL ne sont pas visibles dans les premières pages de résultats, celles de vos concurrents le sont.

Désindexation du Duplicate Content

Il arrive que des sites mal configurés ou ayant été créés sans prise en compte du SEO se retrouvent avec des centaines de milliers d'URL inutiles, avec des pages vides ou non pertinentes pourtant indexées. Google peut d'ailleurs envoyer des messages d'alertes dans ces cas de figure : « *Le nombre d'URL détectées par Googlebot sur votre site est extrêmement élevé. Googlebot risque d'explorer inutilement un grand nombre d'URL distinctes* ». L'objectif de ce message est de vous inciter à faire le ménage pour ne garder que l'essentiel et également d'éliminer tout spider-trap (boucles sur une URL) qui aurait contribué à un crawl excessif non justifié. Le crawl de Google est couteux, mieux vaut optimiser la façon dont il gère le temps consacré à vos sites.

Une désindexation d'un volume important d'URL peut parfois être longue (plusieurs mois), et le sera d'autant plus si vos performances sont médiocres comme nous l'avons constaté précédemment. Ce type de nettoyage peut s'avérer très positif pour le positionnement de sites, la désindexation doit donc être restreinte à une courte période. Le délai nécessaire à la disparition de ces URL de l'index doit vous faire bénéficier d'une visibilité optimale le plus rapidement possible.

Taux de rebond dans les SERP

Enfin, si une page ne se charge pas assez vite à partir d'une page de résultats, l'internaute aura tendance à revenir en arrière pour afficher d'autres résultats, potentiellement plus rapides. Un phénomène qu'on appelle également *pogosticking* et qui joue certainement un rôle dans le classement des pages de résultats. Une page qui ne s'affiche pas au bout de quelques secondes pourrait donc, à terme, impacter vos positions d'une manière ou d'une autre.

Méthode et outils

Pour passer de la théorie à la pratique, de nombreux paramètres techniques vous permettront d'améliorer les temps de chargement de vos contenus de façon globale. Ces optimisations peuvent se trouver côté back-end (optimisations sur le serveur pour que les contenus soient délivrés rapidement), mais aussi côté front-end pour que le navigateur gère au mieux les différentes requêtes envoyées par le site Web et les éléments qui composent chaque page.

Les outils permettant d'optimiser la vitesse d'affichage d'un site fleurissent ces dernières années et ne notent pas les pages de la même manière : Une page ayant un score de 66/100 sur Google Page Speed Insights (<https://developers.google.com/speed/pagespeed/insights/>) pourra avoir un score de 79/100 sur PingDom Tools

(<http://tools.pingdom.com/fpt/>) par exemple.

Il est difficile de prioriser les tâches d'optimisation, ces outils attribuant une pondération différente en fonction des problématiques détectées. Certains critères présentés comme « A corriger éventuellement » relèvent plus du souci du détail, que d'un réel gain de performance.

Côté front-end

Parmi les mauvaises performances d'un site pouvant être les plus pénalisantes pour l'internaute (ou le mobinaute, encore plus impatient), voici une liste non exhaustive :

- Nombre de requêtes HTTP vers le serveur ;
- Pas de mise en cache des éléments de la page ;
- Poids des fichiers images (quantité, taille) ;
- Chargement des fichiers CSS et JS de façon asynchrone.

Chaque navigateur est capable de gérer un nombre limité de requêtes HTTP simultanées effectuées vers un domaine. Sur Chrome par exemple, la limite est à 6 (ce nombre a peut-être évolué pour les dernières versions de Chrome). Supposons qu'une page soit composée de 60 images, un navigateur téléchargera par bloc de 6 images simultanées les éléments, ce qui retardera d'autant l'affichage de l'intégralité de la page.

15. www.yapasdequoi...oji-release.min.js			114 ms
16. www.yapasdequoi...0x1024-150x150.jpg			94 ms
17. www.yapasdequoi...oading-150x150.png			73 ms
18. www.yapasdequoi.com - h1-150x150.gif			86 ms
19. www.yapasdequoi...07-PM-150x150.png			211 ms
20. www.yapasdequoi...glebot-150x150.jpg			185 ms
21. www.yapasdequoi...soldes-150x150.jpg			192 ms
22. www.yapasdequoi...rings1-150x150.png			133 ms

Fig.3. Source : <https://developers.google.com/web/fundamentals/performance/critical-rendering-path/render-blocking-css>

Par ailleurs, à l'heure où les sites *Responsive Web Design* sont de plus en plus fréquents sur le Web, assez peu le sont en réalité de façon complète. En effet, une grande partie d'entre eux se contentent d'afficher des images redimensionnées à l'affichage (``). Les devices mobile peinent avec des connexions 3G ou 4G, parfois timides, pour afficher des images de 1600px pouvant peser plusieurs centaines de Kilo-octets, qui finiront affichées sous forme de vignettes... Il faut garder à l'esprit que les images représentent en moyenne 65% des données contenues dans un page Web.

Quant aux fichiers JS, entre autres, ils bloquent le rendu de la page. Tant qu'un fichier de ce type n'est pas téléchargé complètement, aucune action n'est entreprise par le navigateur (voir figure 3).

Voici les optimisations recommandées afin de limiter le nombre de requêtes HTTP et de réduire le temps de chargement de vos pages (liste non exhaustive) :

- **Asynchrone** : Utiliser les attributs `async` ou `defer` pour les balises `<script>`, afin de ne pas bloquer le rendu de la page.

Source :

<https://developer.mozilla.org/fr/docs/Web/HTML/Element/script>

- **Combiner les fichiers CSS/JS** : Regrouper également les différents fichiers CSS, ainsi que les fichiers JS pour limiter les requêtes (et temps de réponse associés) vers le serveur.

- **Sprites CSS** : Regrouper les images d'habillage du site dans une seule et même image via un sprite CSS. Cette technique réduira considérablement le nombre de requêtes et l'affichage des pages Web se retrouvera donc accéléré. Il sera nécessaire de modifier votre code HTML et vos feuilles de style pour afficher des parties spécifiques de votre sprite.

Un bon outil pour la mise en place de sprites (voir figure 4) : <http://spriteme.org/>

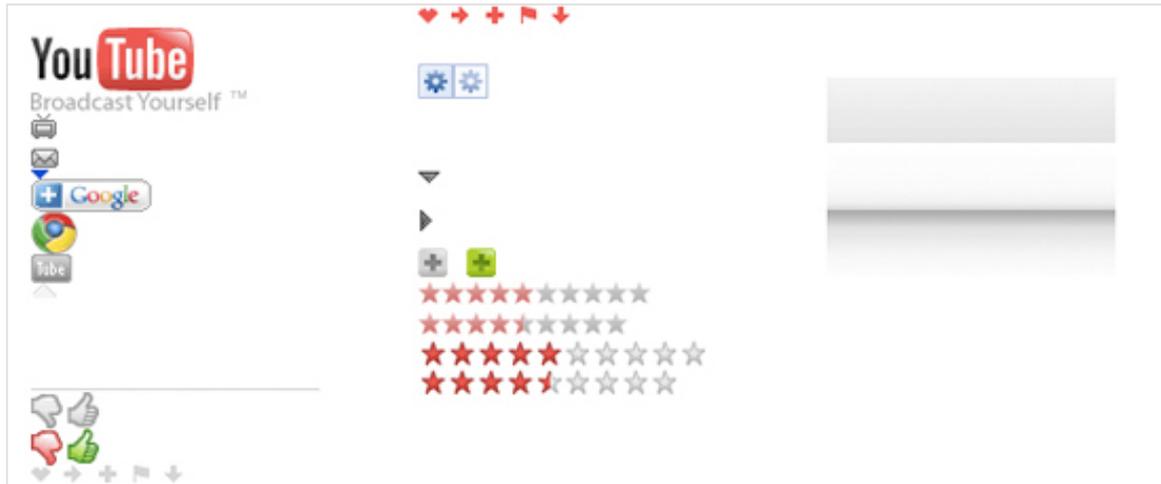


Fig.4. Exemple de Sprite CSS.

• **Poids des images** : Réduire le poids des images qui correspond souvent à la plus grande partie de la bande passante utilisée. Le nouveau format d'image WebP (extension de fichier .webp) mis en place par Google permet de réduire le poids de fichiers images d'environ 25% sur des fichiers de type PNG ou JPG, mais n'est malheureusement pas encore pris en charge par tous les navigateurs. Ce format n'est pas compatible avec Firefox, IE et Safari ; des discussions et tests sont en cours sur Firefox suite à une forte demande des développeurs et intégrateurs.

Plus d'infos :

<https://developers.google.com/speed/webp/>

• **Images et RWD** : Pour les sites Responsive Web Design et les images,

des solutions existent : <http://adaptive-images.com/> vous permettra de délivrer des images redimensionnées en fonction de la taille de l'écran des internautes, via PHP et de la réécriture d'URL. Il existe une autre solution grâce aux évolutions du HTML5 avec de nouveaux attributs (en cours d'implémentation sur différents navigateurs), qui permettront de spécifier différentes versions d'images en fonction de la dimension des écrans : <https://responsiveimages.org/> (voir figure 5).

• **Utiliser le lazyloading** : sur des pages de listing produits, toutes les images ne seront pas nécessairement vues par le visiteur. Le Lazyloading est une méthode qui consiste à ne charger que ce qui est visible dans la fenêtre de l'internaute, les visuels produits seront



Fig.5. Element <picture> et attributs qui permettent l'intégration de différentes tailles d'images en fonction du viewport.

donc chargées au fur et à mesure que l'internaute scrolle. C'est une pratique qui est recommandée sur les sites mobiles, puisque les images sont les fichiers les plus gourmands en termes de bande passante. Son implémentation est conseillée surtout sur des pages étant composés de listes d'images, son intérêt sera limité sur une fiche produit par exemple (image souvent visible au-dessus de la ligne de flottaison). Un Javascript compatible SEO vous facilitera la mise en place du Lazyloading : <http://qnimate.com/lazy-loading-images-and-its-seo-impact/>. Attention cependant, le fichier .JS permettant ce lazyloading pèse quand même 122Ko.

L'utilisation du module Pagespeed (<https://developers.google.com/speed/pagespeed/module/>) qui est capable de combiner JS/CSS, minifier ces derniers, optimiser les images tout en générant des sprites CSS à la volée et autres fonctions d'optimisations pourrait sembler être la solution magique. Il n'en est rien : `mod_pagespeed` pour les serveurs Apache ou Nginx peut s'avérer catastrophique en terme de ressources CPU sur votre serveur s'il est mal configuré, et faire office de pansement sur une jambe de bois : il vaut mieux optimiser directement vos pages et les fichiers qui les composent à la source.

Malgré ces optimisations, certains sites ont encore de grosses lacunes avec la gestion du cache navigateur : il n'est pas rare, lors de la navigation sur certains sites, de voir les feuilles de styles ou images d'habillages retéléchargées entre chaque page. Cela ralentit l'affichage et sollicite le serveur inutilement : en cas de pic de charge, les ressources serveurs seront saturées inutilement.

A titre d'exemple, il suffit de quelques lignes dans la configuration d'Apache, couplées au module `mod_expires`, pour que les CSS et JS soient mis en cache pendant 1 semaine par le navigateur, ce qui rendra plus fluide la navigation des internautes :

```
ExpiresActive On
ExpiresByType text/javascript "access plus 1 week"
ExpiresByType text/css "access plus 1 week"
```

La gestion des en-têtes de Cache fera d'ailleurs l'objet d'un prochain article de la lettre Recherche et Référencement.

Côté back-end

Les faibles performances d'un site peuvent également être liées au back-end et à la façon dont sont générées les pages HTML à destination des internautes par le serveur. Des sites dynamiques, utilisant des bases de données et des langages de programmation, peuvent mettre un certain temps à délivrer le contenu HTML généré à l'utilisateur final : requêtes SQL, includes PHP de blocs, multiples plugins Drupal par exemple, sont autant d'éléments qui risquent de retarder la livraison de la page générée aux internautes.

Ce travail ne doit pas être répété pour chaque internaute : la page HTML générée lors de son premier appel doit pouvoir être mise en cache sur le serveur, pour être délivrée rapidement aux internautes suivants, sans sollicitation de la base de données et de l'applicatif (traitement PHP par ex.). Des plugins existent pour les CMS les plus répandus (Drupal Boost pour Drupal, WP Rocket pour Wordpress),

mais le cache serveur reste souvent le plus performant.

Le cache des pages dynamiques est d'autant plus indispensable que certaines pages de facettes/filtres crawlées par Googlebot peuvent augmenter de façon considérable les temps de chargement présents dans les « Statistiques d'exploration ». Un délai de mise en cache plus long pourra être configuré pour ces pages, ou encore une génération de fichiers statiques la nuit dans certains cas pourra alléger la charge serveur provoquée par le crawl de Google.

Le *Reverse Proxy Varnish*, qui est un intermédiaire entre le serveur et l'internaute, gère justement ce type de mise en cache, et peut vraiment réduire de façon considérable le temps de chargement de vos pages dans Google Search Console. Sa mise en place peut s'avérer complexe, mais vous sera d'un grand bénéfice pour le crawl de Google, ainsi qu'en cas de forte montée en charge de vos sites pendant les périodes de soldes entre autres. Il est d'ailleurs utilisé par de gros acteurs comme The New York Times, The Guardian ou encore Tumblr. Varnish pourra également faire l'objet d'un futur article.

Accélérer un site Web, c'est incontournable !

L'optimisation des performances de vos pages Web est incontournable de nos jours : bien que des connexions rapides avec des fibres optiques se développent, le nombre de mobinautes disposant de connexions Wifi ou 3G/4G demande des optimisations, pour répondre aux mobinautes dans les plus brefs délais, mais aussi pour servir les crawlers au mieux afin d'améliorer le SEO de son site de façon directe et indirecte comme nous avons pu le voir. Ces optimisations sont parfois une histoire de compromis, et ne sont pas toutes réalisables, ne serait-ce qu'à cause des scripts externes dont vous ne maîtrisez par forcément les temps de réponses http et leurs en-têtes de cache. Elle peut s'avérer rapidement complexe sur de gros sites, les différents critères de performances n'ayant pas tous le même impact. La meilleure solution reste de vous faire accompagner par un professionnel, pour éviter de courir le risque de dégrader les performances de votre site avec des outils mal configurés.



Aymeric Bouillat, *consultant*
SEO, Resoneo

(<http://twitter.com/aymerictwit> ou
<http://www.yapasdequoi.com>)