

Drupal et SEO, où en est-on ?



Par Aymeric Bouillat

Domaine :	Recherche	Référencement
Niveau :	Pour tous	Avancé

Drupal est certainement, avec WordPress et Joomla, l'une des plateformes de création de sites web le plus souvent utilisée sur la Toile. Mais ce type de CMS n'est jamais livré "SEO friendly" par défaut. Il est nécessaire de bien le configurer et d'y rajouter les "briques" essentielles à son bon fonctionnement et à l'optimisation pour Google des sites créés. Voici donc une longue (mais nécessaire) check-list des points essentiels à prendre en compte pour obtenir une version de Drupal prête à gravir les positions dans les résultats du moteur de recherche leader !

Plus qu'un CMS, un framework

Le CMS Drupal continue d'évoluer : disponible dans sa version 8 depuis peu - la version 7 reste celle qui est la plus installée quotidiennement - de nombreuses instances de sites Web fonctionnant toujours sous Drupal 6. Bien que Wordpress soit le principal CMS utilisé dans le monde, avec un nombre de recherches mensuelles sur Google toujours en hausse (Fig.1), l'intérêt pour Drupal est quant à lui largement inférieur, mais plutôt stable (à l'inverse, Joomla est plutôt sur le déclin...)

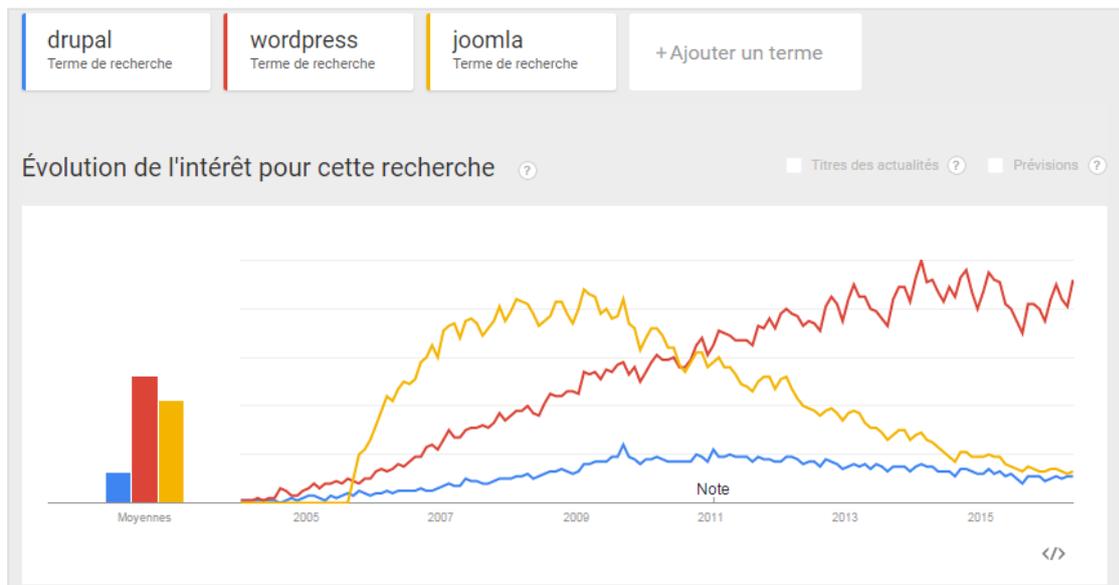


Figure 1 - Evolution du nombre de recherches mensuelles pour 3 CMS (Google Trends)

Drupal a une notoriété moindre car il est plutôt réservé aux initiés. En effet, le fait qu'il se rapproche plus d'un framework que d'un CMS le rend moins accessible : on peut créer n'importe quel type de site sous Drupal, mais le développement sera souvent plus complexe que d'installer un simple site sous Wordpress avec un thème « clé en main ».

C'est également un puissant CMS en termes de robustesse et de modularité. Un fois sa structure bien appréhendée, la mise en place d'un site sous Drupal s'avère conviviale pour un développeur. Il permet également de gérer de grandes quantités de contenus avec simplicité.

Parmi les grands sites sous Drupal et pour montrer la diversité de ces derniers, on pourra citer : <http://rue89.nouvelobs.com/>, <http://www.franceinfo.fr/>, <http://madame.lefigaro.fr/>, <http://www.voyages-sncf.com/>, <http://www.economie.gouv.fr/>, et le site de la Maison Blanche : <http://www.whitehouse.gov/> !

D'un point de vue du référencement naturel, Drupal n'est par défaut pas très « SEO-friendly », c'est une coquille vide sur laquelle il faudra concevoir votre thème et votre structure de site, pour que la couche SEO soit adaptée à l'utilisation qui sera faite du CMS par la suite.

Les thèmes sous Drupal sont bien moins nombreux que les thèmes Wordpress, et demandent souvent de nombreuses adaptations : avant de choisir ou d'acheter un thème, mieux vaudra regarder le code source et décrypter la façon dont le thème fonctionne, pour l'adapter à votre manière afin qu'il soit optimisé pour le référencement naturel. La façon dont il a été développé pourrait le rendre incompatible avec quelques plugins SEO. Bref, avoir un site performant pour le SEO sous Drupal demande de l'investissement en temps.

L'un des points faibles de Drupal est la façon dont les contributeurs non sensibilisés peuvent « casser le SEO » : que ça soit via la modification d'un titre qui modifierait son URL (et générerait une erreur 404), ou par l'éditeur Wysiwyg et l'utilisation abusive de balises <Hx>, il est indispensable de bien configurer l'outil pour chaque niveau d'utilisateur, et la façon dont ils pourront interagir avec le CMS. Drupal a cette capacité à laisser un accès restreint à différentes fonctions ou modules, encore faut-il bien les paramétrer en amont.

Nous allons rappeler dans cet article le fonctionnement de base de Drupal pour que vous puissiez mieux envisager la façon dont le SEO doit s'articuler autour du CMS, ainsi que passer en revue les principaux modules pour gérer votre SEO, et la façon dont ils doivent être configurés.

Principe de fonctionnement

Voici quelques rappels sur le principe de fonctionnement pour mieux appréhender Drupal d'un point de vue SEO par la suite. Mieux vous comprendrez la façon dont est structuré Drupal, mieux vous serez capable d'optimiser ses différents aspects pour améliorer votre référencement naturel. La force de Drupal est sa structure modulaire.

Les nœuds

Les nœuds sont l'élément principal du contenu d'un site sous Drupal. Un nœud (node) est un contenu, que ça soit une image, un article, une vidéo, etc.

Par défaut, Drupal fonctionne avec 2 types de nœuds : article et page, mais vous pouvez créer autant de types de contenu que vous le souhaitez (Fig.3). Il est possible d'ajouter des champs supplémentaires (CCK fields) pour chaque type de contenu, et c'est là que réside la principale force de Drupal : vous pourrez aisément créer un type de contenu vidéo en ajoutant un champ URL pour l'image miniature et l'URL du fichier vidéo, afin d'utiliser par la suite le player vidéo de votre choix, ou créer un type de contenu produit pour lui associer un prix et une disponibilité.

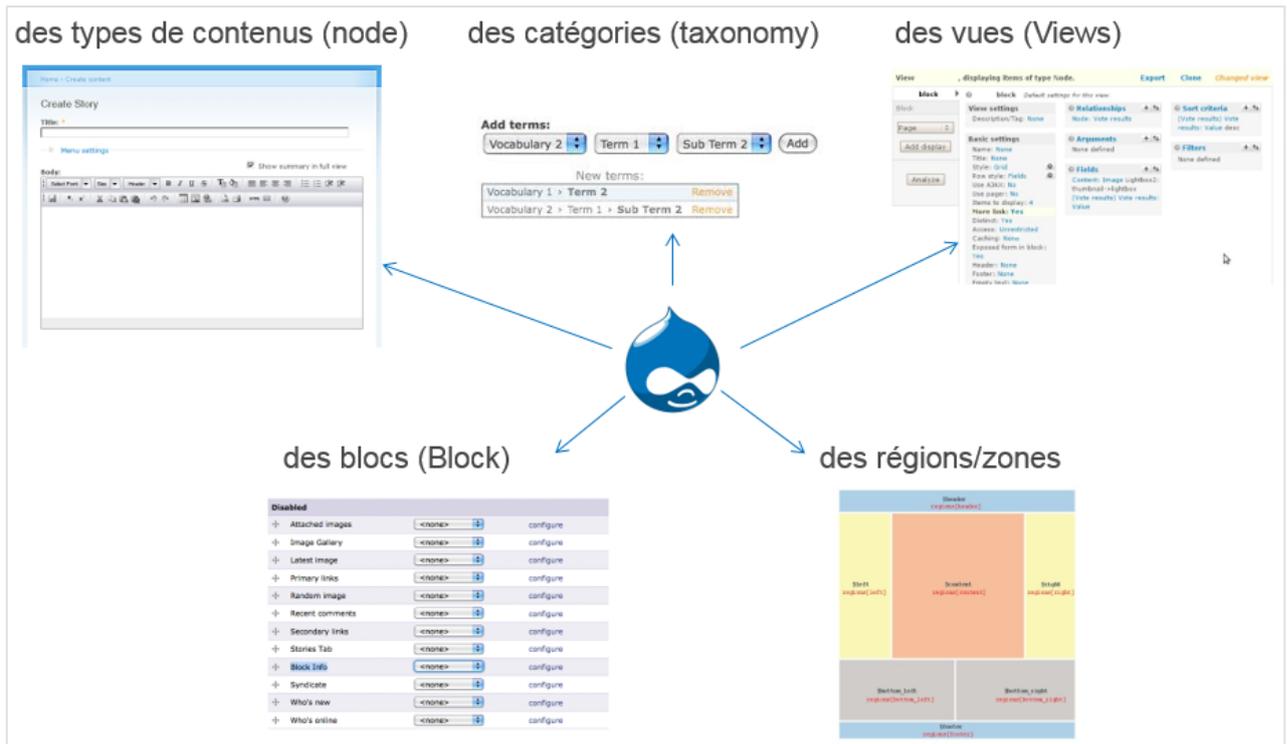


Figure 2 - Structure de Drupal.

Accueil » Administration » Structure

Types de contenu

[+ Ajouter un type de contenu](#)

NOM	ACTIONS
Article (Nom système : article) Utilisez les <i>articles</i> pour des contenus possédant une temporalité tels que des actualités, des communiqués de presse ou des billets de blog.	modifier gérer les champs gérer l'affichage supprimer
Page de base (Nom système : page) Utilisez les <i>pages de base</i> pour votre contenu statique, tel que la page 'Qui sommes-nous'.	modifier gérer les champs gérer l'affichage supprimer
Produit (Nom système : produit)	modifier gérer les champs gérer l'affichage supprimer

Figure 3 - Création de différents types de contenu.

Un certain nombre d'éléments sont paramétrables autour de chaque type de contenu (Fig. 4), à savoir les différents champs disponibles et à afficher (Fig. 5), comme la date ou l'auteur d'un article, la présence de commentaires, la présence dans le menu, mais aussi d'autres paramètres comme la mise en avant sur la page d'accueil ou l'intégration du type de contenu dans le sitemap XML.

Tous ces éléments doivent être définis de façon précise dès la création du site, pour éviter d'avoir à revenir en arrière pour compléter de nouveaux champs sur d'anciens contenus.

Figure 4 - Modification d'un type de contenu.

On utilisera généralement le champ Titre pour remplir l'élément `<h1>` dans le code source du template, la balise `<title>` étant gérée par les variables du module Meta Tags (onglet Meta Tags disponible dans l'édition d'un nœud. Pour récupérer le titre d'un nœud, on utilisera les variables `$title`, ou `$node->title`, ce qui donnera dans le template `node.tpl.php` : `<h1>$node->title</h1>`.

Cela permettra au contributeur de gérer la balise `<title>` et la balise `<h1>` de façon indépendante (Ex : titre : « Les acteurs de la série Dexter », `h1` : « Casting Dexter ») pour jouer sur la sémantique.

La création de champs supplémentaires est possible et vous permettra de gérer la balise hreflang par exemple, si l'utilisation du module portant le même nom ne convient pas à vos besoins.

ÉTIQUETTE	NOM SYSTÈME	TYPE DE CHAMP	WIDGET	ACTIONS
✚ Title	title	Élément du module Node		replace
✚ Body	body	Texte long et résumé	Zone de texte avec un résumé	modifier supprimer
✚ XML sitemap	xmlexport	XML sitemap module element		
✚ URL redirects	redirect	Redirect module form elements		
✚ Paramètres des chemins d'URL	path	Path module form elements		
✚ Meta tags	metatags	Meta tag module form elements.		
✚ Ajouter un nouveau champ				
<input type="text"/> Étiquette		- Choisir un type de champ - ▾ Type de données à stocker.	- Sélectionnez un widget - ▾ Élément du formulaire pour l'édition des données.	
✚ Ajouter un champ existant				
<input type="text"/> Étiquette		- Sélectionnez un champ existant - ▾ Champ à partager	- Sélectionnez un widget - ▾ Élément du formulaire pour l'édition des données.	

Figure 5 - Gestion des champs pour un type de contenu.

L'appel de ces différents champs dans les templates HTML pourra être personnalisé pour l'utilisation de rich snippets par exemple. Les types de nœuds disponibles et la façon dont ils s'affichent doivent donc être adaptés aux besoins SEO.

La taxonomie

La taxonomie représente la hiérarchisation des contenus dans le CMS. C'est la clé du maillage interne de votre site. Mieux vos contenus seront catégorisés et classés par thématique, plus vous pourrez en tirer profit pour votre SEO via les fonctionnalités de Drupal par la suite, afin de développer les relations entre vos différents contenus grâce aux liens internes.

Chaque contenu peut être associé à un ou plusieurs termes. Le « terme » permet de regrouper différents nœuds associés à la même thématique, sémantique... Le « vocabulaire » correspond lui à un groupe de plusieurs « termes ». Exemples :

Vocabulaire : *Séries*, Termes : *Dexter, Games of Thrones, Breaking Bad, Engrenages...*

Vocabulaire : *Type de série*, Termes : *Aventure, Action, Comédie, Thriller,...*

Une même série TV pourra donc être reliée à différents termes, au travers de différents vocabulaires (série, type de série).

La structure de votre site et la façon dont les contenus peuvent être liées (pensez maillage) doit être clairement définie en amont, afin de tirer profit au maximum de vos contenus.

Les vues

Les vues permettent d'afficher de pages de listing : type de contenus (article, vidéo, page) d'une même catégorie (terme). Les vues vont vous permettre d'associer à une URL, les derniers nœuds d'une catégorie spécifique en fonction de plusieurs critères (fig. 6).

The screenshot shows the 'View name' configuration page in Drupal. At the top, the view name is 'Articles' with a system name of 'articles'. Below this, there is a 'Description' section. The main configuration area includes:

- Afficher:** Contenu
- of type:** Article
- tagged with:** sous-categorie
- sorted by:** Newest first

 The 'Create a page' section is checked and contains:

- Titre de page:** Articles
- Chemin:** /articles
- Display format:** Unformatted list, of teasers, with links (allow users to add comments, etc.), without comments
- Items to display:** 10
- Use a pager:** checked
- Create a menu link:** unchecked

Figure 6 - Création d'une vue sous Drupal (listing de contenus).

C'est une sorte de requête SQL simplifiée façon WYSIWIG, mais très puissante (fig. 7). Ainsi, pour chaque vue, pour pourrez décider des éléments à afficher : titre, chapô (exemple : 250 caractères max), image de l'article, date de publication, auteur, ordre d'affichage, etc. Chacun des champs à afficher dans le listing pourra être optimisé à l'affichage (code HTML avant et après chaque champ), ce qui vous permettra aisément de placer des balises Hn sur chaque titre de nœud par exemple.

The screenshot shows the 'Displays' configuration page for the 'Articles' view. It includes several sections:

- Page details:** Display name: Page, view Page
- TITRE:** Titre: Articles
- FORMAT:** Format: Unformatted list | Paramètres; Afficher: Contenu | Accroche
- FIELDS:** The selected style or row format does not utilize fields.
- FILTER CRITERIA:** Contenu: Publié (Oui); Contenu: Type (= Article); Contenu: Has taxonomy term (= sous-categorie) | Paramètres
- SORT CRITERIA:** Contenu: Post date (desc)
- PAGE SETTINGS:** Chemin: /articles; Menu: No menu; Access: Droit | Voir le contenu publié
- ENTÊTE:** Ajouter
- PIED DE PAGE:** Ajouter
- PAGER:** Use pager: Full | Paged, 10 items; More link: Non
- META TAGS:** Meta tags: Using defaults

Figure 7 - Paramétrage d'une vue.

Chaque vue crée disposera de sa propre URL (ex fig.7, « Chemin : /articles »). Les vues peuvent également être affichées dans des blocs (équivalent des widgets sous

Wordpress). C'est par l'intermédiaire des régions définies dans le thème, et des blocs qui seront associés à chacune d'entre elles, que le rendu d'une page sera effectué.

Des régions et des blocs

Chaque thème fonctionne avec un système de régions (fig.8). On va ainsi pouvoir décider des différents blocs à afficher autour de chaque nœud, via le module Blocs (natif), en disposant les différents blocs dans des régions (header, colonne de gauche...). La façon dont les régions sont agencées est défini par le thème utilisé, les régions étant souvent disposées autour du contenu principal de la page.

Pour rendre cohérent le balisage de headings (Hn) dans toutes vos pages, il est prioritaire de prendre en compte la façon dont les blocs vont s'afficher sur chaque template. Le système de blocs est très puissant et vous permettra d'afficher des blocs dans des régions spécifiques, et ce en fonction de différents critères (type d'utilisateur, URL de la page demandée, etc.).

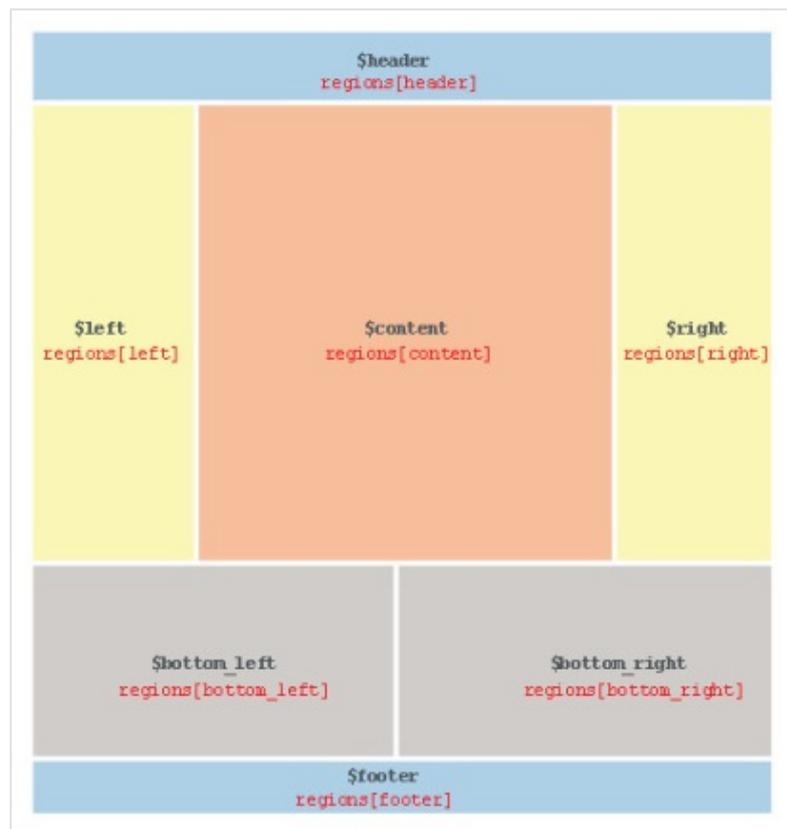


Figure 8 - Régions sous Drupal.

Cela vous permettra par exemple de restreindre l'affichage de certains blocs en fonction de l'URL demandée (fig.9 - exemple : afficher le bloc « articles connexes » via les vues, uniquement sur les pages contenant /articles/ dans l'URL).

Titre du bloc
Contenus récents
Supplante le titre par défaut du bloc. Entrer <none> pour ne pas afficher de titre ou laisser vide pour garder le titre par défaut du bloc.

Nombre d'éléments de contenu récent à afficher
10 ▼

PARAMÈTRES DE LA RÉGION
Préciser pour chaque thème la région dans laquelle ce bloc est affiché.

Bartik (thème par défaut)
Première barre latérale ▼

Paramètres de visibilité

Pages
Réservé à certaines pages

Types de contenu
Non restreint

Rôles
Non restreint

Utilisateurs

Afficher le bloc sur certaines pages seulement

Toutes les pages sauf celles listées

Seulement les pages listées

*/articles/

Figure 9 - Configuration d'un bloc.

Les blocs seront donc un excellent appui pour développer le maillage interne de votre site, en fonction de la façon dont sont organisés vos contenus via la taxonomie, pour renforcer la sémantique de chacune des pages, et développer la popularité des pages plus profondes dans l'arborescence.

Comme vous le voyez, la bonne compréhension de la structure modulaire de Drupal sera un véritable atout pour définir votre stratégie SEO et rendre votre site sous Drupal SEO-friendly.

Les modules SEO

De nombreux modules existent afin d'améliorer votre référencement naturel, que ça soit d'un point de vue technique pour la gestion des URL, ou pour la partie on-site (balises meta). Voici les principaux modules qui vous aideront à optimiser votre SEO sous Drupal.

Module robotstxt

Lors des mises à jour de Drupal, tous les fichiers contenus dans le fichier .zip de la nouvelle version doivent remplacer les existants (à l'exception du répertoire « sites »). Ainsi, si vous avez mis en place des adaptations dans le fichier robots.txt, elles risquent d'être écrasées par le fichier robots.txt par défaut, présent dans l'archive d'update, supprimant donc votre robots.txt personnalisé.

Pour éviter ce genre de problèmes et avoir plus de flexibilité lors des ajouts de directives, il est recommandé d'utiliser ce module qui vous permettra de modifier le contenu du fichier via un champ texte dans le back-office de Drupal (fig. 10). L'URL du

fichier <http://monsite.com/robots.txt> sera quant à elle gérée via une règle de réécriture. Ce module est également indispensable dans la configuration de Drupal en multi-sites.

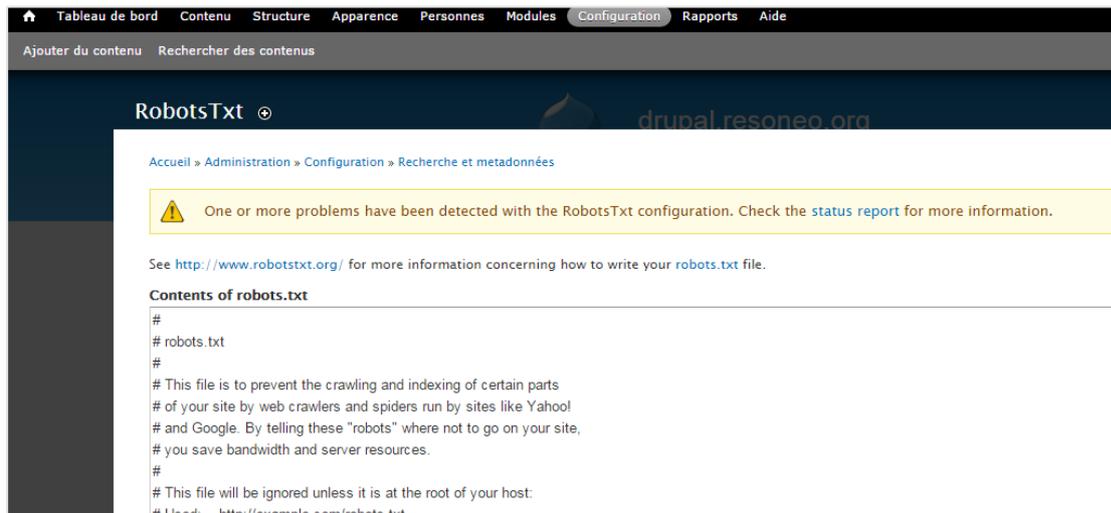


Figure 10 - Modification du robots.txt via le back-office.

Pour que le module fonctionne, il faudra penser à supprimer le fichier robots.txt pour que la configuration du module RobotsTxt soit pris en compte.

Adaptations à apporter au fichier robots.txt :

Le fichier robots.txt de Drupal par défaut n'est pas correct. En effet, il ne tient pas compte des URL avec ou sans trailing slash : www.example.com/admin/ et www.example.com/admin (sans slash à la fin) répondront avec le status http 200 :

```
[...]
# Paths (clean URL)
Disallow: /admin/
Disallow: /comment/reply/
Disallow: /filter/tips/
Disallow: /node/add/
[...]
```

Ce problème peut être résolu par le module "Global Redirect", mais il est préférable d'apporter ces modifications au fichier robots.txt en amont, pour limiter le crawl d'URL non pertinentes :

```
[...]
# Paths (clean URL)
#modified lines (without trailing slash)
Disallow: /admin
Disallow: /comment/reply
Disallow: /filter/tips
Disallow: /node/add
[...]
```

Module Path auto

Le choix de la syntaxe d'URL doit être défini en amont de la création d'un site web, pour éviter d'avoir à gérer des redirections et limiter les erreurs 404. Modifier ou adapter la structure d'URL peut avoir des effets négatifs sur le ranking, le format des URL doit donc être murement réfléchi.

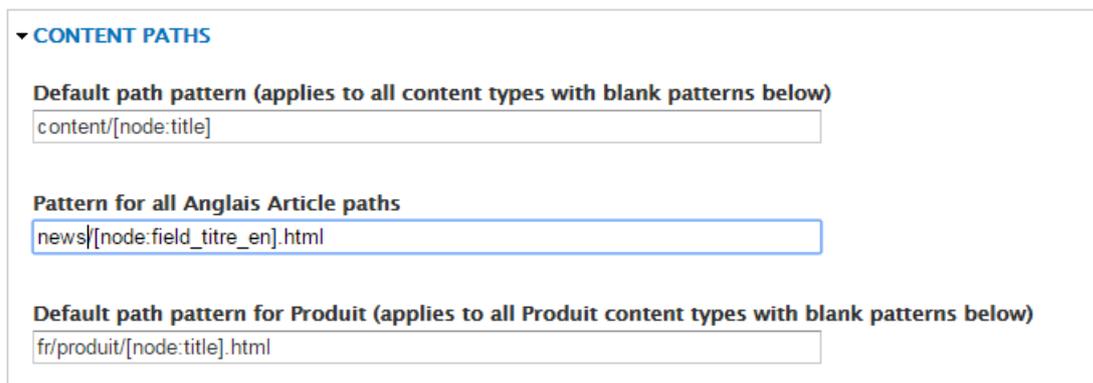
Les URL de base de Drupal ne sont pas SEO-friendly et sont sous cette forme :

- node/123 (contenu/noeud) ;
- user/5 (utilisateur) ;
- taxonomy/term/1 (catégorie).

Pour générer des URL SEO-friendly, l'option « clean URL » doit être cochée dans le back-office, et les alias d'URL seront créés par le « core module » Path, disponible dans la release standard. Le module "Path auto" (qui lui n'est pas installé par défaut) vous permettra d'automatiser la création des URL SEO-friendly, en fonction des différents types de contenu :

Structure des URL

Chaque type de contenu peut avoir sa propre règle de création d'alias. Les règles de création des alias d'URL (URL propres) peuvent être basées sur de multiples critères : la taxonomie (catégorie), l'ID du nœud, le titre du nœud, la langue du contenu, etc. (fig.11)



The screenshot shows the configuration page for the Path auto module. It features a section titled "CONTENT PATHS" with a dropdown arrow. Below this, there are three input fields for defining path patterns:

- Default path pattern (applies to all content types with blank patterns below)**: The input field contains the text `content/[node:title]`.
- Pattern for all Anglais Article paths**: The input field contains the text `news|[node:field_titre_en].html`.
- Default path pattern for Produit (applies to all Produit content types with blank patterns below)**: The input field contains the text `fr/produit/[node:title].html`.

Figure 11 - Path auto module.

Les règles doivent être définies pour chaque type de contenu à l'URL suivante : </admin/config/search/path/patterns>

La configuration du module "path auto" vous permettra également de définir plusieurs paramètres comme :

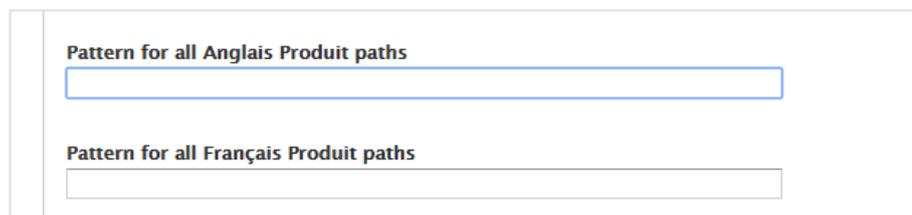
- le séparateur entre les mots dans l'URL : « - » sera donc adapté ;
- le passage de tous les caractères en minuscules ;
- la transformation des caractères spéciaux (retirer les accents) : *prêt à porter* deviendra donc [pret-a-porter](#) ; la transformation des caractères spéciaux n'est appliquée que dans le cas où le module "Translitération" est utilisé (en Drupal 7), c'est natif en Drupal 8.

- les stops words à supprimer des URL : d', l',à, de, le, les, une, un, etc.

Module pathAuto_i18n

Par défaut, juste en activant plusieurs langues, vous pourrez créer des paths anglais pour les contenus anglais et des paths français pour les contenus français (cas d'utilisation du module Content Translation pour le multilinguisme). Si par contre, vous souhaitez une URL anglaise et une URL française pour la même "page" technique (par exemple dans le cas d'utilisation d'Entity Translation), alors pathauto_i18n_node sera nécessaire (sinon, 2 nœuds seront créés : [/node/17](#) pour une page FR, [/node/18](#) pour la page en, et les nœuds seront liés entre eux via le module Content Translation).

Le module pathAuto_i18n offre donc la possibilité de créer plusieurs alias pour chaque nœud, afin de ne pas vous retrouver avec des URL FR sur le site anglais (ex : [/fr/ma-page](#) et [/en/ma-page](#)). Avec les sous-modules "Pathauto i18n node" et "Pathauto i18n taxonomy", un pattern spécifique d'URL pourra être configuré pour chaque type de noeud, et ce en fonction de sa langue (fig.12). La gestion du multilingue devra être activée pour les types de contenu concernés.



Pattern for all Anglais Produit paths

Pattern for all Français Produit paths

Figure 12 - Création des différents format d'alias pour chaque langue.

La création d'alias spécifiques pour les différentes langues fonctionne en combinaison avec le module « Entity Translitération ». Attention toutefois de bien étudier le fonctionnement de chacun des plugins cités dans ce paragraphe avant de vous lancer, la configuration multilingue pouvant être bien plus complexe qu'il n'y paraît.

Module Global redirect

Nous avons vu que par défaut les URL des nœuds avaient la forme [/node/123](#). Si jamais ces URL sont appelées par un navigateur (ou un crawler) et malgré la mise en place d'alias d'URL, les pages répondront avec un status 200, ce qui posera un problème de duplication de contenu.

Avec ce module, les URL par défaut ([/node/123](#)) redirigeront en 301 vers les URL SEO-friendly (alias) quand elles existent (fig. 13).

Non-clean to Clean

If enabled, this option will redirect from non-clean to clean URL (if Clean URL's are enabled). This will stop, for example, node 1 existing on both [example.com/node/1](#) AND [example.com?q=node/1](#).

Figure 13 – Option pour rediriger les URL par défaut vers les URL propres.

Une autre option de ce module vous permettra de rediriger vers la bonne URL si une erreur de casse se produit dans l'URL demandée (pour prévenir d'éventuels liens externes erronés). Ex: [/uk/en/Product-XXXX.html](#) redirigera vers [/uk/en/product-XXXX.html](#)

A noter : le module « Global Redirect » a été intégré au module « Redirect » sous Drupal 8.

Module redirect

Ce module offre la possibilité de créer des redirections directement via le back-office, sans avoir à passer par le fichier .htaccess, ce qui vous donnera plus de souplesse pour la mise en place de redirections.

En cas de modification d'alias

En cas de mise à jour de contenu, si le titre du nœud Drupal a été modifié, un nouvel alias sera généré par Drupal. Une option vous permettra de créer des redirections quand un contenu est mis à jour (ou de ne pas modifier l'alias) pour prévenir d'éventuelles erreurs 404, et de supprimer ou non la chaîne de paramètres si elle est présente en fin d'URL (fig. 14).

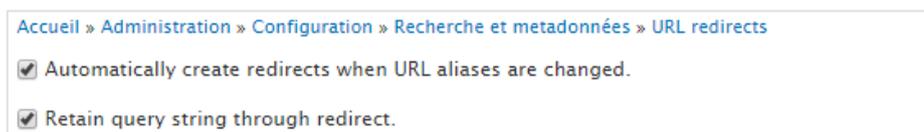


Fig.14. Modification d'alias.

Détection des erreurs 404

Cette option du module vous permettra de tracker les erreurs 404 afin de les corriger rapidement, en ajoutant des redirections quand cela est pertinent (fig. 15) .

PAGE	DÉCOMPTE	LAST ACCESSED	ACTIONS
fr/fr/souscat	8	02/22/2016 - 11:29	Add redirect
taxonomy/term/3	5	02/22/2016 - 11:30	Add redirect

Figure 15 - Analyse et correction des erreurs 404.

Cette fonctionnalité s'avérera très utile pour améliorer votre SEO quand des sites externes mettent en place des liens erronés, pour bénéficier de leur popularité malgré tout.

Module metatag

Ce module permet de configurer les balises SEO (<title>, <meta name=description content="">,..) et d'autres types de balises meta comme celle de l'OpenGraph ou les TwitterCards. Un format spécifique pourra être défini (fig. 16) pour chaque type de page (ex: Node, Views) en utilisant des variables (*title*, *page number*, *views name*, *sitename*), et ce format sera utilisé par défaut pour chaque contenu.



▼ BASIC TAGS

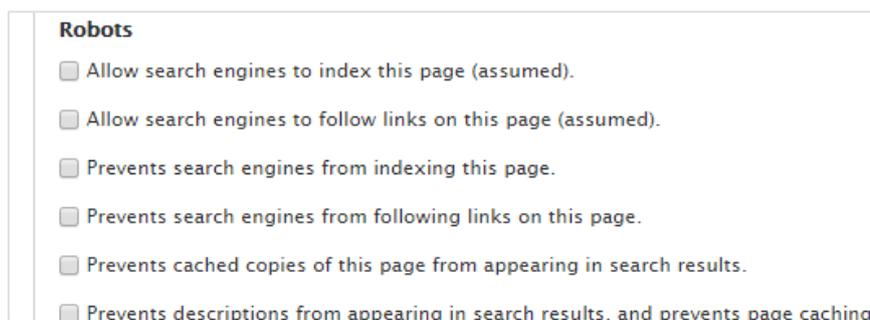
Browse available tokens.

Titre de page
[node.title] | [site.name]
The text to display in the title bar of a visitor's web browser when they view this page. This meta tag may also be used as the title of the page when a visitor bookmarks or favorites this page.

Description
[node.summary]
A brief and concise summary of the page's content, preferably 150 characters or less. The description meta tag may be used by search engines to display a snippet about the page in search results.

Figure 16 - Définition du format de balise title par défaut pour un type de contenu.

Des options spécifiques pour améliorer le temps de chargement de vos pages sont disponibles, afin de maîtriser la durée de validité de vos contenus : *Cache-Control*, *Expires headers*, ainsi que la façon dont les robots d'indexation doivent prendre en compte vos contenus (fig.17) : *index*, *follow*, *nosnippet*, *nocache*, etc.)



Robots

- Allow search engines to index this page (assumed).
- Allow search engines to follow links on this page (assumed).
- Prevents search engines from indexing this page.
- Prevents search engines from following links on this page.
- Prevents cached copies of this page from appearing in search results.
- Prevents descriptions from appearing in search results, and prevents page caching.

Figure 17 - Contenu de la balise meta robots pour chaque type de contenu.

Les réglages pour chaque type de contenu peuvent être écrasés manuellement pour optimiser d'une manière plus fine les titres et meta description de vos contenus et catégories.

Module title

Ce module convertit le champ titre des nœuds avec l'API Field de Drupal, pour qu'ils puissent être traduits en cas de site multilingue (fig. 18). On pourra ainsi créer un champ titre supplémentaire pour chaque langue. A noter que cette fonctionnalité a été intégrée dans le cœur de Drupal 8.

ÉTIQUETTE	POIDS	PARENT	NOM SYSTÈME	TYPE DE CHAMP	WIDGET	ACTIONS
Titre	-5	- Aucun(e) -	title_field	Texte	Champ texte	modifier supprimer
Body	-4	- Aucun(e) -	body	Texte long et résumé	Zone de texte avec un résumé	modifier supprimer
Image	-1	- Aucun(e) -	field_image	Image	Image	modifier supprimer
Langue	5	- Aucun(e) -	language	Language selection		
URL redirects	30	- Aucun(e) -	redirect	Redirect module form elements		
Paramètres des chemins d'URL	30	- Aucun(e) -	path	Path module form elements		
Meta tags	40	- Aucun(e) -	metatags	Meta tag module form elements.		
Titre en	41	- Aucun(e) -	field_titre_en	Texte	Champ texte	modifier supprimer

Figure 18 - conversion du champ Titre par défaut en champ personnalisé.

Le module "title" n'est nécessaire que dans le cas où on utilise "Entity Translation", et ne sera pas utile si vous utilisez le module "Content Translation" pour le multilinguisme.

Module hreflang

Un grand nombre de sites Web servent les utilisateurs du monde entier avec des contenus traduits ou ciblés pour les utilisateurs étant dans une certaine région. Google utilise la balise <link rel = "alternate" hreflang = "x"> pour servir la bonne URL dans les résultats de recherche. Avec plusieurs versions d'une URL, chaque page doit identifier les URL de toutes les versions linguistiques.

```
<head profile="http://www.w3.org/1999/xhtml/vocab">
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
  <link rel="shortcut icon" href="http://drupal.resoneo.org/misc/favicon.ico" type="image/vnd.microsoft.icon" />
  <link href="http://drupal.resoneo.org/en/archive/201602" rel="alternate" hreflang="en" />
  <link href="http://drupal.resoneo.org/fr/archive/201602" rel="alternate" hreflang="fr" />
  <meta name="generator" content="Drupal 7 (http://drupal.org)" />
  <link rel="canonical" href="http://drupal.resoneo.org/fr/archive/201602" />
  <link rel="shortlink" href="http://drupal.resoneo.org/fr/archive/201602" />
```

Figure 19 - Ajout automatique des balises hreflang via un module dédié.

Ce module ajoute l'élément hreflang, dans toutes les versions linguistiques disponibles pour chaque contenu (fig. 19). Parfois, un contenu peut ne pas être pas traduit dans une langue spécifique, mais le hreflang sera malgré tout ajouté à toutes les versions. Une solution existe dans ce cas pour ajouter la balise hreflang uniquement lorsque le nœud n'est pas vide : <https://www.drupal.org/node/246565>

Plus d'informations sur les balises hreflang:

<https://support.google.com/webmasters/answer/189077>

Pagination avec prev et next

Pour éviter la duplication de contenu sur des pages de listing par exemple, et permettre à Google de comprendre qu'une série de plusieurs pages forment un ensemble, il est recommandé d'indiquer les relations entre ces différentes pages d'une même série avec les attributs prev/next associés à l'élément <link>.

Pour l'URL /bags?page=3, nous aurons dans la section <head>:

```
<link rel="prev" href="/bags/?page=2" />
<link rel="next" href="/bags/?page=4" />
```

Actuellement, il n'existe malheureusement pas de module intégrant cette fonctionnalité sous Drupal 7, mais des patches existent (ou des modules en version Alpha, à éviter sur des sites en production) : <https://www.drupal.org/node/1567684> . La gestion du prev/next est maintenant native sous Drupal 8

Module custom breadcrumbs

Ce module permet la construction d'un fil d'Ariane (fig. 20), élément indispensable pour votre maillage interne des pages de catégories trop profondes, mais aussi pour faciliter la navigation remontante vers celles-ci.

Selon les besoins, on utilisera plutôt un module ou l'autre (même si CustomBreadcrumbs est suffisamment générique pour normalement couvrir tous les cas). Attention cependant au thème qui met parfois son grain de sel dans le fil d'Ariane.

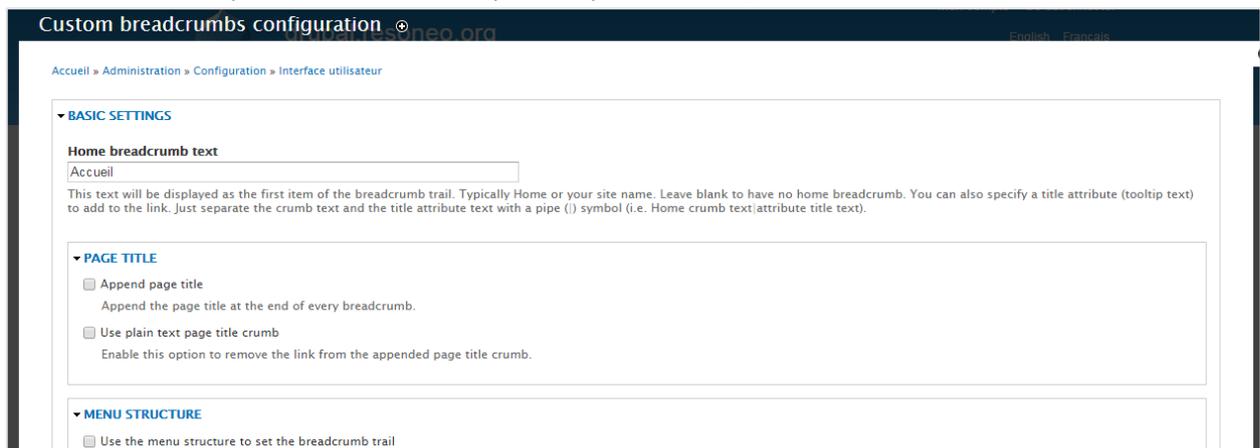


Figure 20 - Configuration du module custom breadcrumb.

Module Transliteration

Ce module nettoie les noms de fichiers lors du téléchargement en remplaçant les caractères indésirables. Il sera notamment très utile pour les images et permettra que les noms de fichiers uploadés soient propres.

Drupal est-il SEO-friendly ?

Comme vous pouvez le constater, la bonne compréhension de l'architecture de Drupal est l'une des clés pour rendre votre site optimal en termes de référencement. Sa structure modulaire en fait une vraie force, mais peut devenir un point faible avec l'utilisation de thèmes « clés en main », ces derniers comportant souvent des lacunes, même les thèmes marqués « SEO friendly ».

L'ajout des modules présentés dans cet article est un complément indispensable pour faciliter la gestion de vos principales balises SEO, mais leur utilisation n'est pas anodine : plus vous installez de modules, plus vous alourdirez votre instance Drupal.

Bien que cela puisse paraître contradictoire, l'installation d'un module supplémentaire de mise en cache serveur comme BOOST (<https://www.drupal.org/project/boost>) vous permettra d'alléger le chargement de vos pages.

Téléchargement / sources

Gestion des URL:

- RobotsTxt : <https://www.drupal.org/project/robotstxt>
- PathAuto : <https://www.drupal.org/project/pathauto>
- Global Redirect : <https://www.drupal.org/project/globalredirect>
- Redirect : <https://www.drupal.org/project/redirect>
- Transliteration module : <https://www.drupal.org/project/transliteration>

Seo on-site :

- Metatag module : <https://www.drupal.org/project/metatag>
- Title module : <https://www.drupal.org/project/title>
- Custom Breadcrumb Module : https://www.drupal.org/project/custom_breadcrumbs

Aider Google à comprendre la structure du site :

- Hreflang module : <https://www.drupal.org/project/hreflang>
- Link prev & next infos : <https://www.drupal.org/node/1567684>
- Sitemap module : <https://www.drupal.org/project/xmlsitemap>



Aymeric Bouillat, consultant SEO, Resoneo (<http://twitter.com/aymerictwit> ou <http://www.yapasdequoi.com>)