Le contenu dupliqué : un cauchemar pour les moteurs ou pour les webmasters ?

Retour au sommaire de la lettre

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

Le contenu dupliqué, ou "duplicate content", est l'un des soucis principaux des webmasters s'intéressant au référencement. Les moteurs de recherche ont fait d'énormes progrès dans la détection des différents contenus similaires, proches ou identiques sur le Web, mais sans obtenir encore des résultats parfaits dans la pratique. Mais c'est également oublier qu'il existe différentes formes de duplicate content, et donc, en fonction de celles-ci, différentes façons de les combattre. Petite revue d'effectif des différents contenus dupliqués identifiés par les moteurs de recherche actuels sur le Web et des remédes à y apporter...

Beaucoup de webmasters évoquent régulièrement sur les forums de discussion leur crainte de recevoir une pénalité pour "contenu dupliqué". En réalité, nous le verrons dans cet article, il n'existe aucune raison pour qu'un moteur de recherche "pénalise" un site pour ce motif. Cela ne veut pas dire que la présence de contenu dupliqué n'est pas pénalisante pour un site. Mais cela n'a rien de systématique, car, point primordial, tout dépend de ce que l'on appelle un "contenu dupliqué"...

Comment définir le contenu dupliqué ?

En effet, il est important de bien définir les différentes formes de contenu dupliqué, car chaque forme a une origine différente et pose des problèmes différents.

Dans un premier temps, il faut distinguer trois cas différents :
- les **documents dupliqués** (qui se retrouvent en double sur la Toile, sous différences URL, mais code, textes, images, autres contenus sont identiques) ;
- les **documents quasi dupliqués** (*near duplicates* en anglais : le contenu peut-être légèrement différent, et/ou le code qui présente ces contenus est différent) ;
- les documents **partiellement dupliqués** (seule une partie du contenu se retrouve dans d'autres pages).

Les documents parfaitement dupliqués tirent souvent leur origine d'un problème ou d'une maladresse technique. On peut citer les cas suivants, qui créent effectivement des situations dans lesquelles la même page est accessible avec des URL différentes :

1. Les cas de **DUST (*Duplicate URL Same text*)** : le même contenu est accessible par des URL différentes au sein du même domaine/sous domaine. Nous reviendrons en détail plus loin sur les causes de DUST, qui sont essentiellement techniques.
2. Les **sites miroirs :** le même contenu est accessible depuis différents domaines, ou sous domaines. Tous ces "hôtes" peuvent appartenir à la même personne ou à la même organisation, ou constituer des miroirs gérés par d'autres webmasters pour assurer une meilleure disponibilité des contenus.
3. Les **marques blanches :** le même contenu est proposé à l'identique par d'autres sites, dans leur domaine, avec l'accord du producteur de contenu.
4. Le **contenu copié :** le résultat est le même que pour une marque blanche, sauf que ce contenu est copié sans autorisation.

Les documents quasi-dupliqués ont des origines tantôt similaires, tantôt totalement différentes des sources de documents dupliqués :
- Le même contenu est employé sur un site différent, avec une maquette différente (cas typique de la dépêche AFP reprise sur de nombreux sites).
- Le même contenu peut être utilisé dans différents modèles de pages au sein du même site : le code change, mais pas le contenu.
- Le contenu est réarrangé, réordonné, mais le contenu reste fondamentalement le même (problème de la navigation "à facettes" et des listes réordonnées par date, par prix...).
- Le contenu est copié, ou repris, sans autorisation du producteur de contenu.

Les documents partiellement dupliqués ont une origine le plus souvent identique à celles évoquées au dessus. La différence essentielle est qu'ici, une portion, parfois très réduite du document se retrouve à plusieurs endroits différents sur le Net. Nous verrons par la suite que l'élimination de tels cas pose des parfois plus de problèmes qu'elle n'en résout, et les moteurs les traitent par conséquent différemment.

Enfin, il existe une catégorie de "*near duplicates*" un peu spéciale, qui se caractérise par une plus grande difficulté de détection : **les textes plagiés**. Dans un texte plagié, les termes changent, mais pas la structure générale du texte ni les informations contenues dans le texte. Suivant la proximité du texte plagié avec son original, la détection du plagiat sera soit aisée, soit franchement impossible. Mais ces textes ne posent pas de problèmes fondamentaux aux moteurs de recherche, sauf un : l'**identification des sources** d'information, et des **documents** originaux.

Pourquoi les moteurs de recherche n'aiment pas le contenu dupliqué ?

Les moteurs de recherche cherchent à détecter les contenus dupliqués pour quatre raisons fondamentales :

- L'**optimisation du processus de crawl** : le nombre de pages que leurs crawlers peuvent télécharger sur une période donnée est limitée : il leur faut donc éviter le plus possible de télécharger plusieurs fois la même page.

- **L'optimisation de la taille de l'index** : Pourquoi conserver plusieurs exemplaires parfaitement identiques d'un document ? Dans de nombreux cas, on ne peut pas prédire que deux pages sont identiques avant de les avoir téléchargées. Lorsqu'on repère des cas de pages parfaitement dupliquées, on peut procèder soit à une **canonicalisation**, soit à une **normalisation** :
  * la **canonicalisation** consiste à identifier que les documents dupliqués correspondent à une variante de la même url, par exemple : http://domaine.com/mapage.html et http://www.domaine.com/mapage.html, http://www.domaine.com/page.html et http://www.domaine.com/page.html?aff=12345, etc.
  Dans ce cas, toutes les variantes sont éliminée au profit d'une version de l'url canonique. Le choix de la version qui doit être conservée est parfois complexe.
  * la **normalisation** consiste à éliminer des pages dont l'url est totalement différente pour les regrouper sous une **url normalisée.** La normalisation n'est en fait pratiquée que dans des cas particuliers, car en général, les moteurs préféreront garder les différentes versions de la page, par peur de confondre un cas de duplication due à une erreur technique, et à un cas de duplication volontaire.

– **La simplification et la rectification de la matrice des liens :** la canonicalisation ou la normalisation conduisent à remplacer, dans la matrice de liens, les variantes d'URL par leur version canonicalisée ou normalisée.
Dans ce cas, l'url conservée reçoit la somme des "jus de liens" (PageRank et autres) reçus par toutes les variantes, ainsi que tous les "anchor texts" des liens pointant vers les variantes. Cette manipulation technique permet donc d'éviter d'avoir des biais, parfois très importants, dans le calcul du PageRank d'une page et dans la prise en compte des signaux contenus dans les liens entrants. Par ailleurs, réduire le nombre d'URL diminue la taille de la matrice des liens, et permet des calculs plus faciles...

- **La pertinence perçue demande de la diversité :** Si la seule réponse présentée à un utilisateur, pour une requête donnée, est une liste de 10 pages au contenu strictement identique, il existe hélas une probabilité élevée que cette page dupliquée 10 fois ne soit pas pertinente, et que les réponses pertinentes soient en page 2 !
Les moteurs essaient donc de présenter autant que possible des pages "proches" de la requête mais différentes les unes des autres.
Signalons également au passage que dans de nombreux cas de spamdexing, les pages générées sont des contenus dupliqués : savoir éliminer les documents non originaux relève donc également de la **lutte contre le spam.**

Cela signifie donc que la détection et l'élimination éventuelle du contenu dupliqué ne s'effectue pas en une seule passe, mais bien par étapes successives :
- au cours du processus de crawl d'abord,
- lors de l'indexation ensuite,
- et enfin, lors de la construction des pages de résultats.

Le fait même que ces trois phases successives existent montre qu'à chaque étape, il reste des pages dupliquées ou quasi-dupliquées dans le système.

En fait, les techniques de détection du contenu dupliqué ont fait des progrès depuis dix ans, mais aucune n'est parfaite et le problème est loin d'être résolu.

Comment éviter d'indexer des documents dupliqués ?

Comparer les empreintes digitales des documents

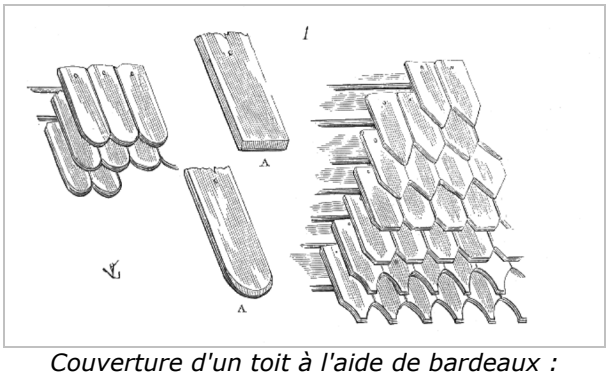
La première idée pour éviter de stocker dans un index des documents parfaitement dupliqués (Altavista était réputé en contenir environ 30% dans ses années de gloire), est de calculer une **empreinte digitale** (*fingerprint*) pour chaque document. Le principe est de créer un code sur 64bits, ou 128bits, caractéristique du document. Les "fingerprints" sont donc des informations de taille réduite, faciles à stocker, et faciles à comparer. La méthode ressemble au "hashage MD5", mais avec un objectif un peu différent.

Chaque nouveau document crawlé voit donc son empreinte digitale comparée aux empreintes connues. Si son empreinte est inconnue, cela signifie que ce document est différent de tous les documents connus. Si son empreinte est connue, cela signifie que ce document est une copie très proche, voire identique, d'un document précédemment crawlé (en effet, deux documents peuvent partager la même empreinte et ne pas être des copies parfaites).

Cette méthode simple est toujours utilisée dans la plupart des moteurs, sous une forme plus ou moins sophistiquée selon l'utilisation. Elle a par contre pour inconvénient de ne pas détecter les cas de quasi-duplications.

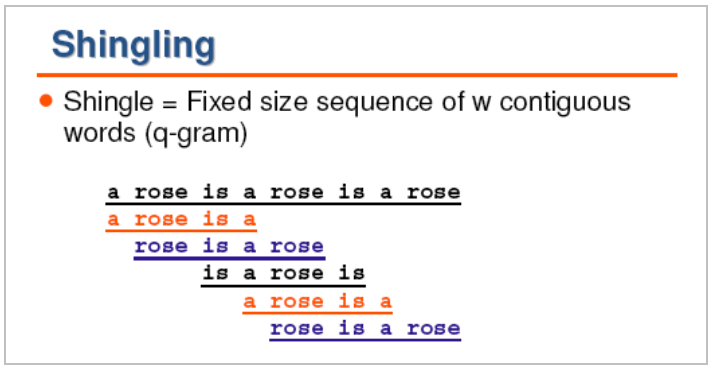
L'algorithme des bardeaux ("shingling")

Cette approche introduite par A. Broder en 1995 a été rapidement considérée comme "la" solution. L'idée consiste à découper les textes en une série de "bardeaux" (*shingles* en anglais) de longueur fixe, et se recouvrant, et de comparer les bardeaux entre eux.



Couverture d'un toit à l'aide de bardeaux :
la comparaison avec la méthode utilisée et illustrée
ci-dessus saute aux yeux

Un calcul de similarité sur les bardeaux à l'aide de l'**indice Jaccard** permet de détecter des documents quasi dupliqués, caractérisés par un indice élevé.



Le problème de cette méthode est qu'elle relativement coûteuse à mettre en oeuvre, car elle nécessite des calculs complexes.

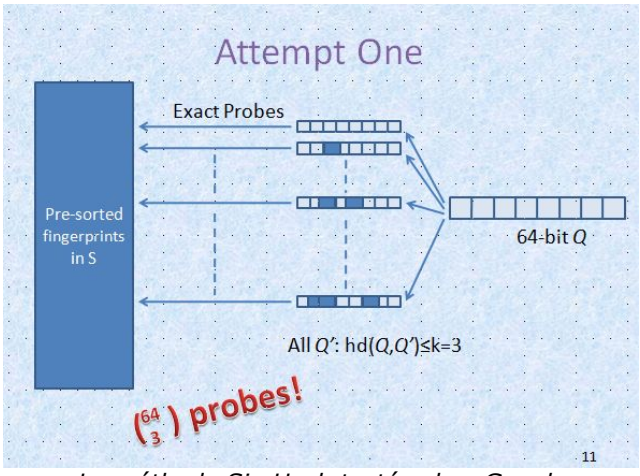
Simhash, la méthode de Charikar

Une deuxième méthode a été introduite en 2002 par Moses Charikar. Elle consiste à ne pas essayer de calculer une empreinte digitale unique, mais un "hash", un code numérique qui essaie de "représenter" le contenu d'un document, en utilisant des projections pour réduire le nombre de dimensions à ce qui est stockable dans le "hash code". Pour un hash sur 64bits, deux documents seront considérés comme similaires s'ils partagent au moins 62 bits en commun.

Cette méthode, baptisée **Simhash** est probablement celle qui a été retenue par Google. Plusieurs indices le laissent penser, notamment un article de Monica Henzinger présentant une évaluation comparée de Simhash par rapport aux bardeaux, et démontrant que les tests effectués chez Google concluaient à la supériorité écrasante de la méthode développée par Chimrikar. Une méthode similaire est citée dans les brevets de Google sur la détection des *near duplicates,* et SimHash est citée dans plusieurs articles signés par Singh Manku et Arvin Jain, deux ingénieurs de Google. Ces derniers ont notamment expliqué comment ils utilisaient **Mapreduce** et **GFS** (Google File System) pour optimiser les calculs des Simhash.

Ces deux méthodes ont fait l'objet de nombreuses recherches et de perfectionnement, et enfanté des nouvelles approches, parfois combinant les deux algorithmes. Notons toutefois qu'elles génèrent toutes des "faux positifs" (deux pages ayant un contenu différent seront pourtant détectées comme en "duplicate content"), et que la détection des documents quasi-dupliqués génère avant tout des cas "gris", où l'on ne sait pas clairement quoi décider : garder les deux documents ? Ou en éliminer un ?

Ces approches ont surtout un gros défaut : elles fonctionnent bien, mais *a posteriori*. Il faut donc crawler la page d'abord pour s'apercevoir qu'il s'agit d'un document dupliqué ou quasi dupliqué. Mais peut-on prédire qu'un document sera une copie ou une quasi copie d'un document connu ? Dans certains cas, la réponse est oui !



La méthode SimHash testée chez Google
extrait d'une présentation de Singh Manku et Arvin Jain

Le problème des "boilerplates"

L'un des problèmes épineux sur lesquels les moteurs de recherche on fait des progrès est l'identification dans les pages des **"boilerplates"**. Les "boilerplates" sont des blocs de code, généralement réservés à la navigation (menus, blocs de liens), que l'on retrouve sur de nombreuses pages, voire toutes les pages d'un site. Evidemment, il est très utile lorsque l'on veut comparer le contenu de plusieurs pages de le faire sur le contenu spécificique à une page, et donc de savoir faire abstraction des "boilerplates", qui par définition sont semblables d'une page à l'autre.

Trois méthodes différentes mais proches ont été publiées par les principaux moteurs de recherche à propos de l'identification de ces boilerplates :
- *VIPS* chez Microsoft.
- *Template Extraction* chez Yahoo!.
- *Segmentation of Visual Gaps* chez Google.

Ce type d'approche permet d'améliorer de manière considérable la détection des "near duplicates".

Les méthodes pour ne pas crawler dans la poussière...

Les moteurs de recherche ont à faire face à la très grande variété des cas de **DUST (Different URL Same Text : URL différentes, même texte)**. Il s'agit d'URL différentes pointant vers un contenu identique, ces URL étant produites par un paramétrage imparfait des serveurs webs, des DNS, ou des programmes web.

Voici quelques exemples classiques de génération de DUST :

- http://domaine.tld/index.html <=> http://domain.tld (pb dit de l'"url canonique") ;
- http://news.google.com <=> http://google.com/news (pb des hôtes virtuels) ;
- http://domaine.tld/~shuri <=> http://domain.tld/people/shuri (alias et liens symboliques) ;
- Réécriture d'URL : http://domaine.com/article_123.html => http://domain.com/article.php?id=123
- Paramètres qui ne modifient pas le contenu ou presque : print=1, color=blue, user=123456, SID=4566HJ8JF8LHS, affiliate=1245, referrer=890
- Changement dans l'ordre des paramètres : http://www.domaine.com/catalogue.php?marque=12&modele=678 <=> http://www.domaine.com/catalogue.php?modele=678&modele=12
- Navigation à facettes en javascript : http://www.domaine.com/catalogue.php?marque=12&modele=678&taille=38&couleur=bleue&order=prix_asc (en fait toutes ces pages ont le même contenu que : http://www.domaine.com/catalogue.php?marque=12&modele=678)
- etc.

Dans les cas les plus extrêmes, certaines sources de DUST génèrent même des **"spider traps"**, des pièges à robots : une erreur technique génère un grand nombre d'URL différentes pour le même contenu, voire un nombre d'URL infini. Evidemment, crawler cet espace infini demande... un temps infini. Cependant, tous les crawlers disposent d'un système de garde fou qui les conduit à arrêter de crawler une zone du web qui semble constituer un "spider trap".

En 2007, **Zif BarYossef**, un chercheur de Google, et **Udi Manber**, un chercheur israelien devenu en 2008 Vice-Président de Google, et qui est en charge aujourd'hui de tous les produits search pour la firme de Mountain View, ont co-publié un article (*Do not crawl in the DUST*... cf. bibliographie à la fin de cet article) présentant un nouvel outil pour gérer la problématique du DUST : **Dustbuster.**

Dustbuster est un outil à double fonction. Il est d'abord conçu pour identifier les cas de DUST, et pour construire des règles à partir des informations collectées sur ces cas. L'objectif de ces règles est de prédire qu'une page ayant une forme d'URL donnée est en fait un cas de

contenu dupliqué, et qu'il est donc inutile de la crawler dans la mesure où son contenu est censé être déjà connu.

La deuxième fonction de Dustbuster est de "tester" des échantillons de pages en les crawlant vraiment, afin de vérifier la validité de la règle.

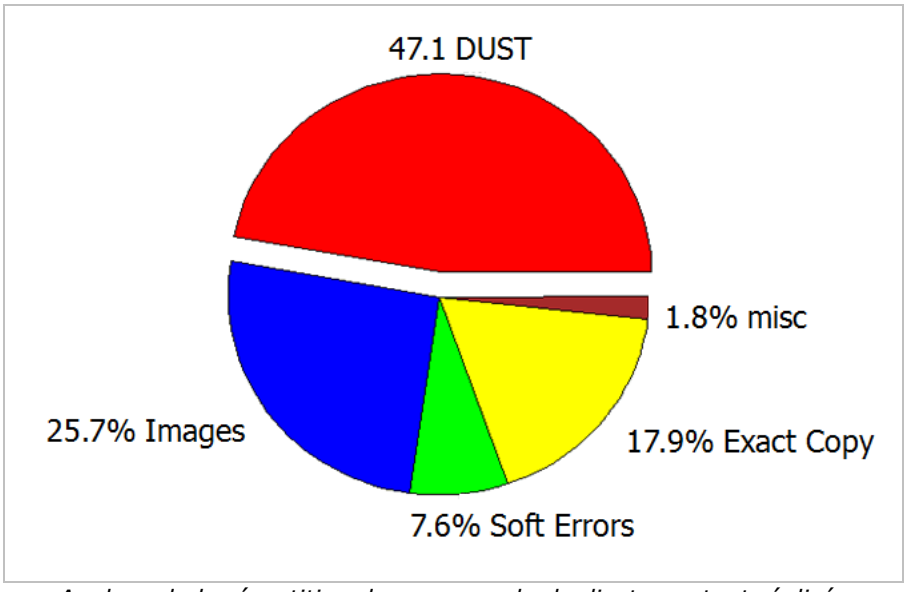
Il semble que ce mode de fonctionnement "prédictif" soit réellement utilisé par Google. L'observation des logs serveurs a permis de constater que le comportement de crawl de Google avait bel et bien changé à partir de l'été 2008, et que certaines formes d'URL, correspondant à des pages DUST, ont cessé d'être crawlées systématiquement par Google.

Il faut remarquer que Dustbuster repose essentiellement sur une analyse fine de la liste des paramètres dans les URL, et de leur rôle, ainsi que des effets d'un changement dans les valeurs de ces paramètres. Cette analyse est beaucoup plus facile avec des URL de type :
http://www.domaine.com/page_produit.php?id_produit=123&marque=12&affiliate=124
qu'avec ce genre d'url réécrite :
http://www.domaine.com?nike/chaussures-la-lakers-jaunes-124

Un effet de bord de Dustbuster est donc de rendre les URL avec paramètres beaucoup plus "Google friendly" qu'avant. Ce qui a conduit Google à communiquer sur ces changements et à préciser notamment que dans certains cas, ils préféraient une syntaxe avec des paramètres dans l'url plutôt qu'une version réécrite difficile à interpréter. Voici quelques exemples cités dans dans ce billet du 22 septembre 2008, issu du blog anglophone destiné aux webmasters :
http://googlewebmastercentral.blogspot.com/2008/09/dynamic-urls-vs-static-urls.html

Dans les exemples cités, figurent ces formes d'url rewriting qui posent toutes un problème à 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

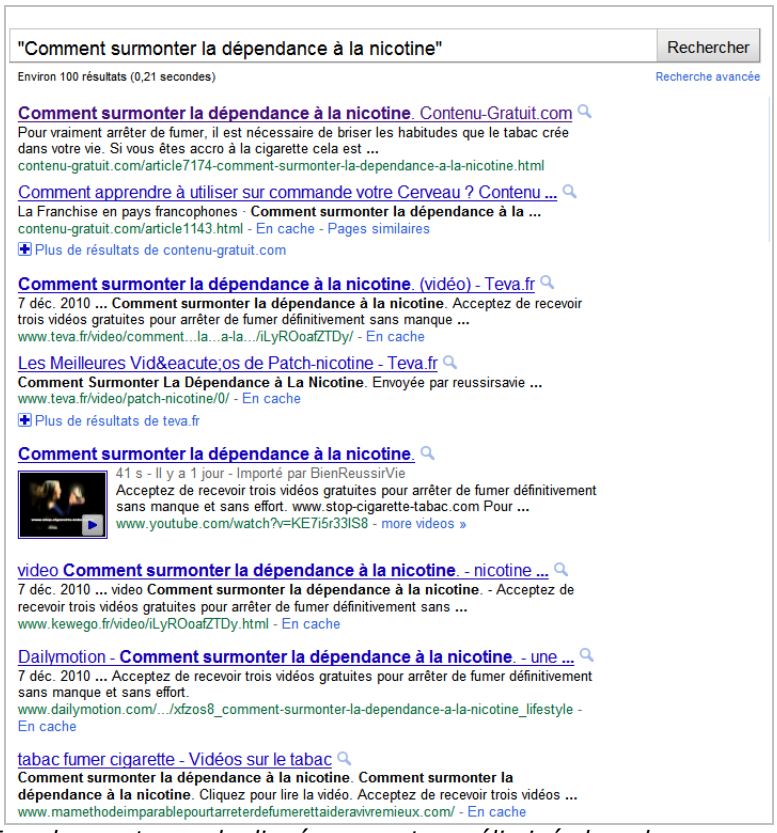
D'une manière générale, toutes les URL dont les paramètres sont difficiles à interpréter dans le cadre de règles de type "Dustbuster" peuvent poser problème à Google.



Analyse de la répartition des sources de duplicate content réalisée par l'équipe d'Udi Mamber : les cas de DUST représentent 47,1% des cas.

Comment éviter que dix résultats identiques polluent les résultats ?

Toutes les techniques de détection du contenu dupliqué, ou quasi-dupliqué, ont leurs limites. Il reste donc des pages dans l'index qui sont réellement du "duplicate content". Ce contenu est parfois volontairement conservé dans l'index, dans les cas "gris" où l'on soupçonne que des internautes pourraient chercher dans le site B un contenu déjà présent sur le site A.



Tous les contenus dupliqués ne sont pas éliminés lors du processus de crawl et d'indexation, loin de là : exemple ci-dessus d'une requête qui renvoie une centaine de versions différentes du même article et de la même vidéo...

Cependant, le souci de présenter avant tout des résultats pertinents conduit à procéder à une "déduplication" des contenus dupliqués sur les pages de résultat.

Dans un premier temps, on procède à une "clusterisation" des résultats émanant du même site.

Ensuite, on détermine à la volée, dans les résultats, la liste des pages qui semblent être des contenus dupliqués. Cette technique s'appelle le **"duplicate collapsing" :** en francais la "réduction des documents dupliqués".

La méthode traditionnelle pour procéder à cette déduplication consiste à **comparer les snippets** : si deux pages ou plus présentent des titres identiques et une description identique, ou proche, l'une des versions sera éliminée. Reste à savoir quelle page conserver...

Il y a quelques années, l'idée dominante était d'essayer de conserver :
- soit la page la plus importante : en gros celle disposant du plus fort PageRank.
- soit la page qui arrivait en tête des résultats.

Il semble que le système soit devenu un peu plus sophistiqué. En effet la page disposant du plus fort PageRank n'est pas forcément la page la plus légitime dans le contexte d'une requête précise. C'est plus certainement la page qui reçoit des liens des autres pages classées (comme dans le principe de l'algorithme du **localrank**).

Par ailleurs, la page la mieux classée n'est pas forcément celle qui aurait été choisie par l'internaute s'il avait eu accès à toutes les pages.

Dans un brevet publié par Microsoft, décrivant un filtre de duplicate content, on a un aperçu des critères qui pourraient idéalement être utilisés pour affiner cette règle :

• L'**extension du nom de domaine** (les internautes préfèrent cliquer sur les .com) ;
• L'**url la plus claire et la plus courte** ;
• L'**url qui mène le plus rapidement au résultat** (pas de redirection, ou le meilleur temps de réponse) ;
• La **popularité** (PageRank ou autre rank : déjà cité) ;
• Le **mot clé cherché figure dans l'url ;**
• La page correspond au **lieu de résidence de l'internaute** ou correspond à **sa langue ;**
• **Analyse des clics** en présentant plusieurs versions, avant de déterminer la page préférée.

A cette liste, il faut ajouter un point qui n'est pas dans le brevet de Microsoft : la détermination du document original (le premier a avoir présenté le contenu, chronologiquement parlant).

Comment reconnaître le texte original sur le Web ?

L'un des effets secondaires les plus désagréables du "duplicate collapsing", c'est que la version qui est écartée peut être la version originale, et la version conservée, la copie. Donc il reste un problème épineux à régler pour les moteurs de recherche : savoir reconnaître quelle est la source originale pour un contenu donné.

Des représentants de Google (Vanessa Fox, en particulier) ont longtemps affirmé dans le passé que Google arrivait particulièrement bien à gérer ce problème. Ce n'est pas toujours vrai : la copie est trop souvent référée à l'original, surtout quand la copie est plus "importante" que l'original, et finit par recevoir plus de liens.

En réalité, il s'agit d'un problème complexe à résoudre.

Notamment, la **date d'apparition du document sur le net** ne peut pas être facilement pris en compte dans la pratique : un moteur de recherche ne connaîtra que la **date où il a découvert l'URL** correspondant à un contenu donné la première fois. Rien ne garantit que le premier document découvert soit le premier document publié. Se fier à la **date de publication, ou à la date de création de la page** ne sert à rien également : cette information ne figure pas toujours sur la page, peut ne pas être la date de première publication, mais la date de republication, et peut être falsifiée ou erronée. Le même raisonnement est applicable à l'indication de la source : rares sont les webmasters qui indiquent de manière claire la source des documents publiés.

Dernièrement (le 24 novembre 2010), Google a introduit à titre expérimental une nouvelle mesure destinée à régler ce problème pour Google News, avec la création de deux balises meta : **syndication-source** et **original-source** :
• <meta name="syndication-source" content="http://www.publisherX.com/wire_story_1.html">
• <meta name="original-source" content="http://www.example.com/burglary_at_watergate.html">
La première balise est destinée à être utilisé sur le site qui reprend le flux, et la seconde sur le site qui publie le flux.

Comment gérer les différents cas de contenu dupliqué ?

La méthode la plus efficace pour éviter de créer des contenus dupliqués, est de "réparer" tous les problèmes techniques et toutes les configurations de sites qui peuvent créer des cas de DUST ou conduire à la diffusion d'un contenu dans plusieurs domaines ou sous domaines.

Dans la pratique, il n'est pas toujours simple de supprimer toutes les causes de duplication de contenus. Il reste néanmoins toute une batterie de mesures possibles pour traiter le problème, ou pour aider Google à repérer l'url canonique à conserver. En voici une liste (non exhaustive) :

- **Indiquer dans le sitemap l'url canonique préférée** (cela suppose que l'on n'insère que celles-ci dans le fichier, et pas les versions dupliquées).
- **Bloquer l'indexation des pages dupliquées** à l'aide de la balise meta name='robots' et de l'**attribut noindex**, ou par une directive **x-robots-tag.** Rappel : on ne peut pas bloquer l'indexation d'une page avec une ligne dans un robots.txt, ce fichier sert à prévenir la découverte du contenu d'une page, mais n'empêche pas que son url soit indexée.
- Créer **une redirection 301** de la page dupliquée vers l'url à conserver.
- Utiliser la **balise <link rel='canonical' href='[url_canonique]'>** pour indiquer l'URL à conserver. Attention, comme pour le sitemap, cette méthode permet de suggérer aux moteurs la version à conserver, rien ne garantit que cette suggestion sera prise en compte.
- Utiliser la **fonctionnalité "traitement des paramètres"** proposée dans les Google Webmaster tools : vous pourrez indiquer tous les paramètres inutiles, et éventuellement corriger la façon dont Google gère les autres paramètres. La même fonctionalité existe dans les Webmaster Tools de Yahoo!. Un paramétrage correct peut supprimer des sources de DUST problématiques de Google.

Attention : si vous êtes débutant ou étourdi, laissez Google gérer cela. Une erreur de gestion des paramètres peut gravement nuire à votre référencement.

Si vous diffusez votre contenu par le biais de marques blanches ou de flux RSS, d'autres mesures s'imposent :

- Essayer autant que possible d'**héberger les marques blanches sur votre site**, éventuellement à l'aide d'une redirection DNS de type CNAME. Cela vous permettra de contrôler étroitement la façon dont votre contenu est repris chez votre partenaire.
- Vous pouvez aussi essayer de proposer un **contenu syndiqué différent** de celui que vous avez sur le site, soit **limité à une partie de votre contenu**.
- Penser à mettre **un lien systématique vers votre site dans votre contenu syndiqué**. Si le contenu est repris tel quel, cela aidera le moteur de recherche à choisir votre page plutôt que celle de vos partenaires ou des "scrapers".

Conclusion : faut-il craindre le contenu dupliqué ?

Même si le contenu dupliqué est un cauchemar pour les moteurs de recherche, il n'y a aucune volonté particulière de "pénaliser" spécialement les webmasters parce que leurs pages sont dupliquées. Au contraire, les moteurs déploient beaucoup d'énergie et de moyens pour crawler et indexer un web imparfait, et ont fait beaucoup de progrès dans cette direction. Tout se passe comme si les webmasters aussi étaient considérés comme des victimes de ces problèmes soit purement techniques soit inhérents au fonctionnement habituel du web.

En règle général, l'impact réel des problèmes de contenu dupliqué est assez marginal par rapport à l'influence du contenu, du netlinking ou de la structure du site. Très souvent, l'énergie déployée par les webmasters pour éviter le "duplicate content" est disproportionnée par rapport au problème. Ne pas lancer une marque blanche, un partenariat, ou un site reprenant une partie du contenu d'un site existant, par peur du duplicate content est donc souvent excessif. Certes l'un des sites (parfois les deux hélas) risque de sous-performer et de recevoir moins de trafic que s'il ne contenait que du contenu original, mais souvent, les deux versions dupliquées rapporteront plus que la seule page originale. De plus, la réutilisation du même contenu dans un *template* et un contexte différent suffit souvent à échapper à la plupart des filtres. Seul le dernier (le "duplicate collapsing") s'activera, et encore, sur certaines requêtes, et pas toutes.

Néanmoins, ne pas traiter les causes de DUST, ou ne pas prévenir la diffusion de pages dupliquées, peut fortement nuire à l'expression du potentiel d'un site, surtout s'il est de taille importante.

Il ne faut donc pas avoir peur du duplicate content, et encore moins d'une pénalité qui n'existe pas, mais il faut néanmoins être conscient que ne pas traiter les sources de contenu dupliqué reste pénalisant pour le référencement. Identifier les cas de "duplicate content" et savoir comment traiter le problème font donc partie, clairement et durablement, des tâches incontournables en matière de SEO.

Bibliographie

Vanessa Fox talks about Webmaster Tools and Duplicate Content
http://www.stonetemple.com/articles/interview-vanessa-fox.shtml

Credit where credit is due : un billet du blog Google News de Google.
http://googlenewsblog.blogspot.com/2010/11/credit-where-credit-is-due.html

Demystifying the "duplicate content penalty"
http://googlewebmastercentral.blogspot.com/2008/09/demystifying-duplicate-content-penalty.html

Articles scientifiques :

"Identifying and Filtering Near-Duplicate Documents"
(Algorithme "Shingling", ou algorithme des bardeaux)
http://www.cs.brown.edu/courses/cs253/papers/nearduplicate.pdf
Andrei Z. Broder., 2000. , Proceedings of the 11th Annual Symposium on Combinatorial Pattern Matching. UK: Springer-Verlag

"Similarity estimation techniques from rounding algorithms"
(Random projection - Simhash)
http://www.cs.princeton.edu/courses/archive/spring04/cos598B/bib/CharikarEstim.pdf
Charikar, M., 2002. , In Proc. 34th Annual Symposium on Theory of Computing (STOC 2002)
Exemple d'implémentation de l'algorithme en Python :
http://bibliographie-trac.ub.rub.de/browser/simhash.py

"Do Not Crawl in the DUST: Different URL with Similar Text",
(Dust buster)
http://www2007.org/papers/paper194.pdf
BarYossef, Z., Keidar, I., Schonfeld, U., 2007.
16th International world Wide Web conference, Alberta, Canada, Data Mining Track, 8-12 May.

"Finding replicated web collections",
http://rose.cs.ucla.edu/~cho/papers/cho-mirror.pdf
Cho, J., Shivakumar, N., Garcia-Molina, H., 2000
ACM

SIGMOD Record, Vol. 29, No. 2, pp. 355 - 366, June.
Duplicate and Near Duplicate Documents Detection: A Review
http://www.eurojournals.com/ejsr_32_4_08.pdf
J Prasanna Kumar, P Govindarajulu

Brevets

Clustering by previous representative
http://patft.uspto.gov/netacgi/nph-Parser?Sect1=PTO2&Sect2=HITOFF&u=%2Fnetahtml%2FPTO%2Fsearch-adv.htm&r=1&p=1&f=G&l=50&d=PTXT&S1=7,836,108.PN.&OS=pn/7,836,108&RS=PN/7,836,108
Invented by Joachim Kupke, David Michael Proudfoot
Assigned to Google, US Patent 7,836,108, Granted November 16, 2010, Filed: March 31, 2008

Duplicate document detection in a web crawler system
http://patft.uspto.gov/netacgi/nph-Parser?Sect1=PTO2&Sect2=HITOFF&u=%2Fnetahtml%2FPTO%2Fsearch-adv.htm&r=1&p=1&f=G&l=50&d=PTXT&S1=7,627,613.PN.&OS=pn/7,627,613&RS=PN/7,627,613
Invented by Daniel Dulitz, Alexandre A. Verstak, Sanjay Ghemawat, Jeffrey A. Dean
Assigned to Google, US Patent 7,627,613, Granted December 1, 2009, Filed: July 3, 2003

System and method for optimizing search results through equivalent results collapsing

http://appft1.uspto.gov/netacgi/nph-Parser?Sect1=PTO1&Sect2=HITOFF&d=PG01&p=1&u=%2Fnetahtml%2FPTO%2Fsrchnum.html&r=1&f=G&l=50&s1=%2220060248066%22.PGNR.&OS=DN/20060248066&RS=DN/20060248066
Invented by Brett D. Brewer
Assigned to Microsoft, US Patent Application 20060248066, Published November 2, 2006,
Filed: April 28, 2005

Philippe Yonnet, *Global SEO Strategist, WEB DMUK (Londres) – Easyroommate / Vivastreet*

Réagissez à cet article sur le blog des abonnés d'Abondance :
http://blog-abonnes.abondance.com/2010/12/le-contenu-duplique-un-cauchemar-pour.html