

Sites en scroll infini : comment les rendre « Google friendly » ?



Par Philippe Yonnet

Domaine :	Recherche	Référencement
Niveau :	Pour tous	Avancé

De nombreux sites prennent aujourd'hui en compte la technique dite du « scroll infini » ou « infinite scroll » pour leur interface utilisateur. En d'autres termes, l'internaute visualise un contenu de départ qui s'étoffe au fur et à mesure que la page est scrollée vers le bas. Ces techniques, à base de Javascript et d'Ajax, sont nativement très peu compatibles avec la compréhension par Google des contenus affichés, ce qui génère de gros problèmes SEO. Il est donc nécessaire de coder ce type de site de façon très précise et adéquate pour les rendre « Google friendly ». Ce qui n'est pas, il faut bien le dire, le cas de très nombreux sites actuels utilisant ce type d'interface...

Les "long scrolling pages" ou "infinite scrolling pages" (pages à défilement infini ou pages infinies en français) sont à la mode et fleurissent de plus en plus sur les sites web. Ces pages font une utilisation intensive du Javascript et de fonctionnalités Ajax. Or, sans précautions particulières, l'emploi de ces techniques peut poser de sérieux problèmes de référencement. Google s'en est inquiété et a diffusé des recommandations pour créer un code crawlable par ses robots.

Qu'est-ce qu'une "infinite scrolling page" ?

Lorsqu'une page web utilise la technique des « pages à défilement infini », un nouveau contenu est chargé dynamiquement à chaque fois que l'utilisateur fait glisser la page vers le bas. Ce contenu est en général chargé de manière asynchrone en Ajax. Suivant la quantité de données qu'il est possible d'afficher, l'utilisateur aura

Nous allons voir dans cet article comment coder correctement ces pages pour conserver une expérience utilisateur avancée, tout en gardant un contenu indexable par les moteurs.

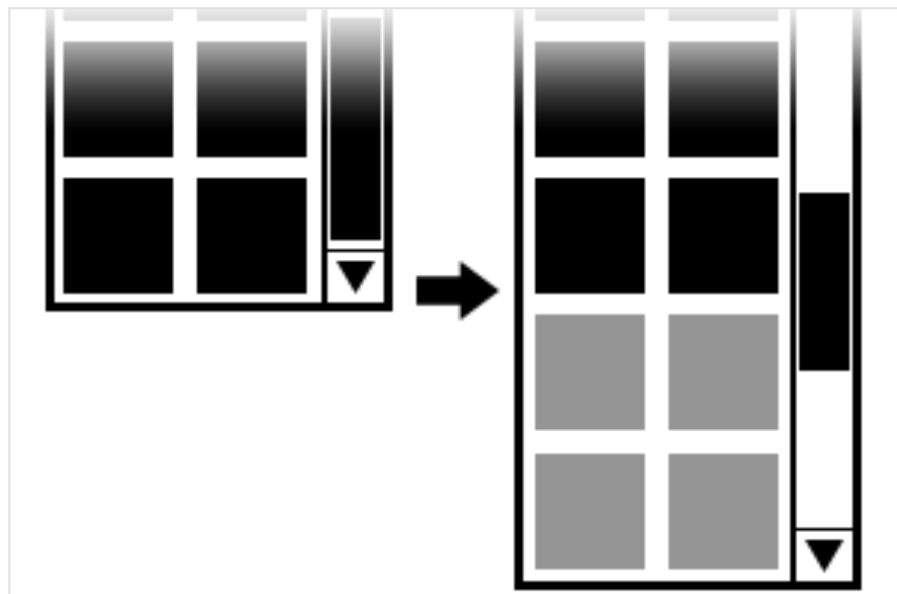


Fig.1. Principe de l'infinite scrolling page

l'impression soit de consulter une très longue page (il finit par atteindre le dernier item), soit même de consulter une page de taille infinie...

L'un des exemples les plus connus d'utilisation de ce type d'interface utilisateur sont les pages de résultats de Google Images et de Bing Images. Mais on les trouve aussi sur un grand nombre d'outils sociaux (Tumblr, Pinterest...), comme chez certains sites éditoriaux et même sur des sites e-commerce.

Ces interfaces sont également de plus en plus adoptées car elles sont beaucoup plus agréables à utiliser sur une tablette ou un smartphone, en mode "touch". Dans ce contexte d'utilisation, le défilement peut être géré aussi bien dans le sens vertical, que dans le sens horizontal.

Quels problèmes pose cette technique pour le SEO ?

Si les interfaces de type "infinite scroll" permettent à un utilisateur d'accéder rapidement à un contenu, ce n'est pas forcément le cas pour les moteurs de recherche !

Dans une implémentation classique d'une page infinie, le code de la page téléchargé suite à l'appel d'une URL via une requête http ne contient qu'un nombre limité d'items à afficher (par exemple 20), suffisant



Fig.2. Implémentation de l'infinite scroll sur Pinterest

pour réaliser le contenu du premier écran affiché. Le code de la page contient un programme en Javascript, qui va s'exécuter sur le navigateur et qui ira demander au serveur 20 résultats supplémentaires, puis 20 autres, puis 20

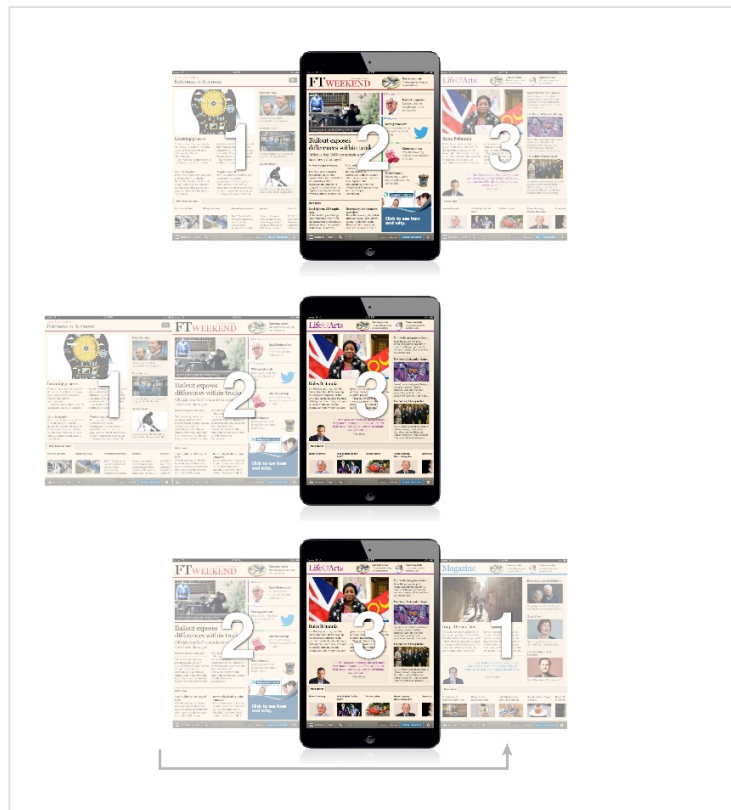


Fig.3. Exemple de « page infinie » avec un défilement horizontal actif en usage mobile sur le site du Financial Times

autres, etc., à chaque fois que l'utilisateur interagira avec la page pour afficher un nouvel écran.

Mais le robot d'exploration du moteur de recherche, lui, n'aura accès facilement qu'au code du premier état de la page, soit les 20 premiers items. Tous les contenus chargés en Ajax sont compliqués à aller explorer pour Google, en dépit des progrès indéniables que le moteur a faits dans ce domaine. Dans la pratique, ces contenus doivent être considérés comme absolument non explorables par les moteurs.

Si de plus ce contenu non explorable contient des liens vers d'autres URL, l'utilisation de ce design peut empêcher l'exploration d'un grand volume de contenus...

Heureusement, il suffit de prendre quelques précautions dans la manière de coder les modèles de pages de type « infinite scroll » pour rendre l'ensemble du contenu parfaitement crawlable.

Comment rendre une page à défilement infini compatible SEO ?

Il existe différentes implémentations compatibles SEO, mais une seule est préconisée par Google, que nous verrons plus loin. La technique la plus courante consiste à :

1. Coder en amélioration progressive, c'est-à-dire écrire un code qui fonctionnera lorsque le Javascript est désactivé, et les cookies non supportés, et sans support des sessions.

Ensuite, si le Javascript est activé, les cookies activés et/ou l'historique de navigation disponible permettront de proposer une expérience plus riche à l'utilisateur.

2. Dans la pratique, cela signifie que le code de départ contient une pagination normale, qui se retrouvera masquée lorsque le Javascript est activé.

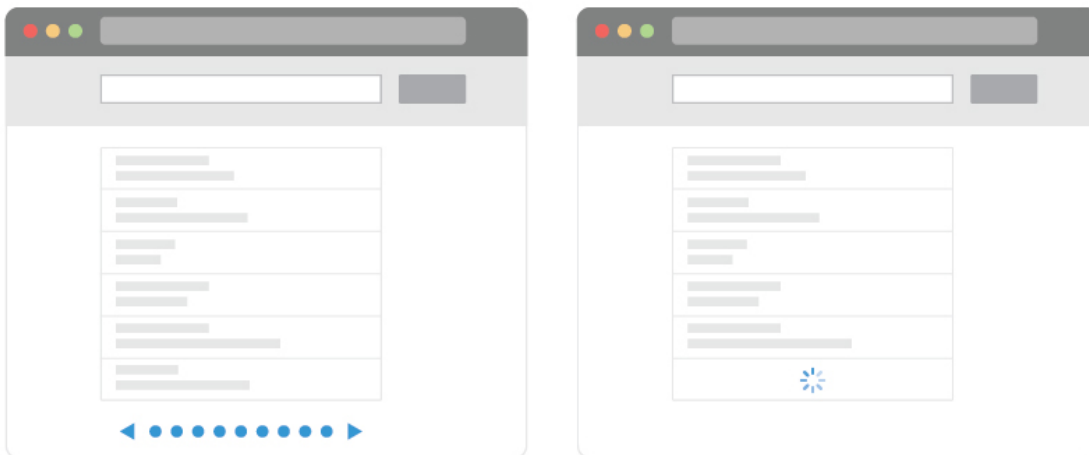


Fig.4. Les deux interfaces en action : à gauche, Javascript désactivé, la page s'arrête et fait apparaître une pagination normale, avec des liens pointant vers des contenus paginés. A droite, la page avec le Javascript activé, la pagination disparaît et le comportement de « scroll infini » apparaît.

3. Les liens contenus dans la pagination pointent vers des pages paginées qui contiennent une partie du contenu de la « long scrolling page ». Dans la pratique, ces pages peuvent parfois nécessiter l'utilisation d'un template différent pour la page 1 et les pages paginées explorables par les moteurs... Ce qui n'est pas idéal en termes de maintenabilité du code et de simplicité de l'implémentation.

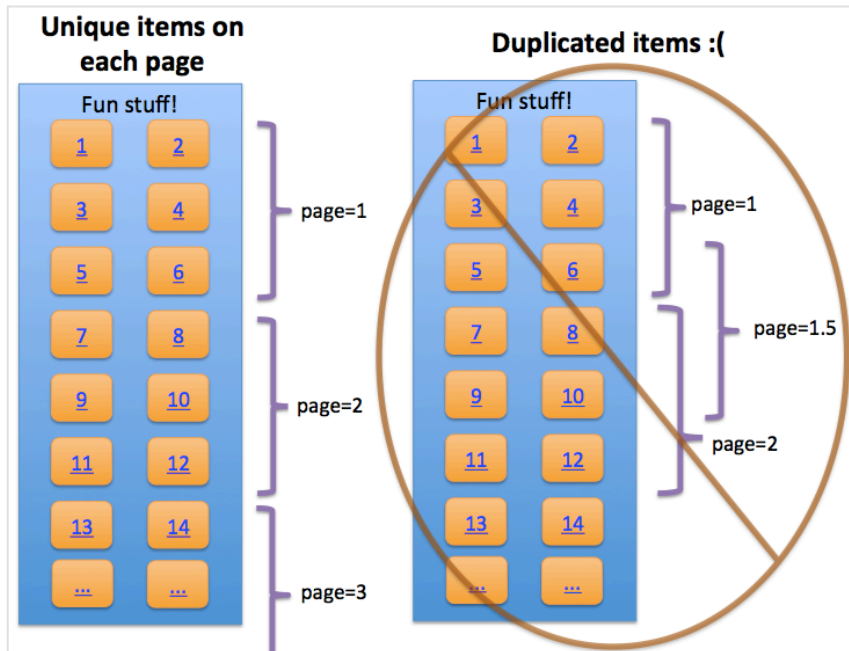


Fig.5. La bonne manière de découper les contenus en sous pages : il faut éviter le recouvrement. Source : Google.

Les recommandations de Google à propos des pages infinies

Dans un post du blog de Google destiné aux webmasters daté du 13 février 2014, John Mueller, Maile Ohye, et Joachim Kupke donnent (en anglais) des conseils très détaillés sur la bonne implémentation de l'infinite scroll : <http://googlewebmastercentral.blogspot.fr/2014/02/infinite-scroll-search-friendly.html>

La méthode proposée est une méthode "moderne" qui exploite les possibilités du HTML5. En voici un résumé pour ceux qui ne sont pas à l'aise avec la langue de Shakespeare :

1. Découper le contenu de la page infinie en "composants" affichant une partie du contenu.

Chaque composant constitue en fait une sous-page, dotée de sa propre URL. Ce groupe de pages doit être conçu pour être accessible quand le Javascript est désactivé dans le navigateur.

La quantité de contenu réparti dans les sous pages doit être un compromis entre l'objectif d'éviter d'avoir à parcourir trop de pages pour trouver un contenu, et le maintien de bonnes performances.

Dans tous les cas, le découpage en sous-pages doit éviter tout recouvrement entre les pages : il ne doit pas y avoir d'éléments communs entre les pages.

2. Choisir une structure d'URL facile à comprendre par un outil de recherche.

Chaque page composant doit être dotée d'une URL propre. L'utilisation de

paramètres derrière un caractère # est déconseillée, car cela conduit classiquement à des erreurs de configuration. L'utilisation d'une chaîne de paramètres à base de couples attribut/valeurs ne pose pas de problèmes, bien au contraire (éviter d'employer des méthodes de réécriture qui empêcheront le moteur d'identifier le couple attribut valeur)

- Syntaxe OK :

example.com/category?name=fun-items&page=1

- Autre syntaxe ok : example.com/fun-items?lastid=567

- Syntaxe moins optimale: example.com/fun-items#1

Vérifiez bien que l'appel de l'URL d'une page composant, affiche bien le contenu attendu, y compris en l'absence d'un cookie utilisateur, et quel que soit l'historique de navigation.

Par ailleurs, les URL des pages composants doivent également répondre à deux contraintes :

- Le contenu appelé par l'URL de la page composant doit pouvoir rester le même pendant deux semaines au moins. En particulier, cela signifie que si les paramètres définissant le contenu affiché correspondent à une période donnée (exemple : les articles du mois de mars, les articles du mois de

février etc...), il faut éviter des URL dont les paramètres sont relatifs !

exemple :

example.com/category/page.php?name=fun-items&days-ago=3

- Les paramètres qui définissent la « tranche » de contenu découpé et placé dans la sous-page composant doivent créer une page d'atterrissage qui a du sens pour l'utilisateur.

Par exemple :

example.com/fun-places?radius=5&lat=40.71&long=-73.40

Cette URL qui utilise des paramètres de géolocalisation ne crée pas forcément une page très utile, surtout en tant que réponse à une requête dans un moteur de recherche.

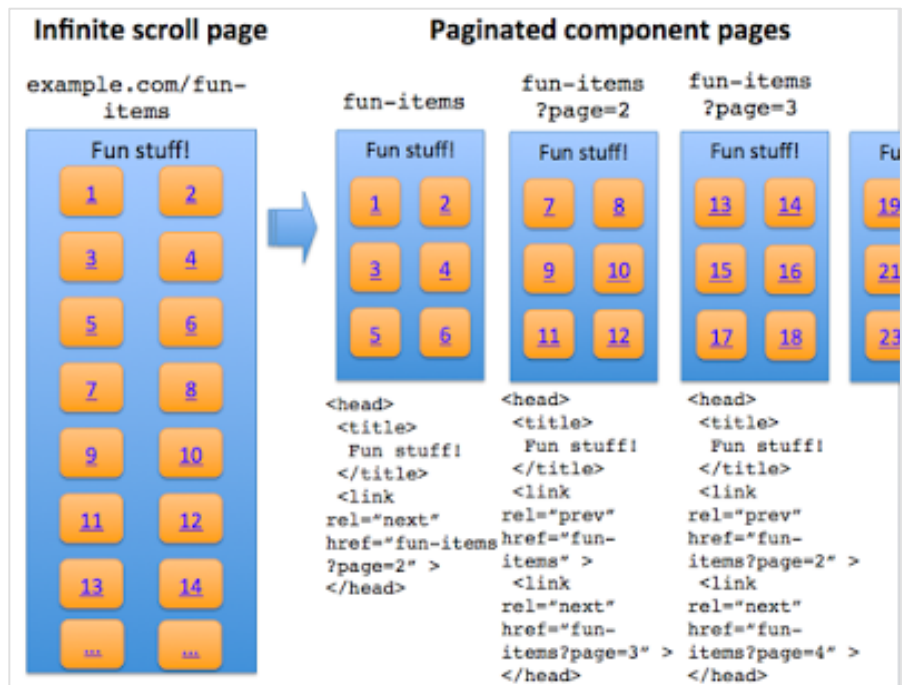


Fig.6. Exemple de découpage d'une page à défilement infini en sous pages paginées, avec un exemple d'url et de paramètres, et le « bon » code à utiliser dans la zone head. Notez que le contenu de la balise <title> reste le même à chaque fois : Google réassemblera la page virtuellement à partir de ses composants, donc inutile de créer des titres différents à chaque fois. Source : Google

3. Utilisez les balises <link rel=[next/prev] ...> dans les sous pages composants.

Attention, cette balise doit être utilisée dans la zone head du code html : les attributs rel=next/prev trouvés dans la zone <body> sont ignorés par Google, car ils peuvent figurer dans du contenu généré par les utilisateurs.

4. Utilisez la méthode *pushState* ou *replaceState* sur la page avec défilement infini (le choix entre *pushState* ou *replaceState* vous revient, en fonction de l'expérience utilisateur que vous voulez créer. *pushState* change l'URL du navigateur, et *replaceState* change les URL stockées dans l'historique).

Quoi qu'il en soit, l'utilisation de *pushState* est recommandée à chaque fois :

- Qu'une action de l'utilisateur ressemble à un clic de souris, ou à un changement de page ;
- Que l'on veut permettre aux utilisateurs de revenir sur le contenu paginé récemment consulté.

5. Testez le résultat !

- Vérifiez que les paramètres de la page changent comme attendu lorsque l'utilisateur fait défiler la page vers le haut ou vers le bas. La page de test créée par John Mueller montre le comportement typique de cette implémentation :

<http://scrollsample.appspot.com/items?page=3>

- Assurez-vous que les pages qui sont hors limites renvoient un code 404. Par exemple [example.com/category?name=fun-](http://example.com/category?name=fun-items&page=999)

[items&page=999](http://example.com/category?name=fun-items&page=999) doit renvoyer un code 404 s'il n'existe que 998 pages de contenu.

- Vérifiez également que votre page ne présente pas finalement de problèmes d'expérience utilisateur, tels que ceux évoqués dans cet article : *Infinite Scrolling Is Not for Every Website* par Hoa Loranger (<http://www.nngroup.com/articles/infinite-scrolling/>)

Attention : ces techniques doivent être utilisées à bon escient

La plupart des implémentations d'« infinite scroll » rencontrées sur les sites ne sont donc pas « Search Engine Friendly ». Dès qu'un site a recours à ces techniques, cela doit inciter un SEO à regarder le code pour le rendre compatible.

Heureusement, on peut donc assez facilement créer des pages « infinite scroll » parfaitement explorables par les moteurs de recherche... Par expérience néanmoins, de nombreuses erreurs sont possibles dans l'implémentation de ces techniques et le code devra être testé et retesté avant toute mise en ligne.

En revanche, ce n'est pas parce que ces interfaces à défilement infini peuvent être rendues parfaitement compatibles pour le SEO qu'elles sont toujours pertinentes. Procéder à un test A/B est recommandé pour voir si l'interface utilisateur choisie ne crée pas de sérieux problèmes d'usabilité.

Les pages à défilement infini démontrent en tout cas à quel point l'utilisation de l'Ajax peut être

problématique pour le SEO. Les moteurs, Google en tête, ont fait des progrès, et peuvent explorer toujours plus de contenu appelé par des scripts Ajax, mais dans la pratique, la plupart des implémentations ne permettent pas une exploration normale. Autant le savoir avant de faire des choix technologiques quasi définitifs pour votre site web...

Bibliographie et liens

Infinite Scrolling Is Not for Every Website par Hoa Loranger

<http://www.nngroup.com/articles/infinite-scrolling/>

Infinite scroll search-friendly recommendations par John Mueller, Maile Ohye, et Joachim Kupke

<http://googlewebmastercentral.blogspot.fr/2014/02/infinite-scroll-search-friendly.html>



Philippe YONNET, Directeur
Général de l'agence Search-Foresight, groupe My Media
(<http://www.search-foresight.com>).