

# Création de sites Web modernes grâce à Drupal

Uwe Hermann

**Si vous avez besoin d'un système de gestion de contenus multi-tâche, capable de gérer plusieurs sites indépendants, traduits en différentes langues, et ce en une seule installation, si vous souhaitez ajouter du dynamisme à votre site via AJAX, par exemple pour optimiser un moteur de recherche et ainsi augmenter la fréquentation de votre site, et si vous souhaitez que l'ensemble du système soit facile à installer, à utiliser et à mettre à jour, ne cherchez plus : nous allons vous présenter Drupal.**

**D**rupal est un gestionnaire de contenus open source modulaire, flexible et adaptable, écrit en PHP, et supporté par une base de données relationnelle (MySQL ou PostgreSQL). Le noyau de Drupal est distribué sous les termes de GNU General Public License, ainsi que la plupart des modules et thèmes élaborés pour les utilisateurs. Bien que Drupal se révèle très puissant en terme de fonctionnalités, l'outil demeure suffisamment flexible pour être utilisé sur pratiquement tous les types de sites Web : portails, sites Web personnels, sites Web d'entreprises, forums de communautés, journal en ligne, photoblogs, émissions balados, etc. Tout est réalisable avec Drupal.

Nous allons vous expliquer, dans le cadre du présent article, comment utiliser Drupal afin de réaliser un portail complexe, en soulignant les fonctionnalités les plus sophistiquées de ce système, dont, entre autres, l'installation multi-sites, la création de sites Web internationaux, la

classification ou le balisage du contenu, l'optimisation du moteur de recherche, et les interfaces utilisateurs AJAX. Nous évoquerons également la version à venir, Drupal 4.7, censée comprendre un certain nombre d'améliorations intéressantes par rapport à la version stable actuelle.

## Concept de notre portail

Nous allons découvrir certaines fonctionnalités de Drupal en réalisant un échan-



### Auteur

**Uwe Hermann** est étudiant en informatique à la Technische Universität München, Allemagne, mais travaille également en tant que programmeur indépendant et contribue à de multiples projets open source tels que Debian GNU/Linux ainsi que Drupal, entre autres. Il est à l'origine, et maintient toujours le projet intitulé Unmaintained Free Software (<http://unmaintained-free-software.org>), dont l'objectif est de trouver de nouveaux mainteneurs pour des projets open sources qui en sont dépourvus. Vous pouvez contacter l'auteur à l'adresse suivante : [uwe@hermann-uwe.de](mailto:uwe@hermann-uwe.de).

### Ce qu'il faut savoir...

Vous maîtrisez idéalement les bases d'Apache, de PHP, de MySQL, et des systèmes de gestion de contenus en général.

### Cet article explique...

Nous allons vous montrer comment créer un site Web international grâce à Drupal, et à ses fonctionnalités parmi les plus sophistiquées, comme l'installation multi-sites, l'internationalisation, les interfaces utilisateurs AJAX et l'optimisation du moteur de recherche.

## Origine du nom Drupal

Le projet Drupal a été lancé par Dries Buytaert en l'an 2000. Le nom Drupal tire son origine du mot allemand *drappel* qui signifie *goutte*. Dries Buytaert a choisi ce nom parce qu'il avait, par mégarde, tapé *dorp.org* (*dorp* signifie en allemand *village*) à la place de *drop.org* lorsqu'il cherchait un domaine pour le site Web de la communauté pour lequel il prévoyait d'utiliser Drupal. Le nom *drop* a fait mouche, et c'est ainsi que le nom Drupal est resté.

illon de portail communautaire destiné aux personnes désireuses de télécharger légalement et gratuitement de la musique sur Internet. Ce site aura pour vocation d'attirer les fans et les artistes indépendants, en leur proposant des forums de discussions, des blogs, des votes hebdomadaires, et plus encore. Nous prévoyons également de créer un site supplémentaire organisé par genre de musique sur notre portail, sur lequel nous appliquerons la fonctionnalité multi-sites proposée par Drupal. Chacun de nos sites sera doté d'un sous-domaine dans notre domaine commun intitulé *example.com*. Le site principal sera consultable à partir de l'adresse *http://www.example.com*. Il sera doté de sous-domaines indépendants pour les genres de musique, comme *http://rock.example.com*, *http://blues.example.com*, ou *http://electronica.example.com* par exemple. Chaque site organisé autour d'un genre de musique disposera de réglages, de thèmes, de blocs, de contenus, de forums, etc. qui leur seront propres. Toutefois, nous souhaitons que les utilisateurs n'aient à créer qu'un seul compte utilisateur par personne, et soient capables de se connecter

### Listing 1. Définitions Apache de nos sous-domaines

```
<VirtualHost *:80>
  DocumentRoot /var/www
  ServerName www.example.com
</VirtualHost>
<VirtualHost *:80>
  DocumentRoot /var/www
  ServerName rock.example.com
</VirtualHost>
<VirtualHost *:80>
  DocumentRoot /var/www
  ServerName blues.example.com
</VirtualHost>
```

à l'ensemble de nos sites, autrement dit une ouverture de session unique. Ainsi, tous les sites partageront l'ensemble des données de nos utilisateurs.

## Installation multi-sites

L'installation multi-sites permet de créer plusieurs sites Web, tous situés sur des domaines indépendants, des sous-domaines ou des sous-répertoires d'un seul domaine. Tous les sites peuvent disposer de réglages, de modules, de thèmes et d'autres éléments différents, tout en ayant recours à la même installation Dupral. Par ailleurs, les sites peuvent avoir des tables de bases de données complètement indépendantes, ou partager *certaines* ou *l'ensemble* des tables, selon les besoins.

## Fonctionnement de l'installation multi-sites

Le mécanisme élémentaire de l'installation multi-sites consiste à héberger plusieurs sites alimentés par Dupral dans l'ordre de recherche que Dupral applique pour trouver les fichiers de configuration, les modules, et ainsi de suite. En règle générale, le fichier de configuration intitulé *settings.php* est situé dans *sites/default/settings.php*. Dans le cadre d'un scénario multi-sites, il faut créer un sous-répertoire *sites* pour chaque site, et placer le fichier *settings.php* dans ce répertoire. Les noms de ces sous-répertoires doivent comprendre le nom du sous-domaine en question, le nom du domaine, le numéro du port, uniquement nécessaire lorsque le serveur Web n'écoute par le port

## Installation

Les instructions d'installation suivantes concernent la version actuelle CVS HEAD de Drupal. Il faudra donc remplacer *cv*s par *4.7* dès que Drupal, version 4.7 sortira.

- Télécharger et extraire le tarball Dupral vers le répertoire */var/www/testsite* :

```
mkdir /var/www/testsite
wget http://drupal.org/files/projects/drupal-cvs.tar.gz
tar xfvz drupal-cvs.tar.gz
mv drupal-cvs/* drupal-cvs/.htaccess /var/www/testsite
```

- Créer la base de donnée Drupal puis charger le schéma de la base de donnée Dupral :

```
mysqladmin -u nom d'utilisateur de la base de données -p create ←
  dbtestsite
mysql -u nom d'utilisateur de la base de données -p dbtestsite ←
  < database/database.mysql
```

- Ajuster les variables *\$db\_url* et *\$base\_url* dans *sites/default/settings.php* :

```
$db_url = "mysql://dbuser:dbpassword@dbhost/dbtestsite";
$base_url = "http://www.example.com/testsite";
```

- C'est fini. Vous pouvez désormais accéder à notre site à l'adresse : *http://www.example.com/testsite*,

Etapes optionnelles, mais recommandées :

- Créer un répertoire *files/* modifiable pour le serveur Web. C'est dans ce répertoire que Drupal va stocker les images et autres fichiers importés :

```
mkdir /var/www/testsite/files
chmod 777 /var/www/testsite/files
```

- Créer une tâche cron exécutant régulièrement le fichier *cron.php*. À titre d'exemple, nous pourrions ajouter la ligne suivante au fichier */etc/crontab* afin d'exiger *cron.php* toutes les heures :

```
0 * * * * wget -O - -q http://www.example.com/testsite/cron.php
```

## Exigences

Les exigences minimum en terme d'installation de Drupal sont les suivantes : PHP (depuis la version 4.3.3 ou PHP5), base de données relationnelle (MySQL ou PostgreSQL), et serveur HTTP (Apache ou IIS).

Exigences optionnelles mais recommandées : extension XML de PHP, support *mod\_rewrite*, et support des fichiers *.htaccess*.

## Terminologie Drupal

- *Node* – contenu actuel de notre site stocké dans des *nodes* ; il existe plusieurs *node types* comme *page*, *blog entry*, *poll*, ou *event*.
- *Block* – petit échantillon de contenu généralement placé à gauche ou à droite de notre site Web. À titre d'exemple, citons *Who's online* ou *Recent Comments*.
- *Module* – extension Drupal (écrite en PHP) chargée d'ajouter certaines fonctionnalités à notre site.
- *Taxonomy* – permet d'organiser le contenu de notre site en catégorie ou en balises.
- *Theme* – modifie les fonctions de thèmes de Drupal, proposant de la sorte un aspect unique sur notre site.
- *Theme engine* – chaque thème exige un certain *theme engine*. Le moteur par défaut est PHPTemplate, mais il existe d'autres moteurs comme XTemplate ou Smarty.

80, et le(s) nom(s) du sous-répertoire du site respectif. Toutes les parties composant le nom du répertoire sont séparées par des points, et les deux points utilisés pour indiquer le port du serveur Web doivent également être remplacés par un point.

Dans le cas de notre site portail, la structure du répertoire et des fichiers pour les sites de `www.example.com`, `blues.example.com` et `rock.example.com` devrait être la suivante :

- `sites/example.com/settings.php`
- `sites/rock.example.com/settings.php`
- `sites/blues.example.com.subdir/settings.php`

Veuillez noter qu'il est fortement recommandé d'utiliser `example.com` au lieu de `www.example.com` en tant que nom de répertoire, si le site est censé être consultable à partir des deux adresses.

### Création des sous-domaines

Nous devons désormais créer les sous-domaines mentionnés plus haut. Pour ce faire, il faut donc modifier les réglages de notre serveur Web afin de créer des hôtes virtuels. Dans la mesure où nous avons recours à Apache, et comme la racine de notre document est `/var/www`, il faudra ajouter les lignes présentées dans le Listing 1 à notre fichier `/etc/apache/httpd.conf`.

Cette manœuvre permet de faire pointer les sous-domaines `www`, `rock`, et `blues` vers la racine de notre document. Nous installerons dans ce répertoire l'unique code Drupal chargé d'alimenter l'ensemble de nos sites. Un tel scénario multi-sites utilisant des sous-domaines est également censé fonctionner sur la plupart des comptes Web hébergés et partagés.

Toutefois, votre fournisseur doit vous régler les sous-domaines ou vous pro-

poser une application Web frontale, censée effectuer cette tâche, dans la mesure où vous n'aurez généralement pas directement accès au fichier `httpd.conf`. S'il est impossible de réaliser cette tâche, vous pouvez toujours avoir recours aux sous-répertoires à la place des sous-domaines afin de créer une application multi-sites.

### Création des tables de base de données

Il faut ensuite créer les tables de base de données pour chacun de nos sites. Plusieurs possibilités sont envisageables dans ce cas. Si vous souhaitez ne partager *aucune* table entre les sites, autrement dit ces derniers seront complètement indépendants, il suffit de créer une seule base de données par site :

```
mysql -u dbuser -p database1 ←
< database/database_1.mysql
mysql -u dbuser -p database2 ←
< database/database_2.mysql
```

À l'inverse, si vous souhaitez partager *toutes* les tables de base de données, tous les sites deviendront alors identiques et partageront les mêmes réglages, thèmes, contenus, etc. Dans ce cas, il suffit de créer une seule base de données pour l'ensemble des sites :

```
mysql -u nom d'utilisateur de la ←
base de données -p db_for_all ←
< database/database.mysql
```

Néanmoins, nous profiterons des capacités de Drupal sur notre portail de musique, afin de travailler avec les *préfixes de table*. Nous stockerons les tables dans une seule base de données pour l'ensemble de nos sites, mais nous préfixerons les noms des tables, afin de savoir quelle

table correspond à quel site. Par souci de clarté, nous attribuerons aux tables de notre site principal le préfixe `shared`, puisque certains éléments du site principal seront partagés avec d'autres sites. Le site `rock.example.com` prendra le préfixe `rock` et le site `blues.example.com` prendra celui de `blues`. Pour ce faire, nous utiliserons pour commencer la procédure `prefix.sh` à partir du répertoire `scripts/` afin de préfixer les noms de tables de la base de donnée dans le fichier `database/database.mysql`, avant de charger les tables dans MySQL :

```
./prefix.sh "shared_" database.mysql ←
> database.mysql.shared
./prefix.sh "rock_" database.mysql ←
> database.mysql.rock
```

et :

```
./prefix.sh "blues_" database.mysql ←
> database.mysql.blues
```

### Contenus des fichiers settings.php

Les réglages les plus importants dans `sites/www.example.com/settings.php` sont les suivants :

```
$base_url="http://www.example.com";
$db_url="mysql://utilisateur:
  mot_de_passe@hôte/nom_de_la_base";
$db_prefix = "shared_";
```

Alors que pour `sites/rock.example.com/settings.php`, nous aurons les éléments suivants :

```
$base_url="http://rock.example.com";
$db_url= ...;
$db_prefix = array(
  'default' =>'rock_',
  'users' =>'shared_',
  'users_roles' =>'shared_',
  'sessions' =>'shared_',
  'role' =>'shared_',
  'authmap' =>'shared_',
  'sequences' =>'shared_',
  'profile_fields' =>'shared_',
  'profile_values' =>'shared_',
);
```

Pour `sites/blues.example.com/settings.php`, nous devons régler comme suit :

```
$base_url=
  "http://blues.example.com";
$db_url = ...;
$db_prefix = array(
  'default' => 'blues_'
  ...
);
```

Cherchez chez votre marchand de journaux

**hakin9** **+CD** SUR LE CD : hakin9.live plein d'outils de la sécurité  
GFI LANguard Network Security Scanner – un des scanners de sécurité les plus connus au monde ; version complète (pour 5 adresses IP) ; CORE IMPACT – RAPID PENETRATION TEST (DÉMO)

**hakin9**  
comment se défendre

Hard Core IT Security Magazine hakin9 Prix : 7.50 EUR N°1/2006 (14) Janvier/Février Bimestriel CD Offert  
ISSN : 1731-7037

**Hacking du Wi-Fi**  
Les réseaux sans fil toujours vulnérables

**LIVE TRAINING CENTER**  
bootez  
testez  
comprenez

**Rootkits sous Oracle**  
Se camoufler dans une base de données

**Pirater Microsoft**  
Windows Server 2003

**Hélas, vous n'êtes pas en sécurité**  
Tromper les pare-feux

**POUR LES DÉBUTANTS**

**IPS basé sur Snort**  
Votre propre système de protection libre

**Chasse aux spyware**  
Comment les détecter et les supprimer

**SUR LE CD**  
GFI LANguard Network Security Scanner  
**+11 tutoriaux**  
dont un nouveau – ips  
NOUVEAUX E-BOOKS : Firewall Piercing, Database Rootkits, Windows Server 2003 Security Guide

Hacking du Wi-Fi • Rootkits sous Oracle • Détournement de pare-feux • Spyware • GFI LANguard Network Security Scanner

shop.software.com.pl/fr

www.hakin9.org

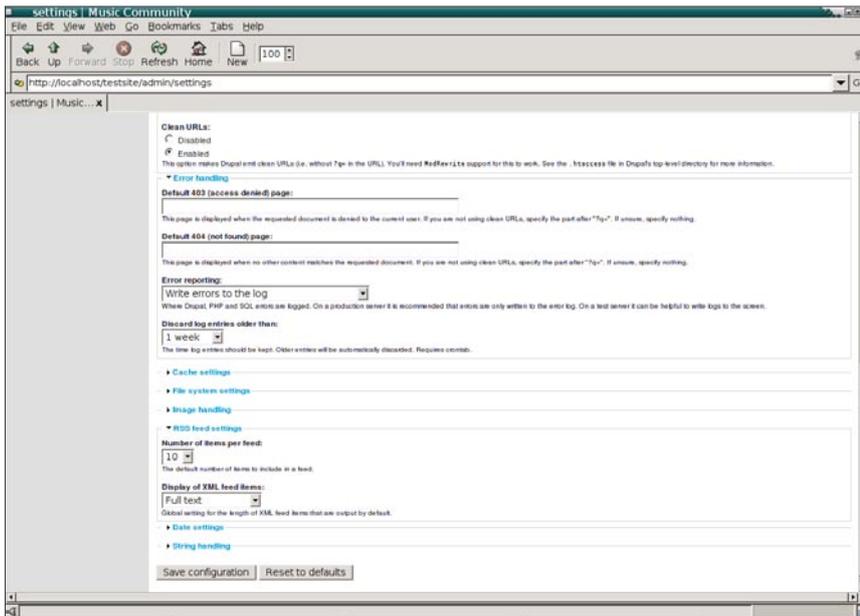


Figure 1. La page de réglage de Drupal permet de personnaliser divers aspects de notre site

En langage SQL, le fichier `database.mysql.shared` contient désormais des lignes telles que `CREATE TABLE shared_users` au lieu des anciennes lignes de type `CREATE TABLE users`, et le fichier `database.mysql.rock` contient désormais la ligne `CREATE TABLE rock_users`, et ainsi de suite. Chaque fichier est désormais doté de son préfixe correspondant, appliqué aux noms des tables. Nous pouvons maintenant charger les fichiers dans la base de données MySQL :

```
mysql -u dbuser -p dbname <
  < database.mysql.shared
mysql -u dbuser -p dbname <
  < database.mysql.rock
```

et :

```
mysql -u dbuser -p dbname <
  < database.mysql.blues
```

## Configuration des sites

La dernière étape de notre installation multi-sites consiste à éditer chaque fichier `settings.php` afin de le personnaliser pour son site respectif. En d'autres termes, il faut au moins entrer les valeurs propres pour les variables `$base_url`, `$db_url` et `$db_prefix`. La variable `$base_url` ne contient que l'URL de niveau supérieur du site correspondant, soit `http://rock.example.com`, alors que la variable `$db_url` stocke les réglages de la base de données du site.

Dans le cas de notre portail, la variable `$db_url` est identique dans tous les

fichiers `settings.php`, dans la mesure où nous utilisons la même base de données pour l'ensemble des sites. La variable `$db_prefix` contient un tableau associatif chargé de faire correspondre les noms de tables aux préfixes de tables (voir la partie intitulée *Les contenus des fichiers settings.php*). Dans le cas d'une création sans aucun partage entre tables, la variable `$db_prefix` ne serait qu'une simple chaîne, c'est-à-dire `prefix_` ou "" (chaîne vide), dans le cas où vous ne voulez pas utiliser de préfixe. N'oubliez pas que nous voulons installer une ouverture de session unique sur les sites de notre portail. Il

nous faut donc partager certaines tables. Savoir *quelles* tables partager n'est pas aisé, et exige une bonne connaissance du fonctionnement interne de Drupal. Le procédé est extrêmement dépendant de ce que vous voulez créer. Dans le cadre de notre portail, nous allons partager les tables de la base de données suivantes : `users`, `users_roles`, `sessions`, `role`, `authmap`, `sequences`, `profile_fields`, et `profile_values`.

En plus des variables `$base_url`, `$db_url` et `$db_prefix`, nous pouvons également figer dans le code certains réglages par site dans le fichier `settings.php` correspondant. Il est plus particulièrement possible de modifier n'importe quelle variable de configuration de Drupal (ces dernières sont généralement stockées dans la table de base de données intitulée `variable`) au moyen de la syntaxe suivante :

```
$conf = array(
  'site_name'=>'Site de discussions de
    musique Rock',
  'theme_default'=>'pushbutton',
  'comment_preview'=>'1'
);
```

Ainsi, il est possible de modifier le nom du site et le thème par défaut du site, et de régler la fonctionnalité de commentaire de visualisation sur `required`. En plus des fichiers `settings.php` pour chaque site, nous pouvons également régler les répertoires `modules/` et `themes/` par site, de sorte que certains modules et thèmes soient spécifiés.

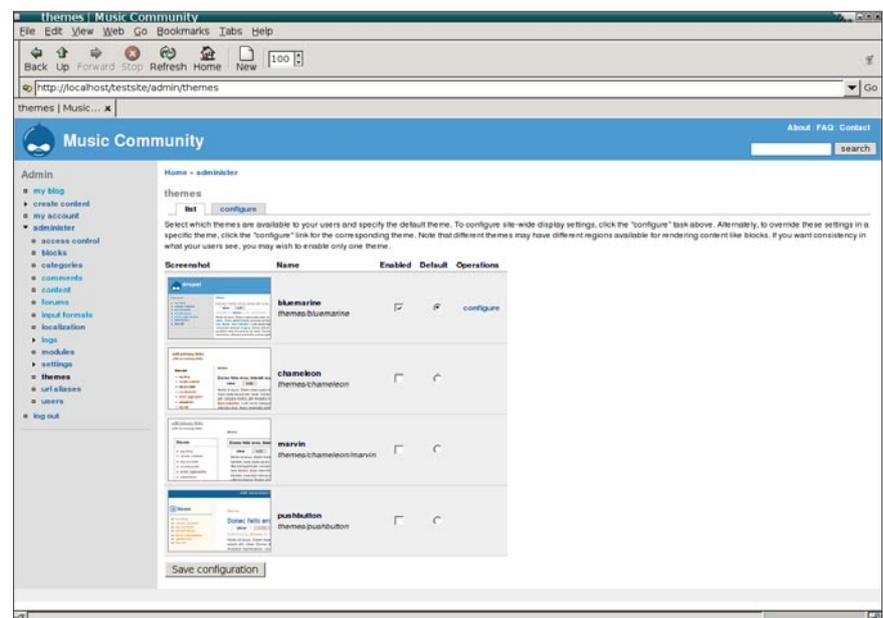


Figure 2. La page de réglage themes permet d'activer un ou plusieurs thèmes, et d'en choisir un en tant que thème du site par défaut

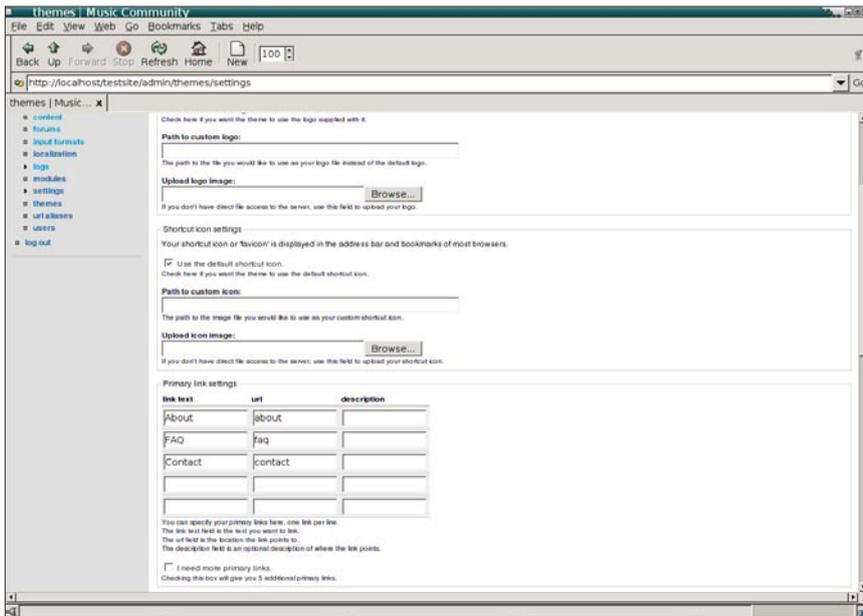


Figure 3. Nous utilisons primary links pour définir le menu du site en haut à droite

ques à certains sites. Ces répertoires sont tout simplement placés dans le répertoire de configuration du site correspondant. Par exemple :

```
sites/rock.example.com/settings.php
sites/rock.example.com/modules
sites/rock.example.com/themes
```

L'installation multi-sites étant désormais achevée, nous pouvons débuter la configuration de notre portail de musique.

## Configuration élémentaire

Il est donc temps de créer notre premier utilisateur. Nous allons donc visiter le site <http://www.example.com> puis cliquer sur *Create new account* dans le bloc intitulé *User login* situé à gauche. Le premier utilisateur que nous allons créer (quelque soit le nom choisi) sera l'administrateur autorisé à faire tout ce qu'il veut sur le site. Une fois le compte ouvert en tant qu'administrateur, nous pouvons ajuster quelques réglages élémentaires comme le nom du site (*Music Portal*) dans *administer* → *settings*. Nous allons également régler le paramètre *Error reporting* sur *Write errors to the log*, afin d'éviter que les erreurs internes ne deviennent visibles pour nos utilisateurs.

Nous allons également activer la fonctionnalité intitulée *Clean URLs* pour une meilleure convivialité, ainsi que le mécanisme de cache intégré pour une performance améliorée. Il ne faut surtout pas oublier que dans la mesure où nous ne partageons que

la table de base de données *users* et quelques tables qui lui sont liées, ces réglages ne seront pas partagés entre nos différents sites. Vous pouvez observer les réglages réalisés dans la Figure 1.

## Thèmes

Drupal propose un certain nombre de thèmes intéressants. Vous pouvez en obtenir plus à l'adresse <http://drupal.org/project/Themes>. La procédure d'installation est relativement simple : télécharger le bon fichier à partir de *drupal.org* puis extraire le fichier et le placer dans le répertoire *themes/*. Nous utiliserons le thème choisi par défaut

pour notre portail principal, mais nous téléchargerons et installerons d'autres thèmes pour les sous-sites organisés par genre de musique. Par exemple, nous utiliserons le thème intitulé *FriendsElectric* pour le site *rock.example.com* en l'installant dans le répertoire `sites/rock.example.com/themes/friendselectric`. Nous choisirons ensuite un thème dans le menu *administer* → *themes* (voir la Figure 2).

Il est également possible de télécharger un logo de site personnalisé ainsi qu'une icône de favori pour notre site dans *administer* → *themes* → *configure*. Nous utiliserons la liste des *primary links* afin d'afficher les liens *About*, *FAQ* et *Contact* (voir la Figure 3) en donnant leur nom et l'URL relative, bien qu'il faille toujours créer les pages ! Enfin, nous souhaitons afficher les informations *posted by user <username> on <date>* dans les envois de blogs ou de commentaires sans qu'elles n'apparaissent sur les pages statiques telles que la page intitulée *About*. Nous allons donc désactiver *Display post information on* pour le type de page *nœud*.

## Modules

Nous allons maintenant étendre la fonctionnalité de notre portail en activant quelques modules Drupal intégrés supplémentaires sur la page *administer* → *modules* (voir la Figure 4) :

- *profile* – permet de définir les champs qu'un profil utilisateur est censé

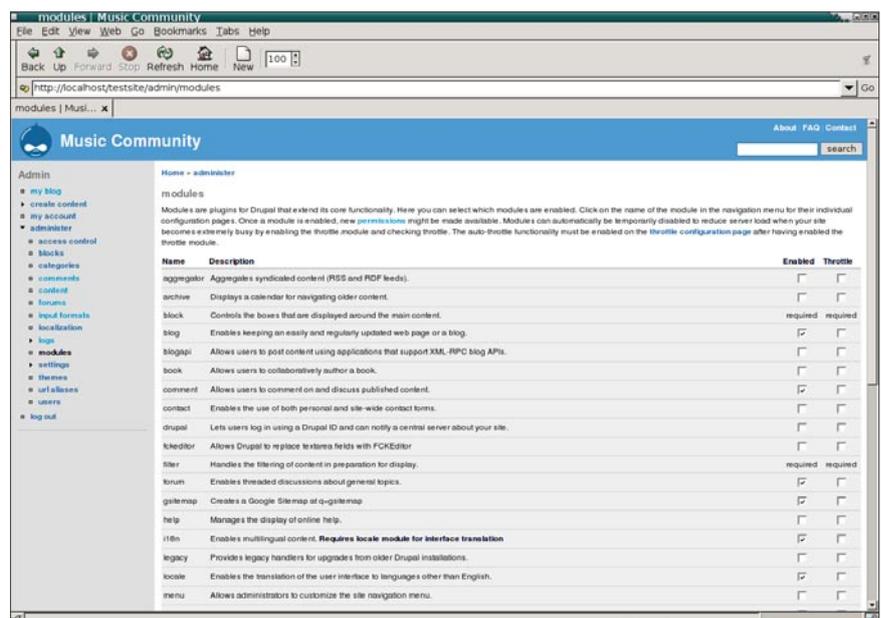


Figure 4. La page de présentation des modules de Drupal affiche tous les modules installés

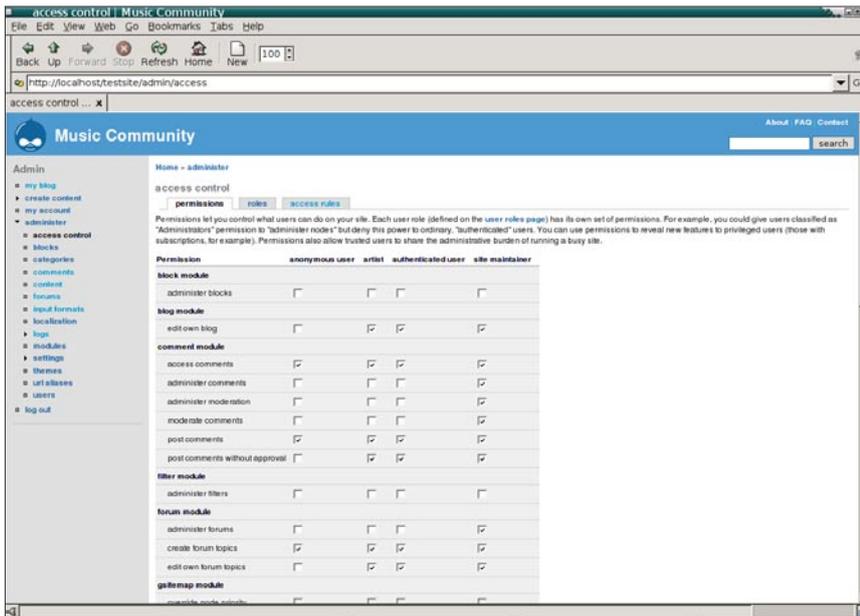


Figure 5. Drupal permet d'implémenter un système de permissions détaillées sur notre site

contenir, comme par exemple, *name*, *birthday*, *email*, *favorite bands*, etc.

- *blog* – permet à chaque utilisateur enregistré de maintenir un blog. L'URL du blog de l'utilisateur 23, par exemple, sur notre site sera la suivante : <http://www.example.com/blog/23>. Par ailleurs, nous obtiendrons automatiquement un *group-blog* (dans lequel les envois de tous les utilisateurs de notre site seront agrégés) en dehors de la boîte : <http://www.example.com/blog>. Chaque blog dispose évidemment de son propre fil d'information en continue, comme par exemple <http://www.example.com/blog/23/feed>. En activant les blogs, nous proposons à nos artistes une plateforme pour parler de leur musique, alors que les fans disposent des moyens nécessaires pour faire partager leurs impressions sur le travail des artistes.
- *ping* – mentionne automatiquement des sites comme *technorati.com* ou *pingomatic.com* lorsque le contenu de notre site est modifié, élément très important et pratique à l'attention des auteurs de blogs.
- *Upload* – permet aux utilisateurs de télécharger des fichiers vers notre site (images ou fichiers MP3, par exemple).
- *forum* – permet de créer un ou plusieurs forums de discussions sur le site.
- *poll* – permet de créer des votes hebdomadaires (questions à choix

multiples) sur lesquels nos utilisateurs peuvent s'exprimer.

- *search* – active le moteur de recherche intégré de Drupal, pouvant être utilisé pour chercher certains contenus ou utilisateurs sur notre site.

Un grand nombre de modules supplémentaires destinés aux utilisateurs est disponible à partir de l'adresse suivante : <http://drupal.org/project/Module>. Nous utiliserons les suivants sur notre portail :

- *image* – permet aux utilisateurs d'importer des images, de les organiser

en galeries, ou de les inclure dans des envois de blogs. Permet de créer automatiquement des vignettes de chaque image.

- *audio* – permet aux utilisateurs d'importer et de gérer des fichiers MP3. Peut également être utilisé par des artistes pour importer leur musique sur le site ou pour une diffusion en ligne. Ce module est proposé avec un lecteur Flash intégré pouvant jouer des morceaux MP3 dans un navigateur Web.
- *trackback* – permet aux auteurs de blogs de notre site d'envoyer et de recevoir des trackbacks.
- *spam* – collection d'outils puissants capables de traiter automatiquement les spam de commentaires et les spam de trackback sur notre site.

## Utilisateurs, rôles et permissions

L'étape suivante consiste à définir certains rôles dans *administer* → *access control* et de leur assigner les permissions souhaitées (voir la Figure 5). Nous allons créer, pour l'instant, les rôles *artist* qui obtiendront des permissions supplémentaires pour importer des fichiers, et *site maintenir* pour les utilisateurs en qui nous avons confiance et qui nous aident à maintenir le site, ce rôle obtiendra également des permissions supplémentaires afin de gérer les forums, etc... Nous utiliserons le rôle prédéfini *authenticated user* pour les utilisateurs connectés et le rôle *anonymous user* pour les utilisateurs

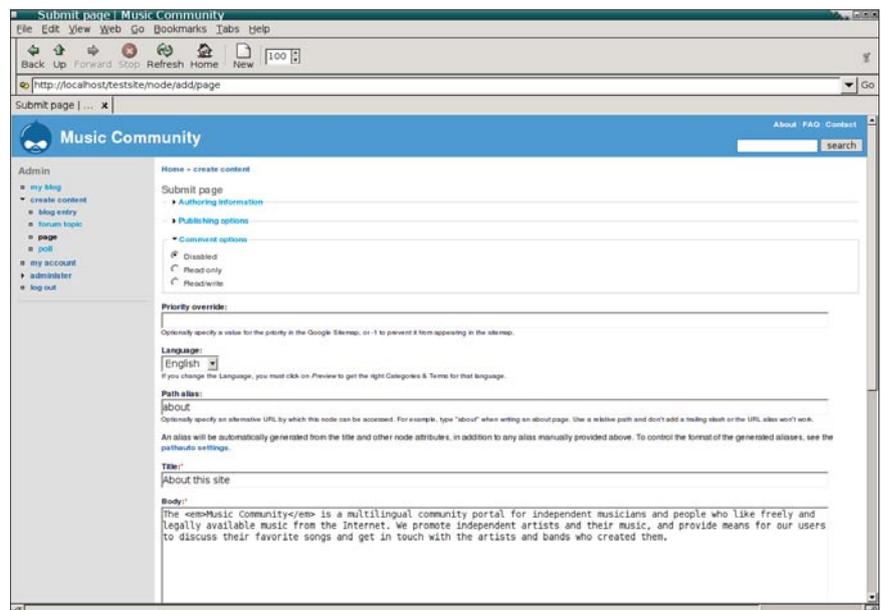


Figure 6. Création d'une page au contenu simple

Figure 7. Champ de formulaire écrasable

non-connectés ainsi que pour l'ensemble des utilisateurs non connus. Enfin, il nous faut assigner les rôles adaptés à tous nos utilisateurs dans *administer* → *users* en cliquant sur le lien *edit* de l'utilisateur choisi. Les dernières permissions de l'utilisateur sont les permissions ajoutées de chaque rôle dans lequel l'utilisateur en question est assigné.

## Création de contenu

Il est désormais possible de créer le premier contenu de notre site Web. Il s'agira de la page statique intitulée *About*, dans laquelle est expliqué le but de notre portail musical. Drupal propose plusieurs *node types* pouvant être utilisés dans ce cas (comme par exemple *page*, *blog entry*, ou *forum topic*). Nous utiliserons *page*. En cliquant sur *create content* → *page*, un champ de formulaire de saisie simple s'affiche, et nous pouvons y entrer le titre du nœud ainsi que son contenu (voir

Figure 8. Choix automatique du nom d'utilisateur AJAX

la Figure 6). Une fois le chemin du module activé, nous pouvons également configurer un chemin pour la nouvelle page. Nous utiliserons dans ce cas *about*, de sorte que notre page se trouve dans *http://www.example.com/about*. Plusieurs réglages sont disponibles pour configurer cette nouvelle page : nous allons désactiver les commentaires pour ce nœud, nous ne publierons pas le nœud sur la page frontale, etc. Il est également possible de créer une nouvelle vision du nœud en activant les boîtes de contrôle correspondantes. Cette manipulation permettra de sauvegarder l'ancien nœud ainsi que le nouveau. Nous pourrions donc retourner vers l'ancienne vision plus tard, ou même comparer les deux.

## Améliorations AJAX et JavaScript

*Asynchronous JavaScript and XML* (AJAX) est une méthode permettant de créer des sites Web interactifs, de plus en plus populaire. Vous pouvez consulter l'article *Améliorer vos applications avec AJAX* dans ce même magazine pour plus d'informations sur AJAX. Drupal 4.7 sera proposé avec une modeste bibliothèque AJAX de base intégrée, déjà utilisée dans certaines zones, afin d'améliorer la convivialité des sites élaborés avec Drupal. Cette bibliothèque propose une interface de programmation pour les modules créés, pouvant facilement faire appel à la fonctionnalité proposée AJAX / JavaScript à des fins personnelles.

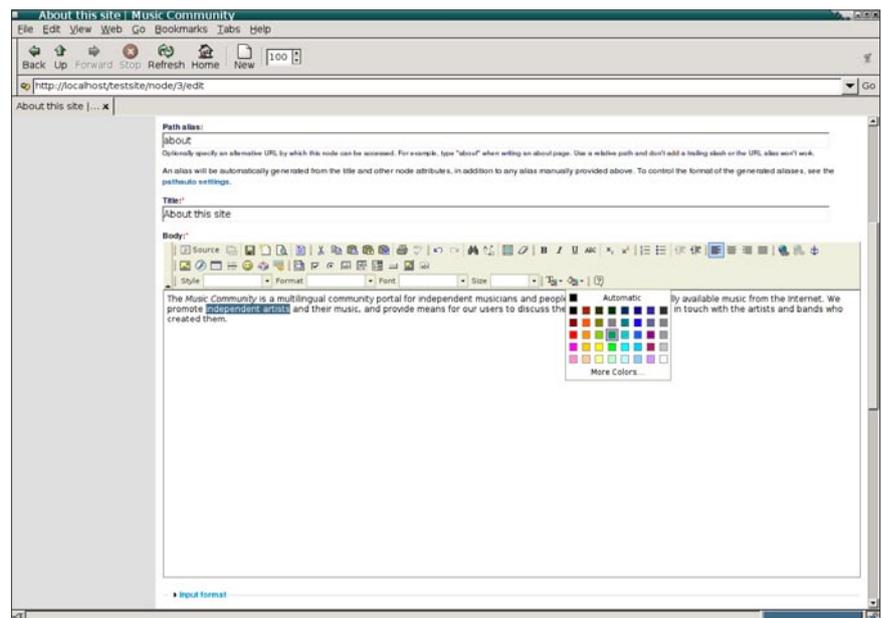


Figure 10. Le module FCKeditor remplace les formulaires d'entrée standard par une version graphique plus conviviale

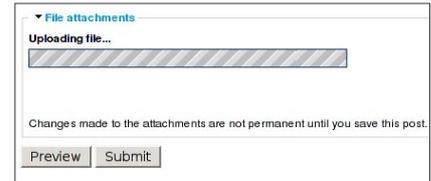


Figure 9. Barre de progrès de téléchargement basée sur JavaScript

*Collapsible form field* est l'une des fonctionnalités AJAX / JavaScript parmi les plus intéressantes. Elle permet, en effet, d'améliorer considérablement la convivialité des pages de formulaire, comme la page *create content* → *blog entry*. Certaines champs de formulaire rarement utilisés sur cette page sont écrasés par défaut (seul le titre est affiché, pas le champ de formulaire entier), de telle sorte que toute la page devienne plus courte et plus précise. Un clic sur le champ de formulaire écrasé permettra alors de l'étendre au moyen de JavaScript (voir la Figure 7). Le navigateur ne demandera pas d'autre page à partir du serveur Web pour réaliser la manipulation. *collapsible form fields* pourrait se révéler très utile sur notre portail, plus particulièrement pour les formulaires d'envoi d'emails à un artiste ou à sa page, ou bien pour ne pas dévoiler certaines informations détaillées de sa biographie.

*Content autocompletion* représente également une autre fonctionnalité AJAX très utile. Elle n'est actuellement disponible que pour les champs et les balises de nom d'utilisateur (catégories). Ainsi, par exemple, si nous voulons assigner

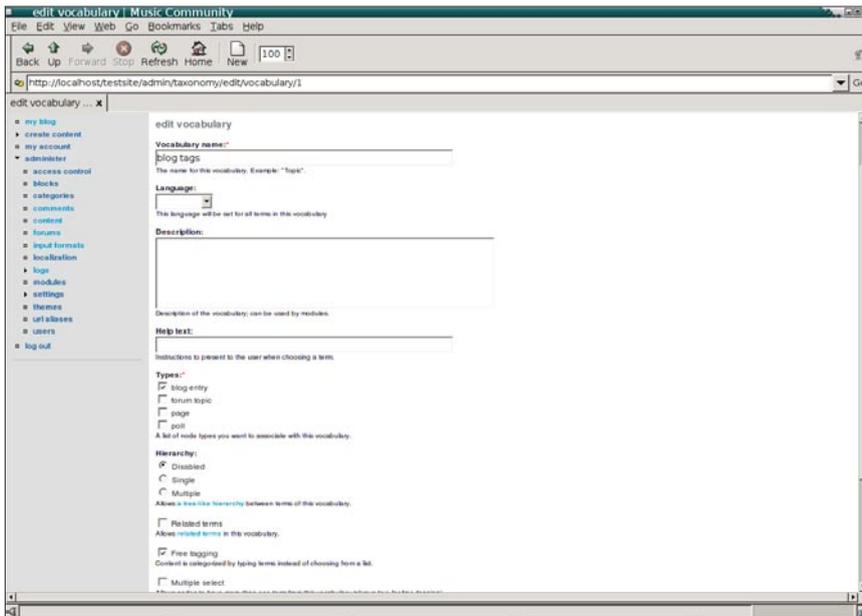


Figure 11. Système de taxinomie permettant d'utiliser un large choix de types et de configurations de vocabulaires différents

de nouveau le propriétaire ou créateur d'un nœud sur notre site, nous devons modifier le champ *Authored by* du nœud correspondant. Par exemple, si nous entrons `li` dans le champ *Authored by*, Drupal affichera automatiquement tous les noms d'utilisateurs contenant la chaîne `li` comme par exemple, *Antonio Salieri*, *Franz Liszt* ou *Georg Philipp Telemann*. Nous pouvons alors facilement choisir l'un de ces noms en cliquant sur l'un d'eux. Le même mécanisme est supporté pour les balises (voir la Figure 8).

Enfin, l'utilitaire d'importation de fichiers sur notre site est amélioré grâce à *JavaScript progress bar*, chargé d'afficher, pour l'utilisateur, le pourcentage du fichier qui a été envoyé (voir la Figure 9).

## Édition WYSIWYG

Afin de faciliter la mise en page du texte sur notre site Web, Drupal propose un environnement graphique simplifié de type WYSIWYG (*What You See Is What You Get*). Il est possible de choisir parmi plusieurs modules destinés aux utilisateurs pour réaliser cette tâche, comme par exemple HTMLarea, TinyMCE, et FCKeditor. Nous porterons notre choix, pour notre portail, sur FCKeditor, basé sur l'éditeur open source du même nom, disponible à partir de l'adresse suivante : <http://www.fckeditor.net>. Cet éditeur offre aux utilisateurs de notre site une incroyable variété de fonctionnalités en matière de mise en page de texte (voir la Figure 10).

## Catégories et balises

Le *taxonomy system* de Drupal est un utilitaire très puissant pour l'organisation du contenu de notre site en catégories. Nous pouvons définir un nombre indéfini de mots dans *administer* → *categories* (comme par exemple *music genre*), chacun contenant les *taxonomy terms* ou *categories* (comme par exemple *rock*, *pop*, ou *trance*). Chaque catégorie dispose d'une page supplémentaire (par exemple, <http://www.example.com/taxonomy/term/1>), dans laquelle sont listés tous les nœuds y appartenant. Par ailleurs, chaque ca-

tégorie propose également un fil d'information en continu auquel les utilisateurs peuvent souscrire (par exemple, <http://www.example.com/taxonomy/term/1/0/feed>).

Drupal 4.7 sera doté d'une nouvelle fonctionnalité appelée vocabulaire *free tagging*, proposant à l'utilisateur un champ de formulaire pour y entrer une liste de *tags* séparées par des virgules. Au moment de sauvegarder le nœud édité, toutes ces balises seront créées à la volée sous forme de catégories normales de Drupal. Nous autoriserons, sur notre site, les utilisateurs qui maintiennent des blogs à mettre des balises à leurs entrées de blogs, à l'instar d'autres sites « à balises sociales », comme *del.icio.us*, *flickr*, ou *technorati*. Nous créerons le vocabulaire *Blog Tags* dans *administer* → *categories* → *add vocabulary*, et le marquerons comme vocabulaire *free tagging* (voir la Figure 11). Le module intitulé *tagadelic* représente dans ce domaine une nette amélioration. Ce module propose des *tag clouds*, grappes de noms de balises chargées de relier la catégorie correspondante (ainsi, par exemple, un clic sur la balise *music* permet d'afficher la page de catégorie/balise musique). La taille des polices de caractères de chaque balise dépend de sa popularité. Le module *tagadelic* offre un nuage de balises par vocabulaire, comme par exemple, <http://www.example.com/tagadelic/chunk/1>, ainsi qu'un nuage de balises supplémentaire chargé de combiner les balises de tous les vocabulaires. Par ailleurs, nous afficherons

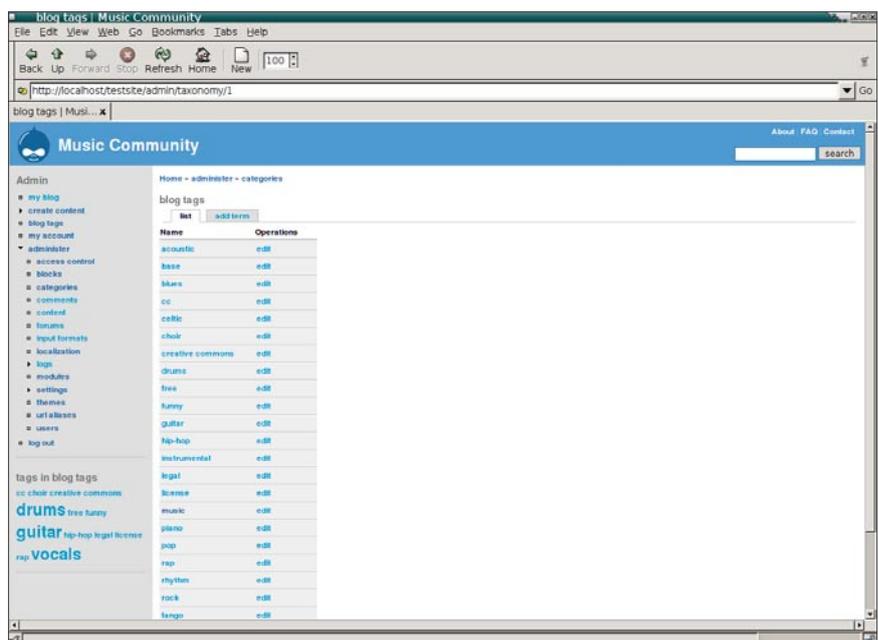


Figure 12. Nuage de balises « blog tags » de notre site, exposé dans un bloc

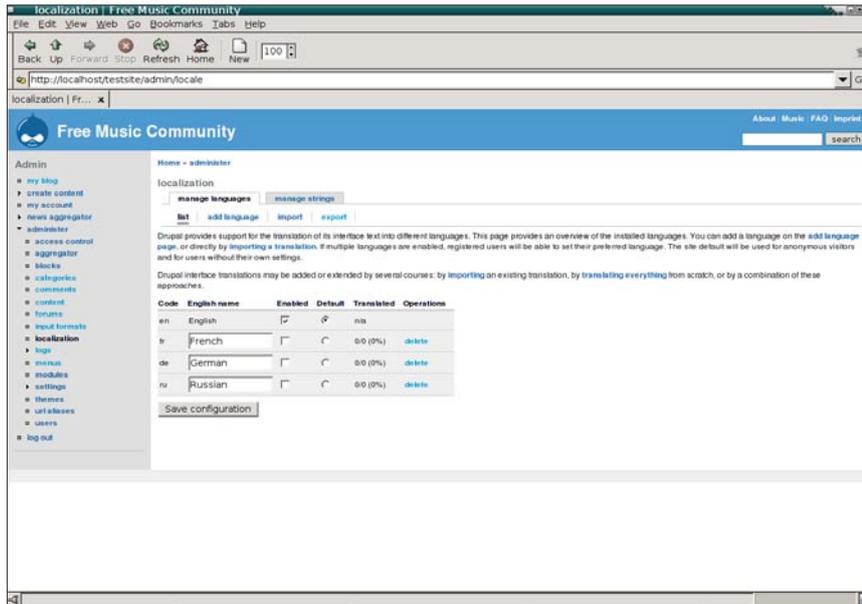


Figure 13. Boîte de réglage du module locale

également le nuage de balise appelé *Blog Tags* dans un bloc en activant le bloc *tags in Blog Tags* dans *administer* → *blocks* (voir la Figure 12). Le nuage de balise offre un moyen flexible à nos visiteurs de naviguer dans le contenu du site en utilisant les balises qui les intéressent le plus.

## Sites internationaux

Nous souhaitons que notre portail soit multilingue. Drupal va nous aider ici sur les points suivants : il va avoir recours au codage de caractères Unicode, et facilitera la création de différentes versions de notre interface et du contenu.

### Module Locale

Le module intégré appelé *locale* permet de traduire l'interface utilisateur Drupal dans une autre langue. Il est possible d'ajouter ou de supprimer des langues dans *administer* → *localization* → *add language*, et de déclarer une de ces langues comme langue par défaut (voir la Figure 13). Nous utiliserons l'anglais comme langue par défaut, tout en ajoutant le polonais, l'allemand, le français et l'italien.

Deux méthodes permettent de traduire l'interface. La première consiste en une interface Web agréable dans *administer* → *localization* → *manage strings*. Il est possible de chercher certaines chaînes (traduites ou non) dans notre site Web, puis de les traduire dans une des langues ajoutées, ou éventuellement d'en modifier la traduction actuelle. Une autre possibilité consiste à importer/exporter l'utilitaire du module *locale*. Les traductions réalisées

par Drupal peuvent, en effet, être importées à partir des fichiers *gettext Portable Object* (.po), ou exportées vers ce format. Ce format est très populaire, et il existe un grand nombre d'applications capables de manipuler ses fichiers, dont les plus connues, en tant qu'applications de traduction bureau, sont *kbabel* ou *poEdit*.

Nul besoin, toutefois, de traduire tous les textes dans la plupart des langues, dans la mesure où *drupal.org* fournit déjà de nombreuses traductions pour les utilisateurs disponibles à l'adresse suivante : <http://drupal.org/project/Translations>. Actuellement, Drupal est traduit en plus de 40 langues. Nous téléchargerons donc pour notre portail, les traductions polonai-

ses, allemandes, italiennes et françaises à partir du site *drupal.org*, puis nous importerons les fichiers *pl.po*, *de.po*, *it.po* et *fr.po* dans *administer* → *localization* → *import*. Il est possible de remplacer les chaînes déjà traduites sur notre site par celles issues du fichier téléchargé, ou de n'ajouter que les chaînes pas encore traduites sur notre site. Veuillez noter qu'il est également possible d'utiliser le module *locale* afin de modifier n'importe quelle chaîne du site. Par exemple, nous pouvons chercher le titre du bloc *Who's online* dans *administer* → *localization* → *manage strings* pour le remplacer par, disons, *Users who are currently online*.

### Internationalisation (traduction du contenu)

Il ne manque plus que le module *Internationalization (i18n)* pour faire de notre portail musical un site parfaitement international. Ce module permet, en effet, de traduire tout le contenu présent sur notre site en d'autres langues. Ce module *i18n* repose sur l'architecture suivante : les traductions sont modélisées sous forme de nœuds, dotées d'un pointeur vers le nœud auquel elles appartiennent (dont elles sont la traduction). L'information relative à la langue est stockée dans l'URL d'un nœud (au lieu d'un cookie de session), comme par exemple, <http://www.example.com/fr/about> pour le nœud français.

Afin de rendre notre site multilingue, nous allons d'abord configurer le module dans *administer* → *settings* → *i18n* (voir la Figure 14). Nous allons activer la fonctionnalité appelée *browser language detection*,

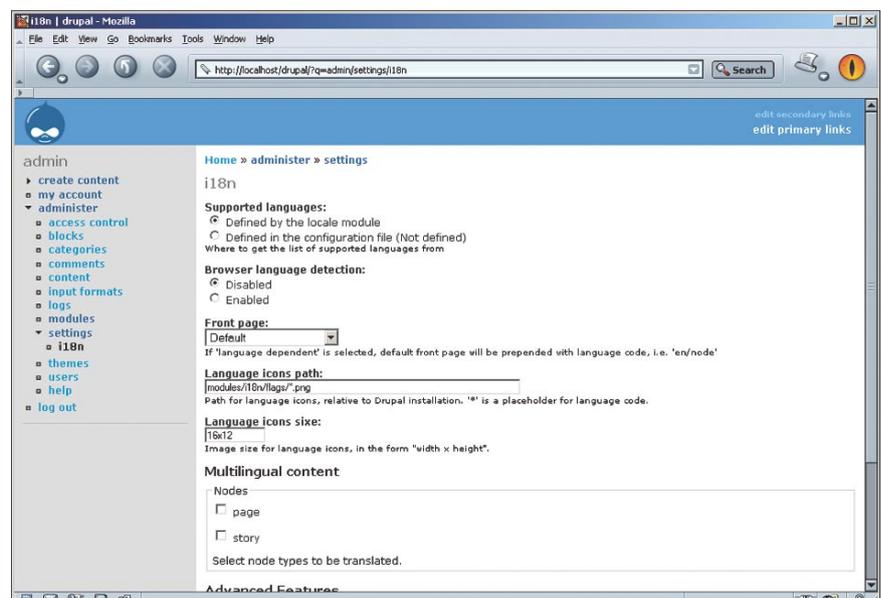


Figure 14. Boîte de réglages du module i18n



Dans chaque numéro :

- fichiers sources
- vidéos pour les tutoriels
- cours multimédia

**Découvrez le cours multimédia sur les nouveautés  
d'Adobe Photoshop CS2 !**

pour plus de détails allez à :

[www.psdmag.org/fr](http://www.psdmag.org/fr)

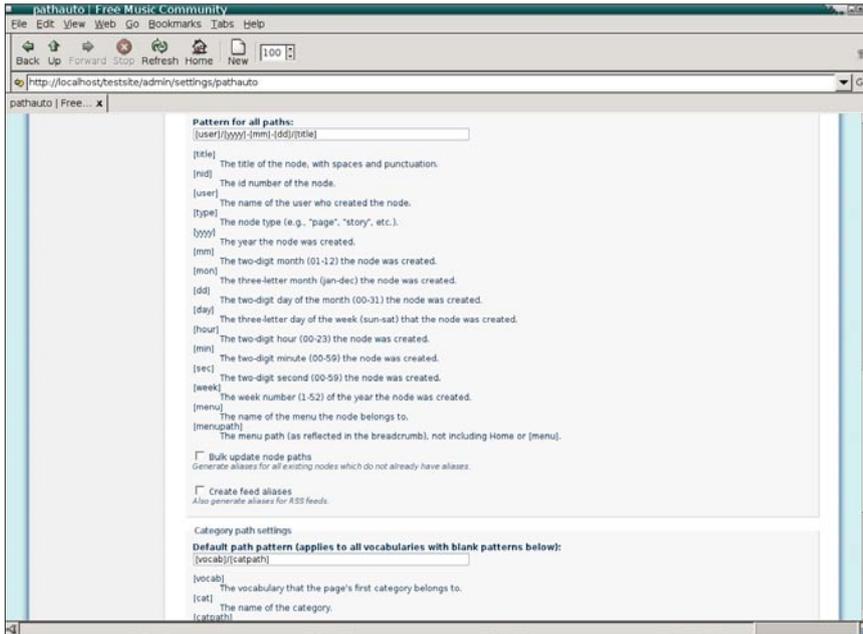


Figure 15. Boîte de réglages du module pathauto

chargée de choisir la langue utilisée pour afficher le contenu du site en fonction des réglages de langues présents dans le navigateur du visiteur. Il s'agit généralement de la méthode la plus pratique. Si nous n'activons pas cette option, la langue sera alors définie à partir de l'URL, des réglages utilisateurs ou de la langue choisie par défaut sur Drupal, dans cet ordre précis.

Nous pourrions également restreindre la traduction de contenu à certains types de nœuds dans *administer* → *settings* → *i18n*, mais, dans le cadre de notre portail, nous autoriserons la traduction de tous les types de nœuds. À titre d'exemple, nous allons traduire la page *About* en français en cliquant sur le lien *translate* du nœud <http://www.example.com/about>. Nous verrons alors la liste des langues définies plus haut dans le module *locale*, puis pourrions choisir de traduire le nœud dans l'une de ces langues. En cliquant sur le lien *create translation* pour le *français*, la boîte de dialogue de création d'un nœud standard s'affiche avec désormais un nouveau champ appelé *Language* (par défaut sur *français* dans ce cas). Nous pouvons désormais entrer le texte traduit dans le corps du nœud, choisir *fr/about* en tant que chemin pour le nœud traduit, puis le sauvegarder.

Le nœud *About* s'affichera désormais dans la langue définie soit à partir de l'URL, soit à partir des réglages du navigateur. Le nœud présente maintenant un ou plusieurs drapeaux en bas, sur lesquels nous pou-

vons cliquer pour obtenir la version correspondante du nœud dans une autre langue.

Le module *i18n* propose également un bloc de sélection de langues pouvant être activé, afin d'afficher la liste de toutes les langues définies. En cliquant sur l'une d'entre elles modifie la langue actuelle (ceci fonctionne également pour les utilisateurs anonymes). Le module présente toutefois certaines limites, comme l'incapacité de traduire le contenu des blocs ou des menus.

## Optimisation pour les moteurs de recherche

Dans la mesure où nous souhaitons améliorer le classement de notre site dans les moteurs de recherche, et augmenter son trafic ainsi que sa popularité, l'optimisation du moteur de recherche est un sujet qui nous intéresse tout particulièrement.

### Nettoyage des URL et du chemin du module

Puisque les principaux moteurs de recherche classent mieux les pages sur lesquelles apparaissent les termes de recherche dans l'URL de la page en question, nous allons optimiser nos URL en fonction de ce critère. Heureusement, Drupal peut être facilement réglé pour créer des URL conviviales pour les moteurs de recherche. Un des réglages parmi les plus importants est *Clean URLs*, qui nous permettra de supprimer `?` de l'URL de notre site, de sorte qu'à la place de <http://www.example.com/?q=node/123>, notre URL ressemble à

<http://www.example.com/node/123>. Il est possible d'activer cette fonctionnalité dans *administer* → *settings*, mais il faut pour ce faire un module Apache opérationnel appelé *mod\_rewrite*. Il est également possible de pousser l'optimisation de nos URL en remplaçant les chiffres qu'elles contiennent par des mots clés pertinents. Cette fonctionnalité est proposée par le module intégré appelé *path*. Une fois ce module activé, l'outil nous demandera de nommer un *path alias* à chaque création, ou édition d'un nœud. Nous pouvons désormais créer une URL à l'aspect suivant : <http://www.example.com/about-this-site>, nous permettant d'être mieux classé par les moteurs de recherche.

### Module Pathauto

Le module intitulé *pathauto* apporte lui aussi de grandes améliorations. Ce module exige le module Drupal standard *path*. Ce module permet d'automatiser de manière flexible la génération de pseudonymes de chemins pour les nœuds, les entrées de blogs, les pages utilisateur, les taxinomies de termes (catégories ou balises), et plus encore. L'objectif de ce module consiste à éviter de fournir un pseudonyme de chemin manuellement à chaque création de nœud, et donc à indiquer un modèle de chemin une seule fois dans *administer* → *settings* → *pathauto* (voir la Figure 15). Jusqu'à présent, à chaque création d'un nouveau nœud, il suffit de laisser le champ *Path alias* vide, et *pathauto* générera automatiquement un pseudonyme de chemin pour ce champ.

Nous allons configurer, pour notre portail, le modèle `[user]/[yyyy]-[mm]-[dd]/[title]` pour les entrées de blogs. Si l'utilisateur *John Doe* crée une entrée de blog intitulée *Why I like good music* la veille de Noël, le module *pathauto* créera automatiquement le pseudonyme de chemin `john-doe/2005-12-24/why-i-like-good-music` pour cette entrée de blog.

### Carte de sites Google

Il est également possible d'augmenter le classement auprès de notre moteur de recherche en utilisant le module spécifique à Google appelé *Google Sitemaps* (*gsitemap*), qui ne fonctionne que pour Google. Ce module crée une carte de site en XML dans <http://www.example.com/gsitemap> au format indiqué par la spécification des cartes de sites Google (<http://www.google.com/webmasters/sitemaps>).

Une carte de site est tout simplement une liste prioritaire de pages issues de notre site Web, que Google va utiliser pour balayer notre site de manière plus intéressante et approfondie, afin de ramener de meilleurs résultats de recherche parmi les plus pertinents. Le module *gsitemap* permet d'ajuster la priorité que certaines pages peuvent obtenir dans la carte du site généré, par exemple, en fonction du type de nœud, du nombre de commentaires pour un nœud, de la date de création/mise à jour du nœud, ou si le nœud est supporté ou non en page frontale. Ce module peut également être configuré afin d'indiquer à Google la date de mise à jour de notre site Web, et pour créer des messages de journaux lorsque le moteur de balayage de Google lit notre carte de site.

### Mots (clés) des nœuds

L'efficacité des méta mots clés par rapport à l'optimisation du moteur de recherche est largement remise en question. En effet, de nombreux moteurs de recherche sont censés les ignorer. Toutefois, en supposant que certains d'entre eux utilisent les mots clés, il est possible de mettre à profit le module intitulé *nodewords*. Ce dernier permet de mentionner un résumé, une description, un avertissement de droit d'auteur pour chaque nœud, ainsi qu'une

liste de mots clés insérés dans la balise `<meta>` de l'en-tête de nos pages. Il est possible de configurer une remarque générale de droits d'auteur, ainsi qu'une liste globale de mots clés qui seront toutes deux utilisées pour l'ensemble des pages, sauf modification nœud par nœud. Nous allons configurer les balises suivantes supplémentaires, à ajouter à chaque page :

```
<meta name="copyright" content="Droit d'auteur 2005 Communauté Musique" />
<meta name="keywords" content="musique, gratuit, indépendant, genre, artistes, portail" />
```

Enfin, nous pouvons utiliser le module *nodewords* pour toujours proposer un ensemble pertinent de méta informations (notamment au niveau des mots clés) pour chaque page de notre site, dans l'espoir d'améliorer un peu plus le classement du site par les moteurs de recherche.

### Performance

Dans l'expectative d'accroître la popularité de notre portail musical, nous allons activer le mécanisme de cache de Drupal dans *administer* -> *settings* ainsi que le module *throttle* proposé avec Drupal. Ce module est chargé de désactiver certains blocs et fonctionnalités sur notre site si la

charge excède certaines limites, elles-mêmes configurables.

Il est possible de sélectionner les modules et les blocs afin de les « réduire » dans le cas d'un effet « slashdot » au moyen de *administer* -> *modules* et de *administer* -> *blocks* en vérifiant les boîtes de contrôle respectives *throttle*.

### Résumé

Comme nous venons de le démontrer dans cet exemple, Drupal est un outil que les utilisateurs de système de gestion de contenus attendent depuis longtemps. Cet outil répond à toutes les attentes que nous avons mentionnées en introduction : il libère l'administrateur de la difficulté à créer et à maintenir des installations indépendantes pour chaque site et version en langue étrangère. Il permet également de faire un usage étendu d'AJAX, en bénéficiant à la fois de ses fonctionnalités dans l'interface et en permettant à l'utilisateur d'utiliser toute sa puissance. Enfin, cerise sur le gâteau, ses capacités en terme d'optimisation pour les moteurs de recherche permet de rendre vos sites Web plus visibles pour les moteurs de recherche.

Grâce à la communauté engagée de Drupal, et à son excellente documentation, vous pouvez être sûr que ce système de contenu de gestion tellement désiré est à ce jour le meilleur choix dans ce domaine. ■

C L U B P R O



#### I-Movix

Développements Oo Php

- Connectivité toutes bases de données
- Apache et IIS
- Hébergements Internet FreeBSD, Sose Enterprise Server 9.0
- Solutions Internet
- Novell Technology Partner

[www.i-movix.com](http://www.i-movix.com)

#### IDEAMA

Ideama est une agence de communication utilisant les diverses supports multimedias pour créer les outils indispensables pour augmenter la concurrence ainsi que pour augmenter l'audience sur le marché par la création des projets web qui satisfèrent vos exigences.

<http://www.ideama.it>

#### CIANCIOLAB

Vous êtes ennuyés par les publicités sur Internet qui ne donnent pas de résultats ? Nous vous offrons les campagnes publicitaires avec l'utilisation des supports publicitaires importants, dirigé vers le secteur commercial et réalisé sur le territoire géographique indiqué par vous. N'attendez pas, contactez :

[info@cianciolab.com](mailto:info@cianciolab.com)  
[www.cianciolab.com](http://www.cianciolab.com)

#### UNOZERO

UNOZERO est une entreprise qui s'occupe de l'analyse, de la création et de la réalisation de logiciel, ainsi que la sécurité et la consultation informatique. Ses services ont pour but d'augmenter la performance des entreprises et ils sont orientés vers le monde Internet/Intranet.

[www.unozero.net](http://www.unozero.net)

#### JCOM ITALIA

- Réalisation et modernisation
- Logiciel pour créer des sites web
- E-commerce – Communication

Sièges:

- Bari-Italia-Via Arcidiacono Giovanni, 48 - 70124
- Padova – Italia – Via Annbale da Bassano, 16-35135

[www.jcomitalia.com](http://www.jcomitalia.com)  
[info@jcomitalia.com](mailto:info@jcomitalia.com)

#### ZEROUNOZERO

- La programmation professionnelle en PHP, conforme aux standards W3C : XHTML, CSS, WAI
- Les sites web et les applications distribuées sur le réseau
- La création et la réalisation des bases de données MySQL, PostgreSQL et Oracle

<http://www.zerounozero.com>  
[info@zerounozero.com](mailto:info@zerounozero.com)

**Plus d'informations :**  
[abonnement@software.com.pl](mailto:abonnement@software.com.pl)  
[www.phpsolmag.org/fr/pro](http://www.phpsolmag.org/fr/pro)