



Juin 2010.

UTILISER

WORDPRESS,

COMME LOGICIEL GESTIONNAIRE DE
COLLECTIONS NUMÉRIQUES,
MODE D'EMPLOI.



Betty Drevet,
Licence professionnelle BDAN.

*Sous la direction de Jean-Michel Mermet,
Chargé de missions documentaires à l'université de Grenoble.*



SOMMAIRE

INTRODUCTION	3
PRÉSENTATION DE WORDPRESS.....	3
LES FONCTIONNALITÉS NATIVES	3
GESTION DES ÉLÉMENTS MULTIMÉDIA	4
GESTION DES UTILISATEURS	4
LES 4 FONCTIONNALITÉS FONDAMENTALES D'UN LOGICIEL DOCUMENTAIRE	5
PERSONNALISATION DES NOTICES DES DOCUMENTS POUR LA SAISIE	5
LES MODULES DE RECHERCHE AVANCÉS.....	6
L’AFFICHAGE DES MÉTADONNÉES.....	6
L’IMPORT ET L’EXPORT DES DONNÉES	8
LES FONCTIONNALITÉS UTILES	10
GÉRER LES DONNÉES MULTIVALUÉES	10
PROPOSER DES REBONDS	10
FAIRE APPARAÎTRE UN FIL D’ARIANE	12
FAIRE APPARAÎTRE LE NOMBRE DE RÉSULTATS	12
INTÉGRER UNE DIMENSION COLLABORATIVE	13
WORDPRESS VERSION 3.0	14
LES CHANGEMENTS.....	14
LES CUSTOM POST TYPES.....	15
LES CUSTOM TAXONOMIES	16
AFFICHER LES CUSTOM POST TYPE.....	16
CONCLUSION	17
LES RESSOURCES À UTILISER.....	18
SITES « OFFICIELS » DE WORDPRESS :.....	18
SITES D'ACTUALITÉS WORDPRESS :	18
SITES D'ACTUALITÉS SPÉCIFIQUES AUX PLUGINS WORDPRESS :.....	18
SITES DE TUTORIELS SUR WORDPRESS :	19
À SUIVRE SUR TWITTER :	19
ANNEXE : FAIRE UN IMPORT D'UNE BASE SUPERDOC VERS UNE BASE WORDPRESS	20

INTRODUCTION

Ce guide est particulièrement destiné aux structures documentaires (musées, bibliothèques, centre de documentation...) souhaitant concevoir une bibliothèque numérique en peu de temps et/ou avec peu de moyens financiers.

Plusieurs cas à titre d'exemple :

- Je suis bibliothécaire dans une université et je souhaite greffer à mon catalogue un site dédié aux travaux d'étudiants (mémoires, thèses, conférences...)
- Je suis employée dans un musée, suite à une exposition, je souhaite mettre en ligne certaines œuvres pour commencer une collection (photographies, cartes...)
- Je suis documentaliste dans une association et je souhaite publier un ensemble de documents (vidéos de conférences, bandes sonores...)

L'idée n'est pas de créer un véritable portail documentaire, ou un catalogue de bibliothèque avancé. L'objectif est de pouvoir, en peu de temps, construire un site capable de gérer une collection de documents multimédias.

Même si quelques connaissances en informatique et en code php peuvent aider, l'utilisation de WordPress ne nécessite pas des compétences techniques pointues.

PRÉSENTATION DE WORDPRESS

Les fonctionnalités natives

WordPress est un logiciel qui permet de créer et gérer facilement l'ensemble d'un site web ou simplement un blog.

Une fois installé, il propose des fonctionnalités correspondant à ses usages classiques d'éditeur de blog. Il gère deux types de documents :

- Les «Pages» : pour créer du contenu statique.
- Les «post» : pour créer des contenus chronodatés.

Les « pages » et les « post » contiennent par défaut les champs suivants : titre, contenu et des champs personnalisés (ce qui permet, par défaut, la multivaluation). Généralement, ces champs personnalisés sont utilisés pour stocker certaines métadonnées supplémentaires, qui peuvent être affichés¹ ou servir à des plugins particuliers.

¹ Voir la partie « Affichage des métadonnées », page 6.

Il est possible d'indexer les documents avec deux types de taxonomies : les mots-clefs ou « tags » et les catégories.

Les tags s'apparentent à des «index» et sont souvent affichés sous forme de « nuage de mots-clefs ». Les catégories s'apparentent à un «thésaurus», elles peuvent être hiérarchisées. L'affichage en front office est généralement sous la forme d'arborescence.

Gestion des éléments multimédia

Au niveau du back office, WordPress possède une bibliothèque dédiée aux éléments multimédia. Les documents sont indexés par les champs titre, texte alternatif, légende, description et notices jointes.

Au niveau du front office, l'affichage de ces contenus est très bien géré par WordPress. Le plus souvent, cela nécessite simplement le téléchargement de quelques plugins.

À titre d'exemple :

- Pour les vidéos : le plugin « Wordpress-flowplayer »,
- Pour les bandes sonores : le plugin « Audio player »,
- Pour les documents pdf ou powerpoint : le plugin « PDF and PPT viewer ».

Bien entendu, la liste n'est pas exhaustive. Mais, pour les avoir testés, ceux-ci fonctionnent bien. Le site « WordPress.org Codex »² est la source de référence quand il s'agit de chercher de nouveaux plugins, n'hésitez pas à la consulter si vous avez des besoins précis.

Gestion des utilisateurs

Wordpress, en tant que blog, est un système multiauteurs, c'est-à-dire qu'il propose, par défaut, la gestion des utilisateurs. Chaque personne peut avoir l'un des rôles prédéterminés : « Administrateur », « Editeur », « Auteur », « Contributeur » ou « Abonné ».

Bien entendu, il est toujours possible de faire évoluer ce module en téléchargeant un plugin spécialisé.

² Voir la partie « Les ressources à utiliser », page 18.

LES 4 FONCTIONNALITÉS FONDAMENTALES D'UN LOGICIEL DOCUMENTAIRE

Pour que Wordpress devienne un outil adapté à la publication de collections numériques il faut :

- Pouvoir définir ou personnaliser des types de notices qui permettent de saisir les métadonnées des objets numériques,
- Pouvoir effectuer des recherches dans un ensemble défini de champs,
- Pouvoir paramétrer précisément l'affichage des informations contenues dans les champs des notices,
- Pouvoir importer/exporter facilement les métadonnées de la collection.

Il n'existe pas, pour le moment, de solution intégrée répondant à toutes ces fonctions. Mais la coordination des pistes proposées ci-dessous permet d'y arriver.

Personnalisation des notices des documents pour la saisie

L'une des premières fonctionnalités documentaires est la possibilité de créer et de personnaliser la fiche descriptive d'un document.

En effet, les besoins en terme d'indexation diffèrent selon chaque structure et selon le type de document (photographie, bande sonore, texte, article...). Il est donc essentiel de pouvoir choisir ses propres champs de description selon chaque type de contenu.

Le plugin « Custom Fields Template » permet de créer des champs spécifiques. Cette extension permet de créer des modèles complets de notices en utilisant des champs personnalisés, simplifiant ainsi la saisie des données. Il est donc possible de créer une notice spécifique aux photographies ou aux vidéos par exemple.

Chaque champ créé est entièrement paramétrable : il peut être de plusieurs types (texte, liste déroulante, bouton radio...), peut contenir des valeurs par défaut, ne pas être pris en compte pour la recherche, etc.

Il est nécessaire de consulter la « Liste des options » afin de connaître toutes les possibilités de configuration de ces champs. Les données multivaluées sont prises en charge avec un code spécifique³.

³ Voir la partie « Gérer les données multivaluées », page 10.

Les modules de recherche avancés

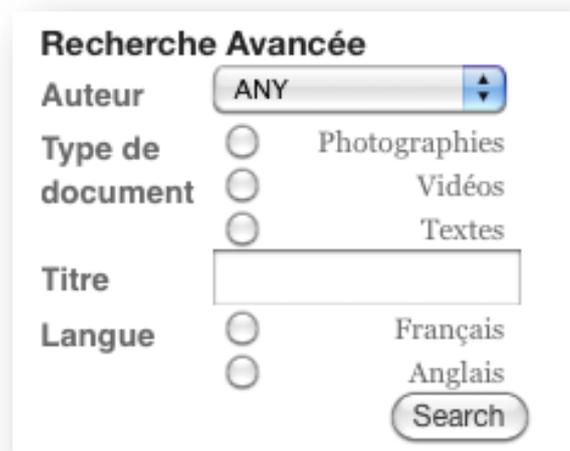
Un CMS doit aussi être capable de proposer des modules et des formulaires de recherche adaptés à la collection (simple, avancé, portant sur les index, full-text...).

« Search everything » est un plugin qui rend la recherche simple plus performante en étendant la recherche aux contenus des commentaires, des tags et des champs spécifiques.

Il est paramétrable et très facile d'utilisation. Il fonctionne très bien.

Pour construire des modules de recherche avancée, c'est-à-dire avec plusieurs champs portant sur différents éléments comme les catégories, les tags, mais aussi sur les champs personnalisés, il est possible d'utiliser le plugin « WordPress Custom Fields Search ».

Ce dernier permet de paramétrer intégralement plusieurs modèles de formulaire de recherche avancée. Il propose de nombreux type de champs (saisie libre, cases à cocher, listes déroulantes, etc.). Il est important de bien sélectionner « champ personnalisé » pour retrouver les éléments que l'on veut proposer pour la recherche.



Recherche Avancée

Auteur

Type de document Photographies
 Vidéos
 Textes

Titre

Langue Français
 Anglais

Avec ce module, il est possible de combiner et de croiser les critères de recherche, comme n'importe quelle autre recherche multi champs.

« Custom Fields Search » fonctionne très bien, son paramétrage est très souple et les possibilités sont nombreuses. C'est un excellent plugin pour la recherche.

L'affichage des métadonnées

Comme nous avons utilisé Custom Fields Template pour créer des notices personnalisées, il faut maintenant les afficher en front office pour les utilisateurs.

Plusieurs solutions sont possibles.

Dans un premier temps, il est possible d'utiliser la fonction WordPress « the_meta » pour faire apparaître tous les champs personnalisés à côté du document.

Il suffit d'insérer dans « single.php » ce script:

```
<?php the_meta(); ?>
```

Le seul inconvénient de cette formule, c'est son manque de souplesse et de contrôle sur les données que l'on fait apparaître. Tous les champs de la notice apparaissent (sauf ceux n'ayant pas de contenu) et dans l'ordre de la saisie de la notice.

La deuxième solution est d'utiliser la fonction « get_post_meta ». Cette dernière est plus souple puisque qu'elle permet l'affichage (ou non) des champs personnalisés que l'on souhaite, dans l'ordre que l'on veut et avec de larges possibilités de « stylisation/paramétrages » des champs.

Le code à utiliser est celui-ci :

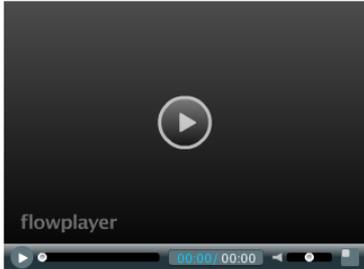
```
<?php if ( get_post_meta($post->ID, « Nom_du_champ_personnalisé », true) ) { ?>
```

Texte que vous voulez voir s'afficher devant la valeur :

```
<?php echo get_post_meta($post->ID, « Nom_du_champ_personnalisé », true); ?> <?php } ?>
```

Il suffit ensuite de l'adapter pour chaque élément que l'on veut voir apparaître en front office. Comme pour la fonction « the_meta », les champs ne possédant pas de contenu ne sont pas affichés.

Le futur de la médecine.
Posted on June 8, 2010 by admin



Titre : Look to the future of medicine.

Auteur(s) / Créateur (s) : [Eva Vertes](#)

Date de la création / publication : 01/02/2005

Editeur(s) / Producteur(s) : TED

Résumé : Eva Vertes, only 19 when she gave this talk, discusses her journey toward studying medicine and her drive to understand the roots of cancer and Alzheimer's.

Format : MP4

Langue : Anglais

À noter, les champs multivalués doivent être affichés avec un code spécifique⁴.

⁴ Voir la partie « Gérer les données multivaluées », page 10.

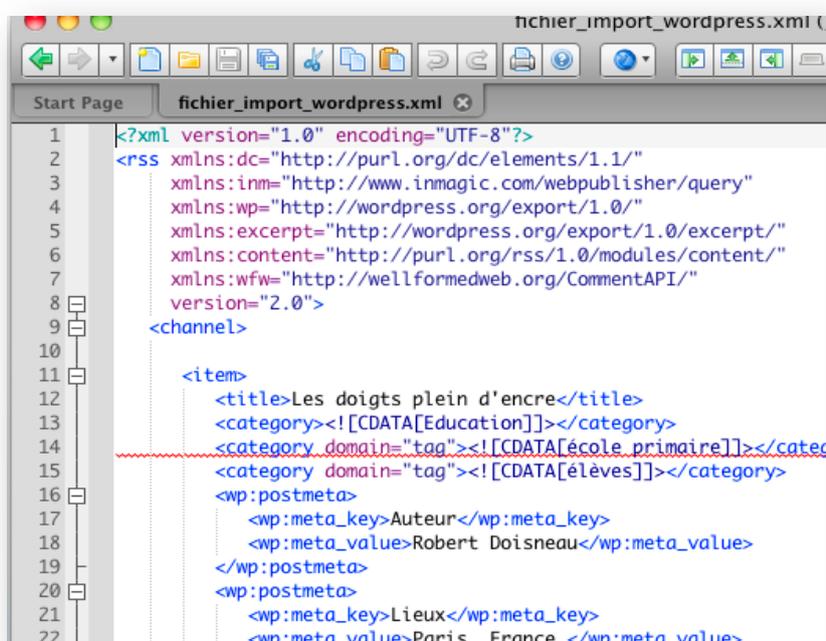
L'import et l'export des données

L'import et l'export des données sont des fonctionnalités primordiales pour permettre la sauvegarde et/ou le partage des informations rentrées dans la base.

Dès la création, WordPress permet d'exporter et d'importer un blog sous un format XML.

Les données telles que les catégories, les champs personnalisés, et toutes les informations relatives aux documents peuvent être sauvegardées et exportées.

En analysant le fichier d'export XML de plusieurs bases WordPress, on peut isoler les parties du code concernant les documents et leur indexation, pour ensuite les adapter (créer une «moulinette»), afin qu'ils puissent être importés dans d'autres bases (WordPress ou autres). Pour cela, il est nécessaire d'utiliser un éditeur XML pour arriver à traiter et modéliser plus facilement les données⁵.



```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <rss xmlns:dc="http://purl.org/dc/elements/1.1/"
3     xmlns:inm="http://www.inmagic.com/webpublisher/query"
4     xmlns:wp="http://wordpress.org/export/1.0/"
5     xmlns:excerpt="http://wordpress.org/export/1.0/excerpt/"
6     xmlns:content="http://purl.org/rss/1.0/modules/content/"
7     xmlns:wfw="http://wellformedweb.org/CommentAPI/"
8     version="2.0">
9 <channel>
10
11 <item>
12 <title>Les doigts plein d'encre</title>
13 <category><![CDATA[Education]]></category>
14 <category domain="tag"><![CDATA[école primaire]]></category>
15 <category domain="tag"><![CDATA[élèves]]></category>
16 <wp:postmeta>
17 <wp:meta_key>Auteur</wp:meta_key>
18 <wp:meta_value>Robert Doisneau</wp:meta_value>
19 </wp:postmeta>
20 <wp:postmeta>
21 <wp:meta_key>Lieux</wp:meta_key>
22 <wp:meta_value>Paris, France.</wp:meta_value>
```

Quand on importe le contenu d'une ancienne base WordPress, on retrouve : les articles, les pages, les catégories, les tags, les liens, etc. Cependant, on ne peut récupérer les données telles que : le thème, les widgets, les plugins de l'ancien blog. Lors de l'import des informations, les anciens liens liés à l'ancien site ou base de données (les retours à l'accueil de certaines pages, les citations d'anciens articles...) ne sont pas automatiquement reformatés (surtout sans la mise en œuvre des permaliens), il faut souvent recréer ces liens...

⁵ Voir l'annexe page 20.

En ce qui concerne les champs personnalisés (entre `<wp:postmeta>` et `</wp:postmeta>`), il faut qu'ils possèdent exactement la même étiquette (`<wp:meta_key>`) pour être correctement importés, casse comprise. Tous les champs personnalisés qui ne correspondent pas à la structure de notice de la base d'accueil sont rajoutés à part.

Custom Field Template

Champs personnalisés "reconnus" Initialiser Enregistrer

Pour les documents

Titre: Traité de dynamique

Auteur: D'Alembert

Ajouter nouveau

Résumé

Ouvrir

Dans le *Traité de dynamique*, il énonce le théorème de d'Alembert (aussi connu sous théorème de Gauss-d'Alembert) qui dit que tout polynôme de degré n à coefficients complexes possède exactement n racines dans \mathbb{C} (non nécessairement distinctes, il faut tenir compte du nombre de fois qu'une racine est répétée). Ce théorème ne sera démontré qu'auXIX^e siècle par Carl Friedrich Gauss. Louis de Broglie présente ce théorème ainsi : « On lui doit le théorème

Chemin:

Lieux: France, Paris

Sujet: Mathématiques

Ajouter nouveau

Champs personnalisés

Nom	Valeur
Biographie	Jean le Rond D'Alembert, ou Jean Le Rond d'Alembert, né le 16 novembre 1717 à Paris où il est mort le 29 octobre 1783. est un mathématicien.

Champ non reconnu, non intégré à la notice

Dans cette logique, il est donc important de bien veiller à ce que la structure des notices importées corresponde à celle de la base de réception.

Lorsqu'il s'agit de faire un import/export d'une base SuperDoc vers WordPress, cela devient un peu compliqué.

Le fichier d'export XML de SuperDoc comporte un type de balisage particulier, qui ne répond évidemment pas à la syntaxe des fichiers d'import XML de données WordPress. On a pu détailler précédemment à quoi ressemblait un import XML pour WordPress, et par conséquent, les balises nécessaires et obligatoires. Il faut donc transformer les balises pour les adapter à la base destinataire des documents. L'outil à utiliser est le langage XSL, puisqu'il permet d'effectuer le « reformatage » depuis n'importe quel export XML vers un XML compatible avec WordPress⁶.

Pour plus d'explication sur la procédure à suivre, un document est fourni en annexe page 19.

⁶ Selon le Dico du web, XSL est un langage permettant de « décrire la manière dont des documents XML doivent être présentés ». Alors que le XML structure le contenu, le XSL est utilisé pour structurer la forme d'un document. « Ce langage permet également de transformer un document XML en un autre document XML, reprenant tout ou partie du contenu original. XSL et les langages associés sont des standards définis par le World Wide WEB Consortium. ».

LES FONCTIONNALITÉS UTILES

Gérer les données multivaluées

La multivaluation est un élément très courant lorsque l'on doit indexer un document. Il n'est pas rare qu'un texte soit écrit par plusieurs auteurs, une conférence animée par plusieurs intervenants, etc. La multivaluation est donc la possibilité d'intégrer plusieurs valeurs dans un champ unique instanciable plus d'une fois.

La multivaluation avec Custom Fields Template est possible à condition de paramétrer correctement le template de création de la notice. Pour « multivaluer » un champ, il faut utiliser les codes :

```
multiple = true  
multipleButton = true
```

De même, l'affichage des valeurs du champ multivalué n'est pas automatique, il faut utiliser un code spécifique :

Auteur(s) du document :

```
<?php $lesauteurs = get_post_meta($post->ID,"Auteur", false);  
/* La variable $lesauteurs contient maintenant un tableau du champ  
multivalué Auteur */  
foreach ($lesauteurs as $value)  
echo " ".$value."<br/>" ;?>
```

Proposer des rebonds

Les rebonds sont des liens sur des valeurs affichées dans une notice qui permettent de relancer une recherche basée sur le texte cliqué. Par exemple, la présence de rebond dans le champ auteur permet d'accéder à tous les documents concernant l'auteur « cliqué ».

Pour rendre possible le rebond, on personnalise l'URL créée par une recherche via le plugin Custom Field Search. Par exemple, si l'on fait une recherche sur le champ « Lieux » (ex. France), l'URL se présente comme ceci (sans les sauts de ligne) :

http://localhost:8888/wordpress_01/?searchclass=DB_CustomSearch_Widgetdb_customsearch_widget&widget_number=preset-default&csTitre-0=&cs-Auteur-1=&cs-Lieux2=France,%Paris&search=Rechercher

Le code à utiliser pour un champ monovalué est celui-ci :

```
<?php if ( get_post_meta($post->ID, "Lieux", true) ) { ?>
```

```

<br> <i> Lieux : </i> <br>
<?php
$lieuvalue = get_post_meta($post->ID,"Lieux", true);
echo '<a href="http://localhost:8888/wordpress_01/?search-
class=DB_CustomSearch_
Widget-db_customsearch_widget&widget_number=2&cs-Titre-0=&cs-
Auteur-1=&cs-Lieux-2=' . $lieuvalue . '&search=Rechercher">' .
$lieuvalue . '</a>';
?>

```

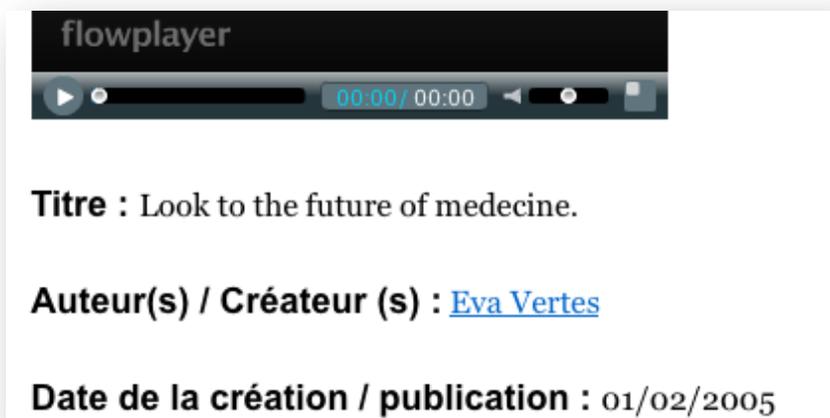
Le code à utiliser en cas de champ multivalué est celui-ci :

```

<i> Auteur(s) du document : </i><br>
<?php $lesauteurs = get_post_meta($post->ID,"Auteur", false);
foreach ($lesauteurs as $value)
    echo '<a href="http://localhost:8888/wordpress_01/?search-
class=DB_CustomSearch_Widget-db_
customsearch_widget&widget_number=2&cs-Titre-0=&cs-
Auteur-1=' . $value . '&cs-Lieu-2=&search=Rechercher">' . $value .
'</a>' . "<br/>" ;
?>

```

Tous ces codes sont à adapter en fonction de ce sur quoi on veut rebondir.



Un problème peut survenir si le nom du champ personnalisé possède des caractères qui ne sont pas interprétés correctement dans le code.

Il faut faire attention à veiller à ce que le nom des champs personnalisés ne comportent pas de tiret, une telle formule :

```

<?php if ( get_post_meta($post->ID, "annee-publication" , true) )
{ ?>

```

Ne pourra pas être prise en compte. Par contre les espaces sont tolérés.

Il faut changer l'intitulé du champ (par exemple « année de publication ») ou intégrer une formule php spécifique qui encodera correctement les tirets.

Les rebonds apportent beaucoup de convivialité et améliorent l'utilisabilité du site. On le voit, leur mise en œuvre n'est pas complexe, inutile de s'en priver !

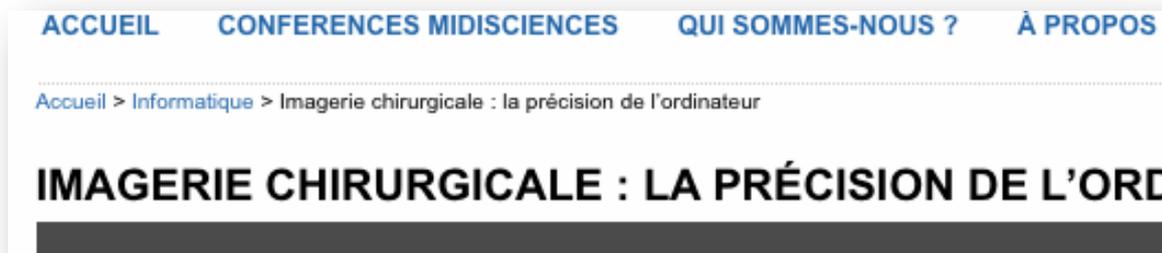
Faire apparaître un fil d'Ariane

Pour faciliter au mieux la navigation, il est possible d'ajouter un fil d'Ariane. Cela permet à l'utilisateur de pouvoir se repérer dans la hiérarchie du document. Quand un utilisateur sélectionne un podcast, le fil d'Ariane apparaît en détaillant la catégorie de la vidéo par exemple.

Il suffit de télécharger plugin « Breadcrumb NavXT » et d'ajouter cette ligne de code :

```
<div class="breadcrumb">
<?php
if(function_exists('bcn_display'))
{ bcn_display(); } ?>
</div>
```

Dans « page.php » et « single.php » juste en dessous du header (/header).



Faire apparaître le nombre de résultats

Pour afficher le nombre de résultats après que l'utilisateur ait effectué une recherche, il faut insérer dans « recherche.php » ce code :

```
<?php echo("<p>Votre recherche a abouti sur $wp_query->found_posts  
r&eacute;sultats.</p>"); ?>
```

INTÉGRER UNE DIMENSION COLLABORATIVE

WordPress offre des services spécifiques au web collaboratif avec plus de facilité que n'importe quel autre outil gestionnaires de collections, puisque que ce sont des fonctionnalités fondamentales des blogs.

Actuellement, des systèmes comme les réseaux sociaux et/ou des applications courantes (telle que Google Search, Maps, Doc, Reader, Delicious ...) sont de plus en plus utilisées de manières « professionnelles ». Proposer des fonctionnalités de partage dans un blog « professionnel » n'est pas incohérent dans la mesure où toutes ces applications deviennent des outils de travail.

Si WordPress est un outil souple et facilement adaptable à différents types de collections, il devient également une solution pertinente si l'on veut élargir sa simple fonction de gestionnaire de contenu.

Utiliser WordPress comme propulseur peut devenir une « innovation » : permettant de proposer, en plus des simples fonctionnalités « documentaires », de nouveaux services tels que les forums, les flux d'informations continus et personnalisés (RSS), des possibilités de partage, etc.

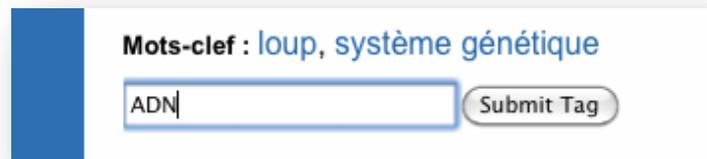
J'ai établi une sélection (bien sûr non exhaustive) de plugins qui donnent au blog une dimension collaborative et participative.

- Sociable : Ce plugin permet d'insérer des liens vers l'ensemble des réseaux sociaux sous forme de mini-icônes que l'on peut disposer, soit en bas de chaque article (attention, cela peut devenir lourd) ou sur chaque page...



- Contact Form 7 : Ce plugin permet de créer, gérer, paramétrer un ou plusieurs formulaires de contact. Il est très souple.
- Wordbook : Ce plugin permet de publier sur Facebook toutes les nouveautés du site WordPress.
- Wordtwit : Dans le même esprit que Wordbook, ce plugin permet à un usager de "twitter" toutes les nouveautés du site WordPress.
- Interesting Link List : Ce plugin permet aux usagers d'alimenter une liste de liens (avec la validation de l'administrateur)

- WP-Folksonomy : Ce plugin permet aux utilisateurs de rajouter des tags, ou mots-clefs, aux articles du site (comme pour Delicious ou Flickr)



WORDPRESS VERSION 3.0

Au jour d'aujourd'hui, la version 3.0 de WordPress n'est pas encore disponible. En revanche, il est possible de télécharger la version bêta 3.02 sur le site de WordPress.org.

Les changements

Les principaux changements que l'on retrouve avec la version 3.0 de WordPress sont :

- Choix du login et du mot de passe :

On peut choisir dès l'installation du blog son propre login et son propre mot de passe, au lieu du login « admin » par défaut qu'il fallait changer après.

- Changement du thème par défaut :

Ce nouveau thème se nomme Twenty Ten. Il est très facilement personnalisable puisqu'il est directement possible de changer l'en-tête et l'arrière-plan du blog à partir de l'interface d'administrateur. De plus, la traduction française de ce thème est déjà fournie.

- La gestion multisite :

Un lien "Network" est accessible dans l'onglet "Outils" de l'interface administrateur, il permet d'accéder aux paramètres de votre système multi-sites. C'est une fusion entre le WordPress classique et WordPress MU (Multi User).

- La gestion de modules d'articles

Il est désormais possible de créer des modules d'articles spécifiques, grâce aux Cutoms Post Type. Cette fonctionnalité très intéressante, d'un point de vue CMS, est détaillée dans la partie suivante. Pour plus d'informations sur les nouveautés de Wordpress 3.0, je vous conseille de consulter ces articles :

→ 10 Features to Look Forward to in WordPress 3.0 (en anglais) :

<http://net.tutsplus.com/tutorials/wordpress/10-features-to-look-forward-to-in-wordpress-3-0/>

→ Looking forward WP 3.0 (en anglais) :

<http://justintadlock.com/archives/2010/03/24/looking-forward-to-wordpress-3-0>

→ Prévoir et paramétrer Wordpress 3.0, un peu de docu (en français) :

<http://lashon.fr/wordpress/se-preparer-a-wordpress-3-0/>

→ Les nouveautés de WordPress 3.0 ! (en français) :

<http://www.geekeries.fr/wordpress/les-nouveautes-de-wordpress-3-0-5412>

Les Custom Post Types

La version 3 de WordPress inclut dans ses nouvelles fonctionnalités la possibilité de créer des « modules » d'édition d'articles. Cette fonction n'est pas automatisée (et ne le sera sûrement pas pour éviter de surcharger l'administrateur « de base » de WordPress), c'est pourquoi il faut de nouveau manipuler un peu de code.

Ces structures d'articles sont totalement personnalisables.

Pour intégrer le menu ci-dessus, il vous suffit de coller le code suivant dans le fichier « functions.php » du thème WordPress :

```
function post_type_videos() { register_post_type( 'videos',
array( 'label' => __('Les vidéos'), 'public' => true, 'show_ui' => true )
);register_taxonomy_for_object_type('post_tag', 'videos'); }
add_action('init', 'post_type_videos');
```

Le code ci-dessus permet d'obtenir un type de notice « Vidéo » contenant les éléments de base gérés par WordPress : titre, contenu, gestion des mots-clés (post-tags).

Il est aussi possible de rajouter des éléments standards ou personnalisés (les « méta-box ») de façon à finaliser le paramétrage de la notice : « post-tag », « comment », « custom fields », ...

Le code devient alors :

```
function post_type_videos() {
    register_post_type(
        'videos',
        array('label' => __('Les vidéos'),
            'public' => true,
            'show_ui' => true,
            'supports' => array(
                'post-thumbnails',
                'excerpts',
                'trackbacks',
                'custom-fields',
                'comments',
                'revisions')));
    register_taxonomy_for_object_type('post_tag', 'videos');
}
```

```
add_action('init', 'post_type_videos');
```

Pour plus d'informations sur les nouveautés de Wordpress 3.0, je vous conseille de consulter ces articles :

→ Impressions of Custom Post Type :

<http://wpengineer.com/impressions-of-custom-post-type/>

→ Custom Post Type in WordPress :

<http://justintadlock.com/archives/2010/04/29/custom-post-types-in-wordpress>

Les Custom Taxonomies

Les Custom Post intègrent des « Custom Taxinomies », c'est-à-dire la possibilité d'ajouter des modules de mots-clefs ou de catégories propres.

Il est possible, par exemple, de créer à l'intérieur d'une notice vidéo, un module « Realisateur » (qui comprend une liste de réalisateurs) et un module « post Tags » (qui utilise des tags/mots-clés spécifiques).

Pour faciliter la création et la configuration de ces « Custom Taxonomies », il est possible de télécharger le plugin « Custom Post UI ». Ce plugin produit exactement les mêmes codes mais offre une interface plus ergonomique.

Pour plus d'information, voici quelques références :

→ Introducing WordPress 3, Custom Taxonomies :

<http://net.tutsplus.com/tutorials/wordpress/introducing-wordpress-3-custom-taxonomies/>

→ Site WordPres du plugin Custom Post Type UI :

<http://wordpress.org/extend/plugins/custom-post-type-ui/>

Afficher les Custom Post Type

L'affichage du contenu des Custom Post n'est pas automatique.

Il faut utiliser un code spécifique (dans functions.php) pour que les documents apparaissent dans le blog WordPress dans la même boucle (au sens WordPress) que les posts :

```
add_filter( 'pre_get_posts', 'my_get_posts' );  
function my_get_posts( $query ) {  
if ( is_home() )  
$query->set( 'post_type', array( 'post', 'textes', 'videos',  
'bande sonore' ) );
```

```
return $query; }
```

Ici, on demande l'affichage des Custom Posts de type « textes », « videos » et « bande sonore ». Vu la relative « jeunesse » de ces fonctionnalités, il reste encore quelques problèmes à régler comme les modalités de recherche, l'intégration des ces nouveaux posts dans les flux RSS (pour l'instant seuls les posts semblent y être intégrés), les possibilités d'affichage des taxonomies, etc. D'autres plugins seront sans doute bientôt disponibles afin de faciliter au mieux la gestion des Custom Post et des Custom Taxonomies.

Pour plus de détails, vous pouvez consulter l'article de Justin Tadlock :

→ Showing Custom post Type :

<http://justintadlock.com/archives/2010/02/02/showing-custom-post-types-on-your-home-blog-page>

Je tiens à préciser que toutes les applications et plugins développés en amont ont été testés et fonctionnent très bien sur la version 3.0 Béta 1 de WordPress.

CONCLUSION

WordPress a l'avantage de posséder une communauté dynamique qui fait évoluer sans cesse le logiciel. Sa simplicité de mise en œuvre et sa flexibilité au niveau des applications disponibles en font un CMS de qualité quand il s'agit de gérer une collection numérique.

Avec la version 3.0, le logiciel tend à se s'élargir davantage en offrant de nouvelles fonctionnalités de plus en plus avancées dans la gestion documentaire.

Il faudra sans doute surveiller de près toutes ces nouveautés : ces évolutions sont autant de promesses pour faciliter l'administration, l'organisation et la publication en ligne d'une collection de documents multimédias.

LES RESSOURCES À UTILISER

WordPress possède l'une des communautés d'utilisateurs et de développeurs les plus dynamiques au monde actuellement. Il existe de nombreux blogs consacrés uniquement à WordPress, en plus des documentations « officielles » disponibles.

Sites « officiels » de WordPress :

Site officiel francophone de WordPress : <http://www.wordpress-fr.net/>

Site WordPress.com : <http://wordpress.com/>

WordPress.org Codex : http://codex.wordpress.org/Main_Page (en anglais)

→ Le site se présente comme une encyclopédie des extensions, fonctionnalités, plugins, et autres codes disponibles pour WordPress. C'est l'une des sources d'information les plus complètes.

The WordPress Blog : [The WordPress Blog](#)

→ Blog d'article spécifique à WordPress.

Sites d'actualités WordPress :

Here With Me : <http://www.herewithme.fr/>

Digging Into WordPress : <http://digwp.com/> (en anglais)

Sites d'actualités spécifiques aux plugins WordPress :

L'écho des Plugins : <http://www.li-an.fr/wpplugins/>

→ Ce blog recense quelques nouveautés intéressantes en matière de plugins de WordPress.

Lashon : <http://lashon.fr/>

→ Blog d'un concepteur de sites web et de blogs, articles intéressants sur des fonctionnalités WordPress.

Geekeries : <http://www.geekeries.fr/>

→ Site qui offre de nombreuses astuces et des articles intéressants sur WordPress.

Justin Tadlock: <http://justintadlock.com/> (en anglais)

→ Site très technique d'un blogueur professionnel. De nombreux articles de référence à consulter sur WordPress 3.0.

Sites de tutoriels sur WordPress :

WordPress Channel : <http://wpchannel.com/>

→ Excellent site de références, propose à la fois des articles d'actualités, des tutoriels, des podcasts et des forums dédiés.

Net.Tuts : <http://net.tutsplus.com/> (en anglais)

→ Site de différents tutoriels WordPress.

WordPressZone.com : <http://www.wordpresszone.com/> (en anglais)

→ Site agrégateur de nombreux articles sur les thèmes, astuces et plugins de WordPress. De très bons tutoriels sont disponibles.

Mister-WordPress : <http://debutant-wordpress.com/>

→ Ce site propose un ensemble de screencast sur les nouveautés WordPress, les démonstrations sont claires et bien construites.

À suivre sur Twitter :

- geekerief : Étudiant en graphisme et créateur de Geekerief.fr.
- Wordpressdotcom : Informe des actualités relatives à WordPress (en anglais).
- DianeBourque : Spécialiste Web dans WordPress et les différents réseaux sociaux... (en français et anglais).

ANNEXE : FAIRE UN IMPORT D'UNE BASE SUPERDOC VERS UNE BASE WORDPRESS

Le fichier d'export de SuperDoc :

La première étape consiste à créer un fichier d'export de la base Superdoc en format XML.

Le code obtenu est semblable à celui-ci :

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<inm:Results productTitle="Superdoc Premium" productVersion="9.00"
xmlns:inm="http://www.inmagic.com/webpublisher/query">
<inm:Recordset setCount="7">
<inm:Record setEntry="0">
<inm:Auteur>Robert Doisneau</inm:Auteur>
<inm:Titre>Les doigts plein d'encre</inm:Titre>
<inm:Sujet>Education</inm:Sujet>
<inm:MotsCles>école primaire</inm:MotsCles>
<inm:MotsCles>élèves</inm:MotsCles>
<inm:Lieux>Paris, France.</inm:Lieux>
<inm:Copyright>Robert Doisneau</inm:Copyright>
<inm:DocumentAssocie>C:..../les_doigts_plein_d_encre.jpg</inm:DocumentAss
ocie>
</inm:Record>
<inm:Record setEntry="1">
<inm:Auteur>Raymond Depardon</inm:Auteur>
<inm:Titre>Combattant à Beyrouth</inm:Titre>
<inm:Sujet>Conflit</inm:Sujet>
<inm:MotsCles>soldat</inm:MotsCles>
<inm:MotsCles>guerre</inm:MotsCles>

<inm:Lieux>Beyrouth, Liban.</inm:Lieux>
<inm:Copyright>Agence Magnum</inm:Copyright>
<inm:DocumentAssocie>C:..../combattant_beyrouth.jpg</inm:DocumentAssocie>
</inm:Record>
</inm:Recordset>
</inm:Results>
```

(Le code a été simplifié pour ne garder que quelques champs et deux items).

Observations:

Les différents champs (<inm:Nom du champ>) vont être traduit comme des champs personnalisés de Wordpress (<wp:postmeta >),

Le champ « Sujet » de Superdoc va être traduit par les catégories de Wordpress

Le champ « Mots-clés » de Superdoc va être traduit par les tags de Wordpress.

Le fichier d'import de Wordpress :

On a pu détailler précédemment à quoi ressemblait un import pour Wordpress, et par conséquent, des balises nécessaires et obligatoires du code XML.

Le résultat que l'on veut obtenir pour établir la compatibilité des données est celui-ci :

```
<?xml version="1.0" encoding="UTF-8"?>
<rss xmlns:dc="http://purl.org/dc/elements/1.1/"
    xmlns:inm="http://www.inmagic.com/webpublisher/query"
    xmlns:wp="http://wordpress.org/export/1.0/"
    xmlns:excerpt="http://wordpress.org/export/1.0/excerpt/"
    xmlns:content="http://purl.org/rss/1.0/modules/content/"
    xmlns:wfw="http://wellformedweb.org/CommentAPI/"
    version="2.0">
  <channel>

    <item>
      <title>Les doigts plein d'encre</title>
      <category><![CDATA[Education]]></category>
      <category domain="tag"><![CDATA[école primaire]]></category>
      <category domain="tag"><![CDATA[élèves]]></category>
      <wp:postmeta>
        <wp:meta_key>Auteur</wp:meta_key>
        <wp:meta_value>Robert Doisneau</wp:meta_value>
      </wp:postmeta>
      <wp:postmeta>
        <wp:meta_key>Lieux</wp:meta_key>
        <wp:meta_value>Paris, France.</wp:meta_value>
      </wp:postmeta>
      <wp:postmeta>
        <wp:meta_key>Copyright</wp:meta_key>
        <wp:meta_value>Robert Doisneau</wp:meta_value>
      </wp:postmeta>

      <wp:attachment_url>C:...../les_doigts_plein_d_encre.jpg</wp:attachment_url
      >

    </item>

    <item>
      <title>Combattant à Beyrouth</title>
```

```

<category><![CDATA[Conflit]]></category>
<category domain="tag"><![CDATA[soldat]]></category>
<category domain="tag"><![CDATA[guerre]]></category>
<wp:postmeta>
  <wp:meta_key>Auteur</wp:meta_key>
  <wp:meta_value>Raymond Depardon</wp:meta_value>
</wp:postmeta>
<wp:postmeta>
  <wp:meta_key>Lieux</wp:meta_key>
  <wp:meta_value>Beyrouth, Liban.</wp:meta_value>
</wp:postmeta>
<wp:postmeta>
  <wp:meta_key>Copyright</wp:meta_key>
  <wp:meta_value>Agence Magnum</wp:meta_value>
</wp:postmeta>

<wp:attachment_url>C:..../combattant_beyrouth.jpg</wp:attachment_url>
  </item>

</channel>
</rss>
</xml>

```

Le fichier XSL

Pour établir et « automatiser » les changements de balises des données, il faut manier le XSL.

Le « dico du web » propose une bonne définition de ce langage :

« XSL(acronyme pour Extensible Stylesheet Language) est un langage qui permet de décrire la manière dont des documents XML doivent être présentés. Ainsi, là où XML structure le fond d'un document, son contenu, XSL est utilisé pour structurer sa forme »⁷.

Il est intéressant de noter qu'un document XML transformé à l'aide de XSL est lui-même un document XML.

Le code XSL nécessaire à la transformation d'un XML de Superdoc en un XML pour Wordpress est présenté ci-dessous. Chaque partie du code est commentée.

Pour commencer, le fichier comporte toujours son format et ses spécificités techniques :

```
<?xml version="1.0" encoding="UTF-8"?>
```

7 Source :Le dico du web, <http://www.olecorre.com/1-29-Definition.php?Id=3622> (dernière visite le 26/04/10)

Il faut ensuite établir la description XSL et le paramétrage des « espaces de noms » :

- « inm » correspond au fichier issu de Superdoc
- « wp » est celui exigé pour l'import dans une base Wordpress

```
<xsl:stylesheet version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:inm="http://www.inmagic.com/webpublisher/query"
  xmlns:wp="http://wordpress.org/export/1.0/">
```

On établit ensuite un nouveau fichier xml :

```
<xsl:output method="xml" version="1.0" encoding="UTF-8" indent="yes"/>
```

On supprime les espaces et lignes inutiles.

```
<xsl:strip-space elements="*" />
```

On examine le noeud : « inm:Recordset » (on se positionne en haut de la base de l'export de Superdoc) et on initialise le contenu du fichier par les balises nécessaires.

```
<xsl:template match="inm:Recordset">
  <rss version="2.0"
xmlns:excerpt="http://wordpress.org/export/1.0/excerpt/"
xmlns:content="http://purl.org/rss/1.0/modules/content/"
xmlns:wfw="http://wellformedweb.org/CommentAPI/"
xmlns:dc="http://purl.org/dc/elements/1.1/"
xmlns:wp="http://wordpress.org/export/1.0/">
  <channel>
```

On établit une suite d'actions sur chaque enregistrement :

```
<xsl:for-each select="inm:Record">
  <item>
```

On remplit la balise « title » avec le contenu de la balise « inm:Titre de chaque inm:Record ».

Important : le champ « title » semble être le seul indispensable pour que l'importation dans Wordpress fonctionne. C'est logique, car c'est à partir du contenu de ce champ que Wordpress détermine le « slug » qui sert, à son tour, à construire le permalien.

```
<title><xsl:value-of select="inm:Titre"/></title>
```

On prévoit la multivaluation de certains champs.

```
<xsl:for-each select="inm:Sujet">
  <category>
```

Pour spécifier un texte qui comprend une syntaxe XML (caractères de balises), on utilise une syntaxe particulière pour éviter l'interprétation par l'analyseur XSL.

```
<xsl:text disable-output-
escaping="yes">&lt;![CDATA[</xsl:text><xsl:value-of select="." /><xsl:text
disable-output-escaping="yes">]]&gt;</xsl:text>
</category>
```

```

        </xsl:for-each>

        <xsl:for-each select="inm:MotsCles">
          <category domain="tag">
<xsl:text disable-output-
escaping="yes">&lt;![CDATA[</xsl:text><xsl:value-of select="."/><xsl:text
disable-output-escaping="yes">]]&gt;</xsl:text>
          </category>
        </xsl:for-each>

```

Un exemple de traitement de champ personnalisé multivalué :

```

<xsl:for-each select="inm:Auteur">
  <wp:postmeta>
    <wp:meta_key>Auteur</wp:meta_key>
    <wp:meta_value>
      <xsl:value-of select="."/>
    </wp:meta_value>
  </wp:postmeta>
</xsl:for-each>

```

Les champs personnalisés suivants sont considérés comme monovalués :

```

<wp:postmeta>
  <wp:meta_key>Lieux</wp:meta_key>
  <wp:meta_value>
    <xsl:value-of select="inm:Lieux"/>
  </wp:meta_value>
</wp:postmeta>

<wp:postmeta>
  <wp:meta_key>Copyright</wp:meta_key>
  <wp:meta_value>
    <xsl:value-of select="inm:Copyright"/>
  </wp:meta_value>
</wp:postmeta>

```

```

<wp:attachment_url><xsl:value-of
select="inm:DocumentAssocie"/></wp:attachment_url>
  </item>
</xsl:for-each>
</channel>
</rss>
</xsl:template>
</xsl:stylesheet>

```

Bien sur, il faut transformer ce code suivant le contexte : espaces de noms, noms des champs, multivaluation...

Concrètement, pour concevoir et appliquer un code XSL sur un fichier XML on peut utiliser les logiciels XMLSpear (logiciel gratuit et multiplateformes) ou XMLSpy (payant).

Pour vérifier que votre code est correct, vous pouvez aussi utiliser cette adresse :

<http://www.w3schools.com/xsl/tryxslt.asp?xmlfile=catalog&xsltfile=catalog>

Le code XSL finalisé, il suffit de l'appliquer sur le fichier d'export Superdoc pour obtenir un fichier XML importable pour Wordpress.