Les dernières évolution de Google en matière de crawl

Retour au sommaire de la lettre

Domaine :	Recherche	**Référencement**
Niveau :	Pour tous	**Avancé**

Pour créer et maintenir leur index à jour, les moteurs de recherche utilisent des logiciels appelés indifféremment "spiders", "crawlers", "bots" ou "agents". Ces outils, chargés de "crawler" le Web à la rencontre de nouvelles pages ou de nouvelles versions de pages déjà indexées, ont grandement évolué depuis une dizaine d'années. Cet article, très complet, explore ces différentes versions et évolutions de Googlebot, le spider de Google, qui n'a plus rien à voir aujourd'hui avec ce qu'il était lors de la naissance du moteur de recherche leader. Prêt à tenter l'aventure et à endosser votre habit de spider ?...

Avant de pouvoir servir des pages de résultats aux utilisateurs, les moteurs de recherche ont besoin de créer des index sophistiqués, après avoir "aspiré" le contenu du World Wide Web. Cette dernière tâche est réservée aux "crawlers" (On appelle également souvent les crawlers "spiders" ou "bots"), dont le plus connu est bien sûr Googlebot. L'importance de cette phase de *crawl* est souvent sous estimée par les webmasters. Or, le fait qu'une page ne soit pas crawlée au bon moment, ou pas crawlée du tout, a logiquement un impact très négatif sur le référencement et le positionnement de ces pages dans un moteur comme Google.

Nous allons voir tout d'abord que la tâche des crawlers est loin d'être facile, en raison des limitations techniques actuelles, mais aussi parce que les webmasters s'ingénient à rendre les choses difficiles pour ces programmes. Nous verrons ensuite comment Google en particulier a amélioré son système de crawl au fil du temps pour résoudre certaines de ces difficultés. Enfin, nous nous intéresserons aux évolutions récentes du crawl de Google, aux défis nouveaux qui se dressent devant les ingénieurs des outils de recherche, et nous donnerons en conclusion quelques recommandations opérationnelles pour surveiller le crawl de Google.

Comment reconnaître Googlebot dans mes logs ?

Un robot d'indexation (crawler) "lit" les pages d'un site web exactement comme le fait un utilisateur normal, c'est-à-dire en téléchargeant la page pour la stocker et la lire localement à l'aide d'un logiciel client http, autrement nommé "agent utilisateur" (*User Agent*). Pour un utilisateur lambda, le client http est un navigateur (Firefox, Safari, Opera, IE...). Chaque client utilisateur est caractérisé dans les logs par une chaîne de caractères constituant la "signature" du User Agent (à condition que le serveur web ait été paramétré pour stocker le champ "user agent" dans les logs). C'est la même chose pour Googlebot et les autres "bots" des outils de recherche.

Voici les principales signatures utilisée par Googlebot

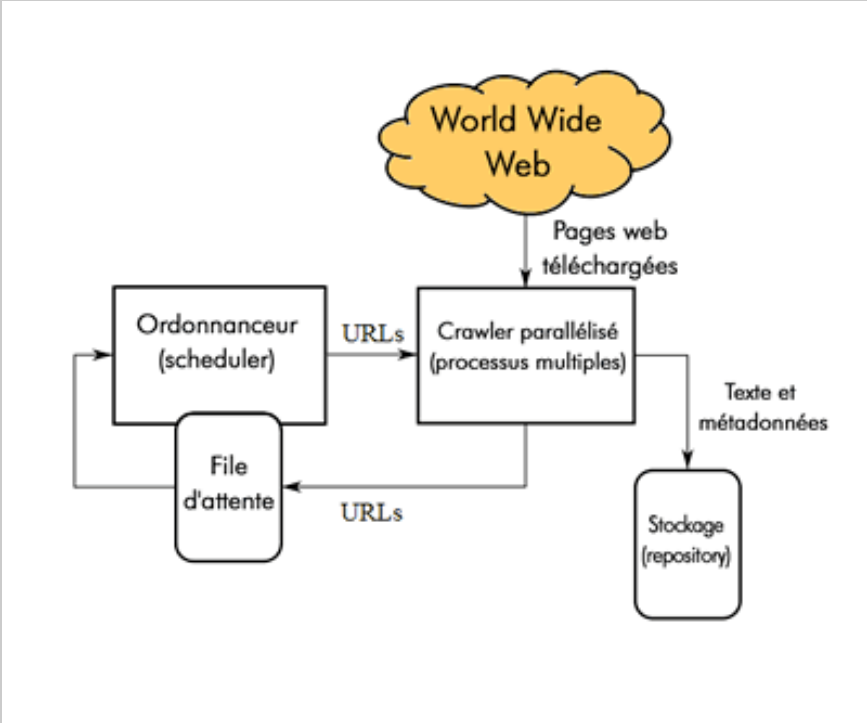
* Googlebot/2.1 (+http://www.google.com/bot.html) : crawl avec parser optimisé
* Mozilla/5.0 (compatible; Googlebot/2.1; +http://www.google.com/bot.html) : crawl "Mozilla like", avec un parser complet
* Googlebot-Image/1.0 : crawler de Google Images
* DoCoMo/2.0 N905i(c100;TB;W24H16) (compatible; Googlebot-Mobile/2.1; +http://www.google.com/bot.html) : crawler de Google Mobile, il y'a de très nombreuses variantes de la signature, car Google teste les sites "cloakés" pour renvoyer un contenu adapté aux différents modèles de mobile
* Mediapartners-Google/2.1 : crawler du système adsense

ATTENTION : Google utilise un système de repository centralisé, une page

crawlée par Mediapartners peut donc être utilisée par le crawler "normal" sans que la page soit à nouveau téléchargée par Googlebot/2.1

Crawler le web n'est pas une tache facile

Le schéma ci-dessous rappelle le principe d'un crawler. Le "bot" est chargé d'aller télécharger les pages dont les urls lui sont transmises par un programme clé : l'ordonnanceur. C'est l'ordonnanceur (*scheduler* en anglais) qui pilote la file d'attente des urls à crawler, et décide quelles urls doivent être téléchargées et dans quel ordre. Les pages téléchargées sont ensuite parsées et stockées dans une base appelée "*repository*", après qu'un programme ait éventuellement ajouté des métadonnées permettant de qualifier les pages web.



La taille du WWW pose problème

Le web contient des myriades de pages, sa taille ne cesse de grandir, et crawler une telle quantité de pages nécessite beaucoup de machines et une bande passante colossale. Heureusement, pour le moment, la puissance des machines et de la bande passante accessible permet de crawler plus de pages et plus vite qu'autrefois. Mais le problème du volume de données à télécharger reste la source de nombreuses limitations techniques, voire financières car l'infrastructure nécessaire coûte très cher...

Google a déclaré en juillet 2008 qu'ils avaient atteint le cap sympolique des 1000 milliards d'urls uniques connues de leur système de crawl. (http://googleblog.blogspot.com/2008/07/we-knew-web-was-big.html) : *We've known it for a long time: the web is big. The first Google index in 1998 already had 26 million pages, and by 2000 the Google index reached the one billion mark. Over the last eight years, we've seen a lot of big numbers about how much content is really out there. Recently, even our search engineers stopped in awe about just how big the web is these days -- when our systems that process links on the web to find new content hit a milestone: 1 trillion (as in 1,000,000,000,000) unique URLs on the web at once!*

Cela ne veut pas dire que l'index de Google atteint cette taille, loin de là. Il faut déjà se dire que 12 à 13% des pages accessibles par les moteurs sont réellement des pages contenant des

informations exploitables dans un outil de recherche (selon plusieurs estimations convergentes). Quant à la taille réelle de l'index de Google, elle n'était que de quelques milliards de pages il y a peu (8 milliards de pages annoncés le 10 novembre 2004), aujourd'hui on peut l'estimer à quelques dizaines de milliards de pages. Pour autoriser une telle taille d'index, Google a dû revoir entièrement son architecture d'index et de serveurs. Cette tâche essentielle a été l'oeuvre d'Anna Patterson, qui a baptisé cette architecture "Teragoogle". Les Datacenters "Bigdaddy" apparus fin 2005 début 2006 sont en particulier la face visible de Teragoogle. Anna Patterson a quitté depuis Google pour fonder le moteur Cuil (http://www.cuil.com/).

Cela signifie en clair que Google ne crawle qu'un faible pourcentage des urls qu'il trouve !

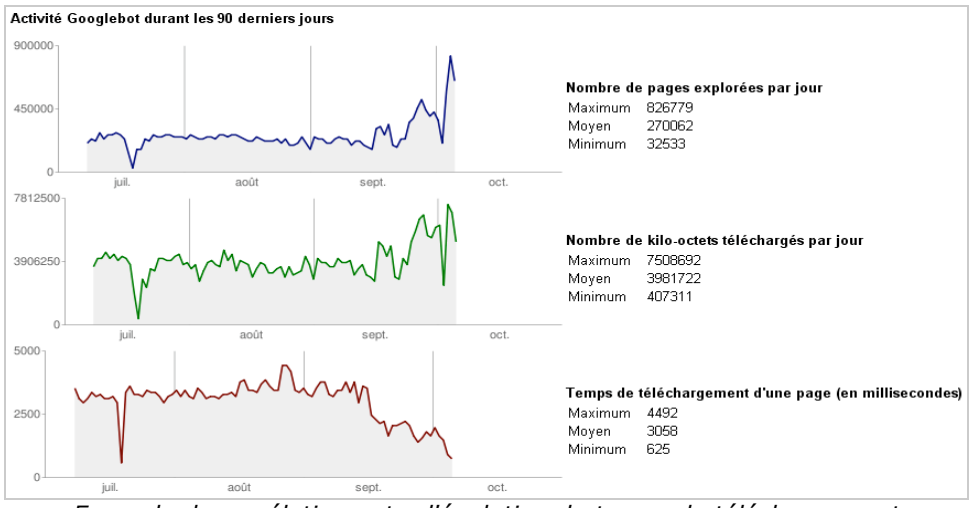
Un crawler efficace est un crawler patient et poli...

Un crawler mal développé peut très bien faire "tomber" un serveur, même puissant. Ce problème a conduit notamment à la création du "*robots.txt exclusion protocol*" pour permettre aux webmasters d'interdire l'accès à certains spiders. Mais il a aussi été obligatoire de développer des ordonnanceurs capables de crawler de nombreux sites web à la fois pour répartir les demandes de pages sur de nombreux sites et d'éviter ainsi la surcharge d'un seul d'entre eux.

Mais les crawlers agissent quand même comme des visiteurs "pressés"

Les crawlers doivent récupérer le maximum de pages utiles sur une période de temps donnée. Il est donc hors de question de laisser des processus attendre longtemps qu'un serveur web réponde ou qu'une page web se télécharge. Quand de tels problèmes de performance surviennent, les crawlers sont programmés pour abandonner rapidement et vont aller visiter d'autres pages qui, elles, répondent beaucoup plus vite (on peut crawler 5 pages qui se téléchargent en 400ms, pendant le temps passé à attendre une page qui se télécharge en plus de 2 sec). Si le problème perdure, les crawlers éviteront dorénavant de venir crawler ces pages, notamment si par ailleurs elles ne sont pas jugées "importantes".

Ce temps de téléchargement est communiqué par Google dans les statistiques des Webmaster Tools, il est donc intéressant de surveiller cet indicateur et sa corrélation avec une baisse du nombre de pages téléchargées



Exemple de corrélation entre l'évolution du temps de téléchargement et du nombre de pages explorées par jour.

Comment crawler dans la poussière (DUST : Duplicate Urls Same Text) ?

Les urls que trouve un crawler sont souvent les adresses multiples de pages en réalité identiques. Les ordonnanceurs des outils de recherche doivent donc essayer de "normaliser" les urls, c'est à dire de faire en sorte que les différentes syntaxes d'urls soient rattachées à une version unique constituant l'adresse de la page. La normalisation des urls est une tâche beaucoup plus difficile qu'il n'y parait, et cela signifie que les crawlers téléchargent à nouveau très souvent un grand nombre de pages pour rien, puisqu'elles figurent déjà en fait dans l'index, sous une autre adresse.

On appelle aussi ce processus "canonicalisation", en particulier quand il s'agit de résoudre les cas d'urls dupliquées de type :

- http://domaine.com = http://www.domaine.com
- http://www.domaine.com/rep1 = http://www.domaine.com/rep1/ = http://www.domaine.com/rep1/index.html

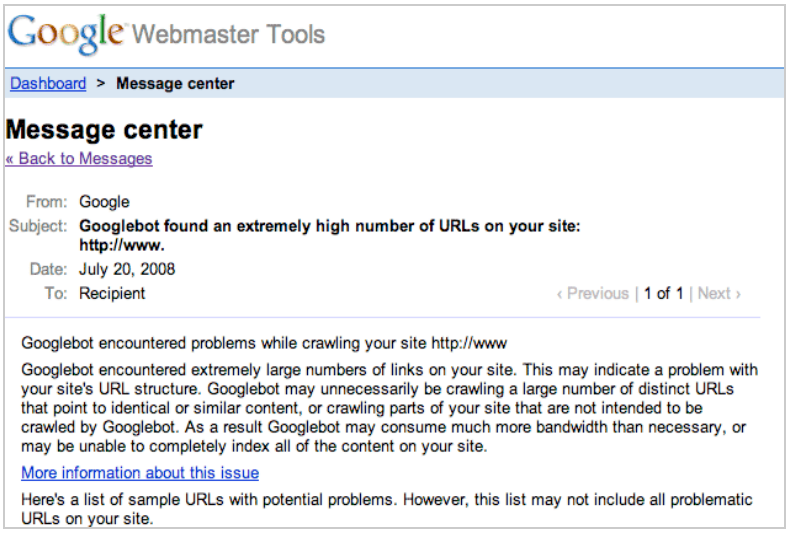
Il faut noter à ce sujet que la nouvelle balise <link rel="canonical" > qui résout en théorie le problème des DUST pour la phase d'indexation (http://actu.abondance.com/2009/02/duplicate-content-google-microsoft-et.html), ne résout pas le problème complètement pour les crawlers (il faut parser la page pour lire cette balise, et donc l'avoir téléchargée avant). Une grande partie de la capacité de crawl des outils de recherche est donc inutilement dépensée à crawler la "poussière" (DUST)

Quand les webmasters tendent des pièges au crawlers...

Dans la plupart des cas d'urls dupliquées, le nombre de syntaxes différentes est fini, et le nombre d'urls générées l'est également le plus souvent. Mais dans certaines situations, c'est un nombre infini de syntaxes d'urls différentes que le système génère. Un crawler peut donc ne jamais parvenir à épuiser la liste des urls qu'il découvre. Cette situation crée ce que l'on appelle un "spider trap", un piège à robots. Les developpeurs de crawler s'ingénient donc à créer des systèmes permettant de détecter l'existence de tels "spider traps", soit pour programmer les bots pour ne pas crawler ces pages, soit pour identifier la cause de duplication et "normaliser" correctement les urls.

Tous les spider traps ne sont pas causés par des cas particuliers de DUST. Certains pièges peuvent aussi être créés par des applicatifs web qui génèrent des urls et des pages de contenus différentes à chaque visite, ou des combinaisons extrêmement nombreuses générées à partir d'une même base de données.

Depuis juillet 2008, Google envoie un message aux webmasters *via* les Google Webmaster Tools en cas de découverte d'un spider trap.



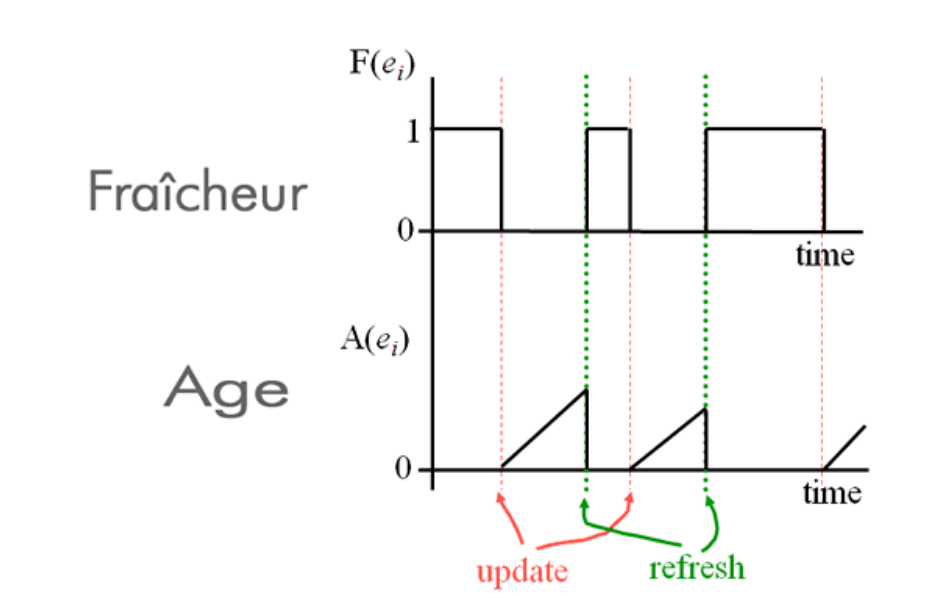
Un bon crawler sait préserver la fraicheur de l'index

Un crawler met un temps assez long (plusieurs jours, classiquement plusieurs semaines) pour télécharger l'intégralité des pages d'un site. Dans ces conditions, le crawler peut être confronté à deux types de problèmes qui vont l'empêcher de fournir des données fraîches pour l'index du moteur :
- il peut ne pas être informé de l'apparition d'une nouvelle page sur un site web.
- il peut ne pas savoir qu'une page a été modifiée depuis son dernier passage.

Ne pas confondre l'âge et la fraîcheur d'un index

L'âge d'un index n'a aucune importance pour la qualité d'un moteur. La fraîcheur de l'index est par contre cruciale.
Il arrive fréquemment que ces deux notions soient confondues, il est donc important de repréciser leur signification...
L'âge d'une page mesure le temps écoulé depuis la dernière modification des pages, ou depuis leur apparition sur le web. Pour un crawler, on mesure aussi un âge en considérant que si la page vient d'être crawlée, son âge est de 0, sinon, son âge est le temps écoulé depuis le dernier crawl.

La fraîcheur indique si la page dans l'index est à jour, c'est à dire si elle correspond à la version en ligne sur le site à un instant t. La fraîcheur de l'index correspond à la proportion de pages "à jour" dans l'index.



Une partie du web est inaccessible aux moteurs

Par ailleurs, un grand nombre d'autres obstacles empêchent les crawlers de découvrir toutes les pages qu'ils souhaiteraient indexer. La plupart de ces obstacles sont techniques. Dans certains cas, il peut s'agir de barrières créées involontairement par les webmasters, comme par exemple :
- des pages utiles cachées derrière des robots.txt
- des pages liées à l'aide de scripts javascript, ou actionscript (Flash) et qu'un moteur ne sait pas lire
- des pages cachées derrière des formulaires
- des pages accessibles uniquement sur authentification
- les liens finissent par "pourrir" (phénomène de "link rot"): au bout d'un certain temps, ils aboutissent sur des 404 ou, au mieux, des redirections... L'ampleur du phénomène est importante, et un moteur de recherche dont l'index présente des problèmes de mise à jour est

lui même victime du phénomène.

Les trois générations de bots chez Google

Première époque : le crawl fermé ou "par lot" (batch crawl)

Google a commencé à crawler le web sur le modèle du "*crawl par lot*" : on identifie la liste des urls à crawler, on confie ensuite à l'ordonnanceur le soin d'aller "aspirer" l'ensemble des pages. Le crawl a donc un début et une fin.

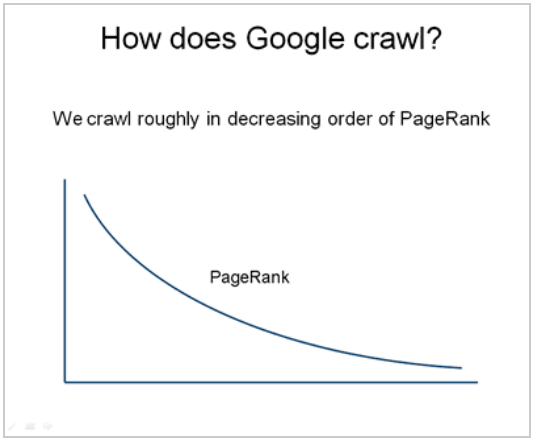
Ce comportement caractéristique était observé facilement par les webmasters dans leurs logs, qui voyaient Googlebot crawler frénétiquement leurs pages pendant quelques jours (période du "crawl profond", ou *deep crawl* en anglais), puis le voyaient disparaître pour revenir un mois plus tard. Un tel modèle de crawl posait évidemment beaucoup de problèmes du côté de la fraîcheur de l'index : Google avait donc recours à un deuxième crawler (baptisé "*fresh crawler*") chargé d'aller chercher quelques pages apparues récemment entre deux périodes de *deep crawl*. L'efficacité de l'ensemble laissait néanmoins grandement à désirer.

Deuxième époque (Eté 2003->Eté 2004) : le crawl continu "incrémental" et "ciblé"

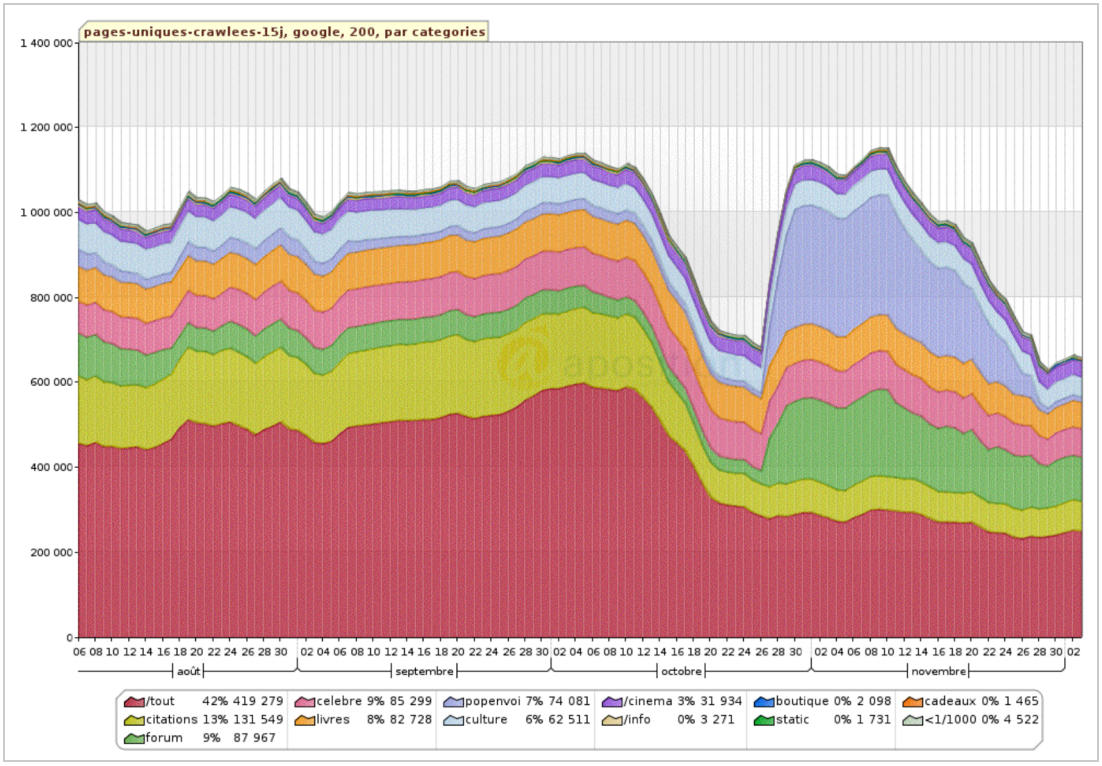
Au printemps 2003, les progrès faits sur la mise à jour du PageRank (mise à jour au fil de l'eau, "*rolling update*", voir article du mois dernier à ce sujet) permet de casser le rythme mensuel des Googledance (Deepcrawl, calcul du PR, GoogleDance = synchronisation des datacenters, re-deepcrawl), pour passer à une mise à jour continue de la base (baptisée "*everflux*").

Google passe alors à un crawl incrémental, et ciblé, à l'occasion de la mise à jour baptisée "Fritz", qui a eu lieu au cours de l'été 2003. Le crawl est incrémental, car il ne repart jamais de zéro, ne s'arrête jamais. Les urls découvertes sont communiquées à l'ordonnanceur. Ce dernier peut décider d'aller recrawler des pages qu'il connait déjà pour obtenir une version à jour, ou peut décider d'aller découvrir des pages nouvelles. Le crawl est ciblé ("*focused crawling*") parce que l'ordonnanceur ne crawle pas toutes les urls qui sont découvertes. Il délaisse volontairement les pages sans importance et tient compte d'autres éléments comme le rythme de changement des sites et des pages pour déterminer les pages à crawler.

Cette version du crawler est directement issue des travaux effectués très tôt par Larry Page avec l'équipe d'Hector Garcia Molina à Stanford. Les modèles évoqués dans les articles de Molina et Cho (*voir Bibliographie*) ont beaucoup de choses en commun avec le comportement de crawl de Google post 2003, et cela se vérifie encore aujourd'hui. Par exemple, Matt Cutts a récemment précisé que le PageRank des pages était l'un des critères majeurs que Google utilisait pour déterminer les pages à crawler en priorité. L'intérêt de ce critère était déjà expliqué dans l'article de 1998 de Cho et Molina co signé avec Larry Page.



Slide présenté par Matt Cutts au WordCamp 30 mai 2009



Un exemple du comportement de Googlebot sur un site observé à l'aide de l'appliance "ApoboxTM" d'Aposition et qui démontre le caractère "ciblé" des modifications du crawl de Google. Le "creux" observé fin octobre correspond à une baisse du crawl sur la catégorie "rouge", la hausse qui suit correspond à une hausse des pages crawlées dans les catégories verte et bleue.

Troisième époque : (Printemps 2003-> Eté 2004 ?) : arrivée d'un parser évolué

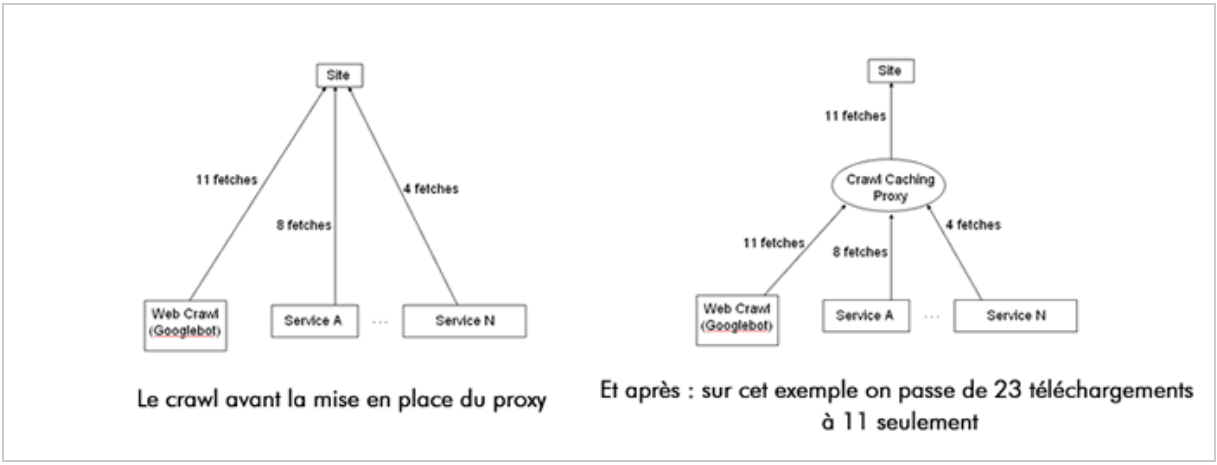
En juillet 2004, Google change la signature de Googlebot. Et à la fin de l'été 2004, après plusieurs mois de tests, un crawler particulier fait son apparition sous la signature *"Mozilla/5.0 (compatible; Googlebot/2.1; +http://www.google.com/bot.html)"*. Il s'agit d'une version du crawler destinée à récupérer une version complète d'une page, avec ses CSS et les javascripts. Ce crawler doublé d'un parser évolué montrera par la suite une capacité toujours plus grande, soit à lire le dynamic HTML, les scripts, le flash et même les CSS.

Commentaire de Googleguy (Matt Cutts) sur ce nouveau crawler : *The primary reason for this is that some web servers assume that unless a user agent is IE, Netscape/Mozilla, or maybe Opera, that your browser won't support JavaScript, frames, etc. As Googlebot gets better over time, it gets closer to a regular user and browser in our ability to handle features like that. This user-agent change would let vanilla webservers assume Googlebot is more like a regular user, while still allowing site owners to recognize that Google is visiting and still providing clear contact info in case of questions or issues. We would still respect "Googlebot" in robots.txt, so the overwhelming majority of webmasters wouldn't have to change anything at all; most site owners probably wouldn't even notice the difference. This bot that a few people noticed was a test crawl with the new user agent. It looks like things went very smoothly, but we'll still be conservative when making this change. If people have any positive or negative comments or questions, just post here. If we don't hear any showstopping objections, we'll look at making this name change on the user-agent when we're convinced everything would be a smooth transition.*

Quatrième époque : (-> fin 2005) : crawl optimisé

A partir de la fin de l'année 2005, le comportement des crawlers de Google a changé une nouvelle fois. Une partie de ces changements a pour origine la migration vers la nouvelle architecture de Google (Teragoogle *alias* BigDaddy). Google a mis en place un système de

"proxy centralisé" pour éviter à certains de ses crawlers spécialisés (notamment le crawler des news, ou le crawler "Mediapartner" pour le programme adsense) de recrawler une deuxième fois une page déjà crawlée par l'un de ses "collègues".



Novembre 2006 : l'avènement des sitemaps XML

Le 16 novembre 2006, Yahoo!, Google et Microsoft ont publié un communiqué de presse commun indiquant qu'ils géraient désormais le protocole "sitemaps". Les sitemaps sont des fichiers XML que les webmasters peuvent placer sur leur site pour indiquer aux moteurs la liste des pages de leurs sites, la fréquence de changement du site, ainsi qu'un ordre de priorité :

Exemple de contenu pour un sitemap :

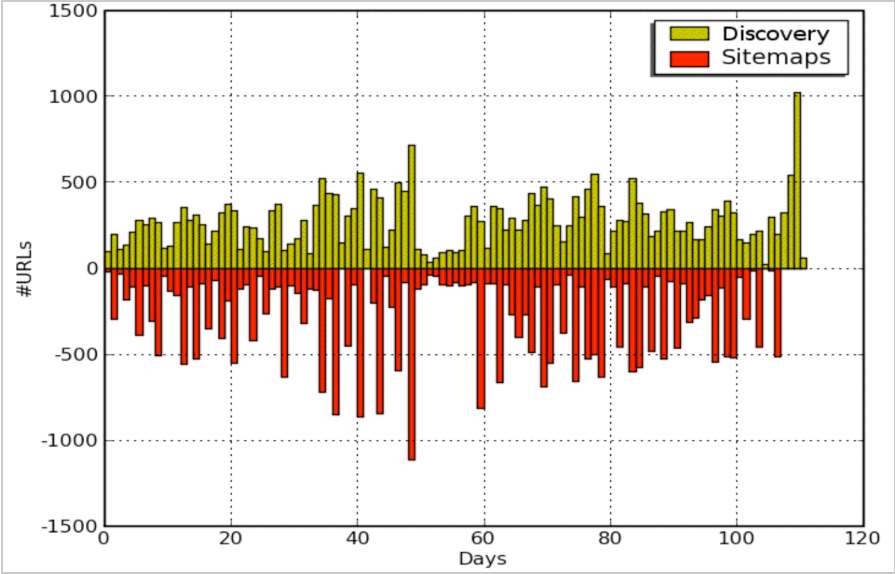
```
<url>
<loc>http://www.example.com/</loc>
<lastmod>2005-01-01</lastmod>
<changefreq>monthly</changefreq>
<priority>0.8</priority>
</url>
```

Beaucoup de webmasters croient que l'ajout d'un sitemap va permettre à l'intégralité des urls de leur site d'être crawlées : ce n'est pas toujours vrai dans la pratique. Beaucoup pensent également que cela change la manière dont google crawle leur site (fréquence de crawls en particulier) : cela se vérifie encore moins souvent. Voici par ailleurs ce que Google dit à propos de l'influence des sitemaps sur le crawl

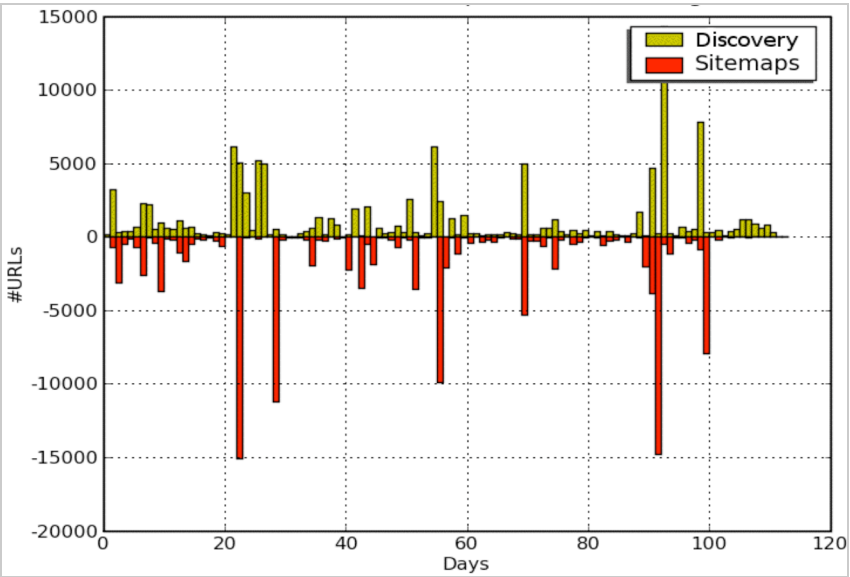
Nous ne garantissons pas que nous explorerons ou indexerons la totalité de vos URL. Par exemple, nous n'explorerons ou n'indexerons pas les URL d'image contenues dans votre sitemap. Les données figurant dans votre plan Sitemap nous permettent en revanche d'identifier la structure de votre site et de mieux planifier les explorations futures de votre site, lesquelles seront également plus performantes. Les plans Sitemap se révèlent généralement utiles pour les administrateurs Web et ne peuvent en aucun cas pénaliser ces derniers.

Pourquoi les sitemaps sont parfois efficaces, et parfois sans effets ?

En fait, les sitemaps n'ont jamais été conçus pour donner la main aux webmasters sur le crawl des moteurs. Il s'agit d'aider en fait les crawlers à identifier les cas où un crawl exploratoire classique (on suit les urls découvertes) passe à côté de pages utiles. Les graphique ci-dessous, montrés par Google dans une étude sur l'utilité des sitemaps, montrent deux cas de sites connus dans lesquels la proportion de pages "oubliées" par le crawl exploratoire est soit faible, soit important. Il est clair que dans le cas de PubMedCentral, l'exploitation des données du sitemap est indispensable pour obtenir un index frais et complet.



Comparaison des pages découvertes par le crawler et par le biais des sitemaps pour CNN : les différences de volume sont faibles, le problème de fraîcheur très relatif.



Pour Pubmedcentral.com, le graphe ci-dessus montre que le site connait des pics de mise en ligne révélé par le sitemap. Les pages sont découvertes par le crawler, mais avec un décalage certain, posant de réels problème de fraîcheur.

Les données du sitemap sont donc en fait ajoutées aux données à la disposition de l'ordonnanceur. Dans certains cas, ces données vont changer le comportement de l'ordonnanceur, qui va crawler des pages qu'il n'arrivait pas à découvrir, ou va changer la vitesse de rafraîchissement. Mais dans bien des cas, l'ordonnanceur ne va rien changer à son comportement, soit parce que le sitemap confirme les données dont il disposait, soit parce que d'autres critères sont privilégiés dans le choix des pages à crawler : par exemple les pages à faible PageRank du site peuvent demeurer désespérément non crawlées.

L'ajout d'un sitemap peut donc s'avérer utile, notamment quand un certain nombre de pages importantes sont ignorées par les bots. Mais son efficacité n'est pas garantie. Dans certains cas, le sitemap peut même avoir des effets de bord non désirés comme un transfert du crawl d'une zone à une autre.

Eté 2007, le "Minty Fresh Index"

Petit à petit, Google a visiblement réussi à améliorer progressivement ses stratégies de crawl et d'indexation, pour permettre l'affichage dans l'index de pages toujours plus "fraîches". En juillet 2007, plusieurs personnes ont relevé qu'il était à présent possible de voir parfois apparaître une page dans l'index quelques minutes après sa mise en ligne. Ce constat a été rendu possible par un changement dans l'affichage de la date de crawl dans les SERPs, et a été confirmé par Matt Cutts en août 2007 sur son blog. Matt Cutts a baptisé cette évolution le "*Minty fresh index*".

Exemple cité par serountable.com :

Avant :

Après :

De nouveaux changements du côté de Googlebot ? (printemps 2008)

Avec l'intégration de la technologie pour crawler le web profond (*voir plus loin*), certains webmasters ont aperçu dans leurs logs des comportements tout à fait nouveaux de Googlebot. Le crawler de Google parait depuis quelques temps :
- soit inventer des syntaxes d'urls.
- soit adopter des comportements de type "moissonneuse" (Les crawlers développés pour aspirer les pages du web invisible ont souvent ce comportement qualifié de "*harvester*" (moissonneuse) : face à une url de type /rep1/rep2/rep3, le crawler essaiera les syntaxes /rep1/, et /rep1/rep2).
- soit tester des paramètres aux hasard dans les urls.

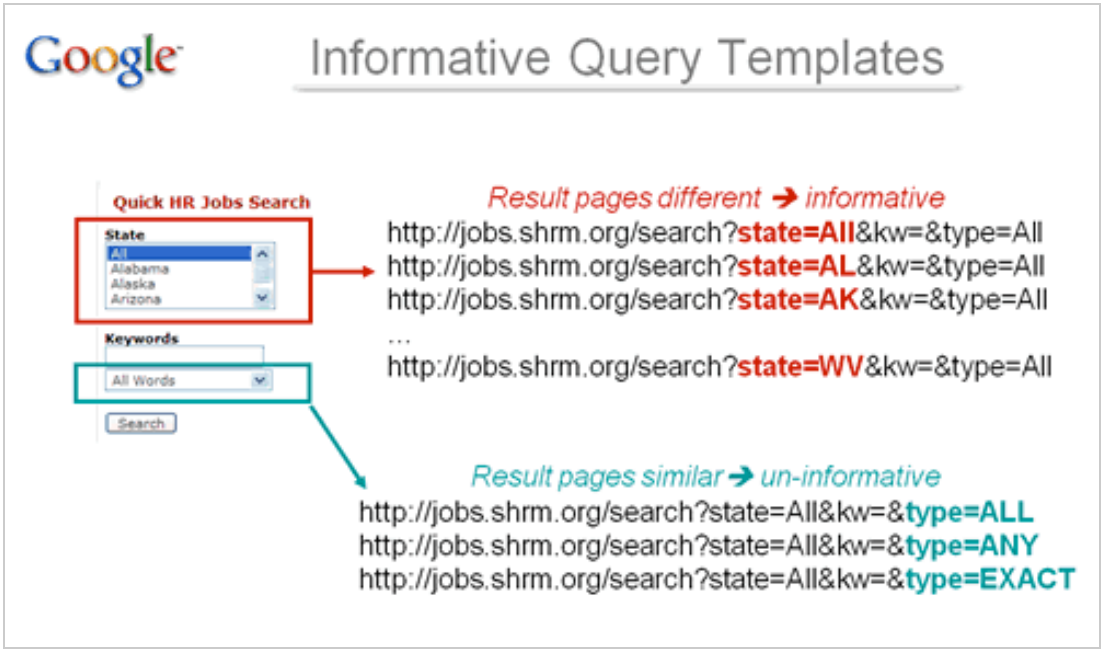
Nous allons voir que ce nouveau comportement induit l'apparition de nouveaux type de "spider traps" et a probablement conduit Google à recommander d'utiliser dorénavant l'*url rewriting*... avec modération.

Google a commencé à crawler le web invisible !

Dans un billet du 11 mars 2008 sur le blog de Google, un ingénieur de Google en charge du développement des crawlers, Jayant Madhavan (Jayant Madhavan et Alon Halevy ont en fait développé les bases de cette technologie au sein d'une startup baptisée Transformic voilà plusieurs années. Il semble que la société ait été rachetée par Google en 2005) a révèlé qu'ils avaient entrepris de lancer une expérience en vraie grandeur pour crawler le "web profond", et en particulier les pages cachées derrière un formulaire. Seules les pages appelées par des formulaires utilisant la méthode "get" sont crawlables avec ce système...

La méthode utilisée analyse le contenu des champs du formulaire et les paramètres dans l'url pour arriver à déterminer quelles sont les valeurs possibles dans les champs du formulaire. Une deuxième phase d'analyse cherche à deviner ensuite quelles combinaisons correspondent à des pages existantes. Au cours d'un colloque, Madhavan a expliqué que le site d'annonces automobiles US cars.com contient 500 000 pages de résultats, mais que les combinaisons des valeurs des champs du formulaires produisent 250 millions d'urls différentes !

Un exemple présenté par Jayant Madhavan montre comment son système est capable de recréer les syntaxes d'urls correspondant aux champs du formulaire :



De nouveaux types de spider traps ?

Depuis l'hiver 2008, on voit donc de plus en plus fréquemment Google tester des formes d'urls qui n'existent pas vraiment sur les sites. Il peut s'agir dans certains cas de tests (il y a eu par exemple des tests en vue d'un nouveau système d'authentification pour Google Webmaster Tools), de tentatives de crawl dans les pages de listings des sites cachés derrière des formulaires, ou d'erreurs classiques (url erronée collectée mais suivie par le crawler).

Parfois, on peut soupçonner aussi un comportement du crawler destiné à tester le rôle des paramètres pour éliminer des cas complexes de DUST (*Duplicate Urls Same Text*) difficiles à identifier et normaliser.

Ces nouveaux comportements ont fini par créer de nouvelles formes de spider traps. Depuis l'été 2008, Google informe les webmasters par le biais d'un message dans les webmaster tools :
Le nombre d'URL détectées par Googlebot sur votre site XXXXXXX est extrêmement élevé. Googlebot a trouvé un nombre de liens extrêmement élevé sur votre site. Cela peut signifier qu'il existe un problème de structure dans l'URL de votre site. Googlebot explore peut-être un grand nombre d'URL distinctes qui pointent vers un contenu identique ou semblable, ou explore des parties de votre site qu'il n'est pas censé explorer. Par conséquent, il est possible que Googlebot utilise une quantité de bande passante beaucoup plus importante que nécessaire, ou encore il peut être dans l'incapacité d'indexer intégralement le contenu de votre site.

Dans bien des cas, les webmasters qui reçoivent ces messages sont perplexes, car Google donne aussi un échantillon des urls qui posent problème, et il arrive fréquemment que ces urls ne soient présentes nulle part sur le site. Un coup d'oeil sur les logs néanmoins permet de se rendre compte que ces urls sont bel et bien appelées par les crawlers de Google.

Certaines formes d'url rewriting gênent à présent le crawl de Google ?

Ce nouveau comportement des crawlers pose d'autant plus de problèmes si la syntaxe des urls ne permet pas d'identifier correctement le rôle de chaque composant de l'url (répertoires, variables, paramètres). Dans ce cas, la normalisation des urls devient impossible.

Ceci explique probablement pourquoi Google a publié le 22 septembre 2008 un billet dans son blog pour expliquer dans quels cas l'url rewriting posait problème. Il s'avère que certaines

formes d'urls sont impossibles à "tester". Voici quelques exemples des formes d'urls posant problème d'après Google :

- www.example.com/article/bin/answer.foo/en/3/98971298178906/URL

- www.example.com/article/bin/answer.foo/language=en/answer=3/sid=98971298178906/query=URL

- www.example.com/article/bin/answer.foo/language/en/answer/3/sid/98971298178906/query/URL

- www.example.com/article/bin/answer.foo/en,3,98971298178906,URL

Quelques recommandations pour suivre le crawl des moteurs...

Trop de webmasters négligent dans leur travail de référencement le suivi du crawl des moteurs. Il est pourtant indispensable de suivre plusieurs indicateurs :

- le nombre de pages téléchargées, le volume de données téléchargé, et le temps de téléchargement (grâce à GWT : Google Webmaster Tools).
- les messages diffusés dans GWT (détection de spider traps).
- les erreurs liées au sitemaps (un sitemap erroné ou qui n'est plus à jour peut perturber le crawl) toujours dans GWT.
- et analyser le comportement de crawl pour détecter d'éventuels problèmes à l'origine de changements de comportements.

Liens utiles

Le site commun de Yahoo!, Google et Bing sur les sitemaps :
http://www.sitemaps.org/fr/

La page de l'add on *Firefox user-agent switcher* pour voir comment Googlebot voit vos pages
https://addons.mozilla.org/en-US/firefox/addon/59

Pour analytics, vous trouverez la méthode ad hoc ici :
http://www.analytics.fr/google-analytics-seo-comment-mesurer-les-vistes-des-robots-et-crawlers-sur-votre-site/

Bibliographie

Trois articles complémentaires sur les crawlers publiés dans le passé par l'auteur de cet article sur le site Webmaster Hub :

Sur les différences pré et post printemps 2003 :
Quelques pistes pour comprendre le nouvel algorithme de Google (juillet 2004) :
http://www.webmaster-hub.com/publication/Quelques-pistes-pour-comprendre-le.html

A propos des techniques de crawl incremental
Les techniques évoluées d'indexation dans les moteurs de recherche : le problème de la fraîcheur des index (mars 2004)
http://www.webmaster-hub.com/publication/Les-techniques-evoluees-d.html

Les techniques évoluées d'indexation dans les moteurs de recherche (2e partie) : Les techniques de crawl sur index "ouverts" (mars 2004)
http://www.webmaster-hub.com/publication/Les-techniques-evoluees-d,83.html

Articles de référence :

Synchronizing a database to improve freshness. Cho, Junghoo; Hector Garcia-Molina (2000). Proceedings of the 2000 ACM SIGMOD international conference on Management of data: 117-128

Efficient Crawling Through URL Ordering. Cho, J. and Garcia-Molina, H. and Page, L. (1998) article inclus dans : Seventh International World-Wide Web Conference (WWW 1998), April 14-18, 1998, Brisbane, Australia.

The Evolution of the Web and Implications for an Incremental Crawler : Junghoo Cho, Hector Garcia-Molina / Department of Computer Science Stanford, CA 94305 December 2, 1999

Junghoo Cho. Crawling the Web: Discovery and Maintenance of a Large-Scale Web Data. Ph.D. thesis, Stanford University.

Parallel Crawlers : Junghoo Cho / University of California, Los Angeles - Hector Garcia-Molina / Stanford University 2002

Breadth-first search crawling yields high-quality pages. Marc Najork and Janet L. Wiener. In Proc. 10th World Wide Web Conference, pages 114–118, 2001.

An Adaptive Model for Optimizing Performance of an Incremental Web Crawler : Jenny Edwards, Kevin McCurley, John Tomlin 25 février 2002

Ranking the Web Frontier, by Nadav Eiron, Kevin S. McCurley, and John A. Tomlin, WWW Conference 2004.

Adaptive On-Line Page Importance Computation, Serge Abiteboul (INRIA-Xyleme), Mihai Preda (Xyleme SA), Grégory Cobena (INRIA-Xyleme) WWW2003, May 20--24, 2003, Budapest, Hungary

Focused crawling: a new approach to topic-specific Web resource discovery, Soumen Chakrabarti, Martin van den Berg, Byron Dom, WWW1999

Sitemaps: Above and Beyond the Crawl of Duty, Uri Schonfeld UCLA Computer Science Department, Narayanan Shivakumar
Google Inc. WWW2009

Recrawl Scheduling Based on Information Longevity : Christopher Olston Yahoo! Research , Sandeep Pandey Carnegie Mellon University, WWW2008

A three-year study on the freshness of Web search engine databases, Dirk Lewandowski1 Hamburg University of Applied Sciences

Brevets

Fonctionnement de Googlebot (avant 2003)

Distributed crawling of hyperlinked documents
Inventors: Jeffrey A. Dean. Craig Silverstein, Benedict Gomes, and Sanjay Ghemawat
Assigned to Google
United States Patent 7,305,610
Granted December 4, 2007
Filed: August 14, 2000

Crawl du web invisible :

Searching through content which is accessible through web-based forms
Invented by Alon Y. Halevy, Jayant Madhavan, and David H. Ko
US Patent Application 20060230033

Published October 12, 2006
Filed on April 5, 2006

Crawl prédictif

System and method for crawl ordering by search impact
Invented by Christopher Olston and Sandeep Pandey
Assigned to Yahoo.
US Patent Application 20090164425
Published June 25, 2009
Filed December 20, 2007

Anchor tag indexing in a web crawler system

Invented by Huican Zhu, Jeffrey Dean, Sanjay Ghemawat, Bwolen Po-Jen Yang, and Anurag Acharya
Assigned to Google
US Patent 7,308,643
Granted December 11, 2007
Filed July 3, 2003

Automatically Fetching Web Content with User Assistance
Invented by Amit Jaiswal, Arup Malakar, and Binu Raj
Assigned to Yahoo
US Patent Application 20090019354
Published January 15, 2009
Filed September 11, 2007

Yahoo!
Handling dynamic URLs in crawl for better coverage of unique content
Invented by Priyank S. Garg and Arnabnil Bhattacharjee
US Patent Application 20080091685
Published April 17, 2008
Filed: October 13, 2006

Philippe Yonnet, ***Directeur Technique @Position (http://www.aposition.com) et président de l'association SEO Camp (http://www.seo-camp.org/)***

Réagissez à cet article sur le blog des abonnés d'Abondance :
http://abonnes.abondance.com/blogpro/2009/07/les-dernieres-evolution-de-google-en.html