



Yoran Brault

accès
libre

Concevoir et déployer
ses sites web avec

Drupal

Drupal 6
Présentation
de Drupal 7

Préface d'Edwy Plenel

EYROLLES

Concevoir et déployer
ses sites web avec

Drupal

Linux aux petits oignons. Les meilleures recettes pour bien débiter !

K. NOVAK.

N°12424, 2009, 524 pages avec DVD-Rom.

OpenOffice.org 3 efficace.

S. GAUTIER, G. BIGNEBAT, C. HARDY, M. PINQUIER.

N°12408, 2009, 408 pages avec CD-Rom.

MediaWiki efficace. Installer, utiliser et administrer un wiki.

D. BARRETT, adapté par S. BLONDEEL.

N°12466, 2009, 374 pages.

Réussir un site web d'association... avec des outils libres !

A.-L. QUATRAVAUX ET D. QUATRAVAUX.

N°12000, 2^e édition, 2007, 372 pages.**Bien rédiger pour le Web... et améliorer son référencement naturel.**

I. CANIVET.

N°12433, 2009, 412 pages.

Ergonomie web. Pour des sites web efficaces.

A. BOUCHER.

N°12479, 2^e édition, 2009, 458 pages.**Améliorer son taux de conversion web.**

S. ROUKINE.

N°12499, 2009, 250 pages.

Joomla et Virtuemart. Réussir sa boutique en ligne.

V. ISAKSEN, T. TARDIF.

N°12381, 2008, 306 pages.

Réussir son site web avec XHTML et CSS.

M. NEBRA.

N°12307, 2^e édition, 2008, 306 pages.**Réussir son site e-commerce avec osCommerce.**

D. MERCER.

N°11932, 2007, 446 pages.

Réussir un projet de site web.

N. CHU.

N°12400, 5^e édition, 2008, 246 pages.**Ubuntu efficace.**

L. DRICOT, K. NOVAK.

N°12362, 3^e édition, à paraître 2009, 360 pages avec CD-Rom.**Boostez votre efficacité avec FreeMind.**

X. DELENGAIGNE, P. MONGIN.

N°12448, 2009, 260 pages.

Gimp 2.4 efficace. Dessin et retouche photo.

C. GÉMY.

N°12152, 2^e édition, 2008, 402 pages avec CD-Rom.**Inkscape efficace. Réussir ses dessins vectoriels.**

C. GÉMY.

N°12425, 2009, 280 pages

La 3D libre avec Blender.

O. SARAJA.

N°12385, 3^e édition, 2008, 456 pages avec DVD-Rom.**Dessiner ses plans avec QCad. Le DAO pour tous.**

A. PASCUAL.

N°12397, 2009, 278 pages.

Mise en page avec OpenOffice.org Writer.

I. BARZILAI.

N°12149, 2007, 338 pages.

Scenari – La chaîne éditoriale libre.

S. CROZAT.

N°12150, 2007, 200 pages.

PGP/GPG. Assurer la confidentialité de ses mails et fichiers.

M. LUCAS, AD. PAR D. GARANCE, CONTRIB. J.-M. THOMAS.

N°12001, 2006, 248 pages.

Monter son serveur de mails sous Linux

M. BÄCK et al., adapté par P. TONNERRE.

N°11931, 2006, 360 pages.

Tiny ERP/Open ERP. Pour une gestion d'entreprise efficace et intégrée.

F. PINCKAERS, G. GARDINER.

N°12261, 2008, 276 pages.

Collection « Poche Accès Libre »**SPIP 2. Premiers pas pour créer son site avec Spip 2.0.3.**

A.-L. QUATRAVAUX, D. QUATRAVAUX.

N°12502, 2009, 300 pages.

Dotclear 2. Créer et administrer son blog.

A. CAILLIAU.

N°12407, 2008, 242 pages.

Réussir son blog professionnel.

T. PARISOT.

N°12514, 2009, 270 pages.

Économie du logiciel libre.

F. ELIE.

N°12463, 2009, 195 pages.

Gimp 2.6. Débuter en retouche photo et graphisme libre.

D. ROBERT.

N°12480, 4^e édition, 2009, 352 pages.**Inkscape. Premiers pas en dessin vectoriel.**

N. DUFOUR, E. DE CASTRO GUERRA.

N°12444, 2009, 376 pages.

Hackez votre Eee PC. L'ultraportable efficace.

C. GUELFF.

N°12437, 2009, 306 pages.

Premiers pas en CSS et HTML.

F. DRAILLARD.

N°12390, 2^e édition, 2008, 256 pages.**Chez le même éditeur**R. HERTZOG, R. MAS. **Debian Lenny. Gnu/Linux.** N°12433, 2009, 462 pages avec DVD.L. GODARD, B. MARCELLY. **Programmation OpenOffice.org 3 – Macros OOoBASIC et API.** N°12522, 2009, 920 pages.S. GAUTIER. **Memento Writer OpenOffice.org 3.** N°12533, à paraître 2009, 14 pages.I. HURBAIN, E. DREYFUS. **Memento Unix/Linux.** N°11954, 2006, 14 pages.S. BLONDEEL. **Wikipédia. Comprendre et participer.** N°11941, 2006, 218 pages.F. LE FESSANT. **Le peer to peer.** N°11731, 2006, 168 pages.F. DUMESNIL. **Les podcasts. Écouter, s'abonner et créer.** N°11724, 2006, 168 pages (collection « Connectez-moi ! »).B. BOUTHERIN, B. DELAUNAY. **Sécuriser un réseau Linux.** N°11960, 3^e édition, 2006, 266 pages.

Yoran **Brault**



Concevoir et déployer
ses sites web avec

Drupal

Préface de Edwy Plenel

Avec la contribution de Daniel Caillibaud

EYROLLES

ÉDITIONS EYROLLES
61, bd Saint-Germain
75240 Paris Cedex 05
www.editions-eyrolles.com



Le code de la propriété intellectuelle du 1^{er} juillet 1992 interdit en effet expressément la photocopie à usage collectif sans autorisation des ayants droit. Or, cette pratique s'est généralisée notamment dans les établissements d'enseignement, provoquant une baisse brutale des achats de livres, au point que la possibilité même pour les auteurs de créer des œuvres nouvelles et de les faire éditer correctement est aujourd'hui menacée.

En application de la loi du 11 mars 1957, il est interdit de reproduire intégralement ou partiellement le présent ouvrage, sur quelque support que ce soit, sans autorisation de l'éditeur ou du Centre Français d'Exploitation du Droit de Copie, 20, rue des Grands-Augustins, 75006 Paris.

© Groupe Eyrolles, 2009, ISBN : 978-2-212-12465-1

Préface

La révolution Drupal

Qu'y a-t-il de commun entre Mediapart, Rue89 et le gouvernement français ? Entre deux sites d'information indépendants et le portail officiel du pouvoir ? Entre la libre production de nouvelles indociles par les premiers et la diffusion contrôlée des communications ministérielles par le second ? Entre les deux *pure players* qui travaillent à inventer la presse numérique de demain et les dispositifs gouvernementaux qui accompagnent le bouillonnement démocratique du Net ? Juste un nom, qui est au ressort technologique et éditorial des uns et de l'autre : Drupal. Depuis juin 2009, sous un premier ministre, François Fillon, qui se revendique comme « un vrai geek », le portail du gouvernement a en effet migré de SPIP à Drupal, adoptant le système de gestion des contenus (CMS ou *Content Management System* en version originale) utilisé par Rue89 et Mediapart.

Cette unanimité technologique, par-delà des objectifs éditoriaux différents – en somme, information contre communication –, confirme la *success story* de Drupal depuis son invention tâtonnante, il y a moins de dix ans, par un informaticien belge, Dries Buytaert. Étudiant à l'Université d'Anvers, il en est venu à créer Drupal, dont la version 1.0 est sortie début 2001, à partir du bricolage en 1999 d'un système de messagerie interne pour sa résidence étudiante. Ce point de départ n'est pas indifférent, tant il confirme l'esprit originel de Drupal : loin d'une vision close de la technique, repliée sur elle-même et fermée aux non-initiés, la communauté d'utilisateurs, la facilité de ses usages et la commodité de ses pratiques sont le premier souci de Drupal, ce CMS que son inventeur définit comme un « assembleur rapide de site web » (*rapide website assembler*).

Dans l'univers du logiciel libre (*Open Source*), Drupal, modeste logiciel de forum à l'origine, est donc rapidement devenu l'un des systèmes de gestion de contenus les plus populaires. Aux États-Unis, en 2008 comme en 2007, il a été classé premier au concours du meilleur CMS dans la catégorie en accès libre. Avec plusieurs milliers de contributeurs de par le monde, sa communauté est de plus en plus active et inventive, à l'image de sa dernière conférence internationale tenue à Paris, en septembre 2009. C'est dire si ce précis pédagogique et pratique, sans équivalent jusqu'alors en langue française, tombe à pic alors que Drupal offre désormais une version très mature, la 6.13, et tandis que s'annonce déjà la très prometteuse version 7.

Le succès de Drupal s'explique d'abord parce qu'il répond à une attente vitale, tant elle est au cœur de la révolution démocratique dont le numérique est potentiellement le moteur, si nous nous en donnons les moyens et les instruments. À l'heure de l'avènement du média personnel, il s'agit de permettre à chaque lecteur d'être un auteur potentiel, de construire des agoras où s'élabore le savoir dans le croisement des compétences et des expériences, et de faciliter par conséquent un dialogue direct de qualité entre chaque auteur et ses lecteurs.

De ce point de vue, l'exposé méthodique de Yoran Brault a pour première qualité de ne jamais s'écarter de l'essentiel, autrement dit de ne pas perdre de vue les deux principes fondamentaux de Drupal : d'abord, mettre les contenus au centre – leur intégrité, leur conservation, leur amélioration, leur partage et leur circulation ; ensuite, se placer du point de vue de l'utilisateur, qu'il soit contributeur chevronné ou potentiel, de ses commodités d'accès et de ses facilités de participation.

Facilité, rapidité, simplicité, fiabilité sont les valeurs maîtresses – et l'on comprend aisément qu'elles sont philosophiquement démocratiques : pas d'aristocratie technique, pas de privilège savant, pas d'élite propriétaire. Aussi les conseils pratiques qui font le sel de ce manuel ont-ils l'immense intérêt de signaler les pièges qui peuvent en éloigner, qu'il s'agisse de fausses manœuvres ou de distorsions excessives (par exemple, les recommandations sur l'usage des caches). Car Drupal est évidemment un système complexe, et à ce titre suppose rigueur, précaution et doigté.

Tout informaticien, même amateur, n'a pas besoin d'avoir lu Edgar Morin, notre théoricien français de la « complexité », pour savoir d'expérience que complexe, ici, ne veut pas forcément dire compliqué, et peut donc s'appuyer sur des exposés simples et limpides. Mais, sous ce mot synonyme d'interdépendance et d'interaction, se cache une ambition extrêmement précieuse, dont l'actuelle révolution industrielle introduit les bouleversements dans notre votre quotidien, sans que nous les ayons encore accompagnés des révolutions démocratiques et intellectuelles qu'ils appellent.

C'est tout simplement la réalisation pratique de cet idéal pascalien, où la partie est indissociable du tout, et inversement le tout dans la partie, comme le définissait de façon imagée et poétique Pascal lui-même dans ses *Pensées* : « Toutes choses étant causées et causantes, aidées et aidantes, médiates et immédiates, et toutes s'entretenant par un lien naturel et insensible qui lie les plus éloignées et les plus différentes, je tiens impossible de connaître les parties sans le tout, non plus que de connaître le tout sans connaître particulièrement les parties. »

Tel est le défi intellectuel que nous lance Drupal, et l'exercice salvateur auquel il nous oblige. Ici, idéal technologique et idéal démocratique se rejoignent et s'imbriquent. Tandis que le système modulaire de Drupal oblige à penser en même temps le tout et les parties, leur logique et leur dialectique – leur « dialogique », dirait Morin –, la dimension participative des plates-formes de contenus qu'il permet de construire oblige à défaire l'ancienne étanchéité entre auteurs et lecteurs, journalistes et publics, spécialistes et citoyens, et par conséquent invite à repenser ensemble, dans la solidarité et dans la liberté, les rôles, statuts et actions des uns et des autres.

Dans le dernier tome de *La Méthode*, monument qui me semble être l'œuvre morale indissociable de la révolution numérique, tant elle la réfléchit et l'accompagne, Edgar Morin insiste sur l'auto-éthique, comme vertu individuelle et comme vertu sociale, qu'appelle cette complexité nouvelle des réseaux, de leur virtualité technologique et de leurs usages sociaux. « L'excès de complexité, écrit-il, détruit les contraintes, distend le lien social, et la complexité, à son extrême, se dissout dans le désordre. Dans ces conditions, la seule sauvegarde d'une très haute complexité se trouve dans la solidarité vécue, intériorisée en chacun des membres de la société. Une société de haute complexité devrait assurer sa cohésion non seulement par de justes lois, mais aussi par responsabilité, solidarité, intelligence, initiative, conscience de ses citoyens. Plus la société se complexifiera, plus la nécessité de l'auto-éthique s'imposera. »

Si je cite longuement cette mise en garde, c'est parce qu'elle me semble parfaitement refléter la démarche de Yoran Brault, son esprit et son exigence. La technique appelle une éthique. Et, par dessus tout, une éthique du savoir, entre précision et transmission, rigueur et partage.

Edwy Plenel

Table des matières

AVANT-PROPOS.....	1
Pourquoi ce livre ? • 2	
À qui s'adresse-t-il ? • 2	
Structure de l'ouvrage • 3	
Remerciements • 5	
1. RAPPELS UTILES SUR LES CMS.....	7
L'histoire du Web • 8	
Le fond et la forme • 9	
Taxonomie des contenus • 10	
Du contenu statique au contenu dynamique • 11	
Des droits et des rôles • 12	
Les commentaires • 13	
Un CMS, mais pour quoi faire ? • 14	
En résumé • 15	
2. DRUPAL, UN CMS EXTENSIBLE.....	17
L'architecture de Drupal • 18	
Un contenu au centre de tout • 20	
Les caractéristiques majeures de Drupal • 21	
La liberté des formats d'entrée • 21	
Un moteur de thème performant • 22	
Une taxonomie native • 22	
Internationalisation • 23	
Gestion des droits • 23	
Ouverture aux bases de données • 23	
Performances • 24	
Dimensionnabilité • 24	
La licence GPL v2 de Drupal • 25	
En résumé • 25	
3. BIEN DÉFINIR SON PROJET.....	27
Quel est votre objectif ? • 28	
De l'objectif au storyboard • 28	
Mettre en place le scénario (storyboard) • 29	
Se plonger brièvement dans les détails • 30	
Quels sont vos types de contenu ? • 31	
Les rôles de vos utilisateurs • 31	
Prévoir vos contraintes techniques • 32	
Le système d'information • 33	
Le contexte technique • 33	
Votre public et la fréquentation escomptée • 33	
En résumé • 35	
4. METTRE EN PLACE SON ENVIRONNEMENT DE DÉVELOPPEMENT.....	37
Les technologies mises en œuvre • 38	
Le service HTTP • 38	
Le langage PHP • 39	
La base de données • 39	
Le système d'exploitation • 40	
Création de la plate-forme de développement • 41	
Création d'un environnement WAMP • 41	
Installation de WampServer • 41	
Test de la plate-forme • 42	
Repérer les fichiers source • 44	
Création d'un environnement LAMP • 44	
Considérations sur les distributions • 44	
Installation d'Apache • 45	
Repérer les fichiers source • 46	
Installation de MySQL • 47	
Installation de phpMyAdmin • 48	
Téléchargement de Drupal • 49	
Site de téléchargement • 49	
Téléchargement de la traduction française de Drupal • 50	
Installation des sources de Drupal • 51	
Décompression des archives • 51	
Comprendre la structure des dossiers • 51	
Création du paramétrage initial • 52	
Création de la base de données • 53	
Initialisation de l'installation Drupal • 54	
Sélection de la langue • 54	
Configuration de la base de données • 55	
Configuration du site • 56	
Informations sur le site • 56	
Compte administrateur • 56	
Fuseau horaire • 57	
URL simplifiées • 58	
Notification des mises à jour • 58	

- Fin de l'installation • 58
- Résolution des problèmes techniques • 60
 - Problèmes de droits • 60
 - Mauvais fichier de configuration • 60
 - Impossible d'activer les URL simplifiées • 61
 - Problème de base de données • 62
 - Courrier de confirmation non reçu • 62
- En résumé • 63

5. ADMINISTRER DRUPAL ET SES MODULES..... 65

- Premier regard sur l'administration • 66
 - Accéder à l'administration • 66
 - La construction du site • 66
 - La configuration du site • 66
 - La gestion des utilisateurs • 67
 - La gestion du contenu • 67
 - Les rapports • 68
- Terminer l'installation de Drupal • 68
 - Utilisation du tableau de bord • 68
 - La tâche de maintenance cron • 69
- La gestion des modules • 70
 - Activer et désactiver un module • 70
 - Trier l'administration par modules • 72
 - Les modules standards • 73
 - Téléchargement d'un nouveau module • 76
 - Choisir vos modules • 77
 - Ai-je vraiment besoin de ce module ? • 77
 - Ce module est-il stable ? • 78
 - Ce module survivra-t-il ? • 78
 - Installer un module • 80
 - Désinstaller un module • 81
 - Soumettre une anomalie • 82
- Le paramétrage général de Drupal • 83
 - Boîte à outils image • 83
 - Date et Heure • 84
 - Informations • 85
 - Journalisation et alertes • 86
 - Rapports d'erreur • 86
 - Transfert de fichiers • 87
 - Système de fichiers • 88
- En résumé • 89

6. GÉRER LES UTILISATEURS..... 91

- Rôles, permissions et utilisateurs • 92
- Créer vos propres rôles • 93
- Gérer les permissions par rôle • 93
 - Permission du module Comment • 94
 - Permissions du module Node • 95

- Permissions du module Contact • 96
- Permissions du module Path • 96
- Permissions du module Search • 97
- Permissions du module Upload • 97
- Permissions du module User • 97

Paramétrage des comptes utilisateur • 98

- Qu'est-ce qu'un compte utilisateur ? • 98
- Choix du mode de création d'un compte utilisateur • 99
 - Mode Administrateur seul • 99
 - Mode Sans approbation • 99
 - Mode Avec approbation • 100
- Paramétrage des courriels liés aux étapes de création d'un compte • 101
- Signatures et portraits • 102

Gestion des utilisateurs • 104

- Création manuelle d'un utilisateur • 104
- Création d'un compte par l'utilisateur lui-même • 106
- Administrer les comptes utilisateur • 107

Changer rapidement d'utilisateur • 109

Règles d'accès • 111

La page Mon compte • 112

- Accéder à Mon compte • 113
- Modification du profil • 113
- Étendre le profil de l'utilisateur • 114

En résumé • 117

7. CRÉER DU CONTENU..... 119

Paramétrer les types de contenu • 120

- Les types de contenu • 120
- Modification du type de contenu Article • 120
 - Identification du contenu • 121
 - Les champs du formulaire de contribution • 121
 - Procédure de publication • 122
 - Paramètres des commentaires • 123
 - Mode de discussion • 123
 - Ordre des commentaires • 124
 - Pagination des commentaires • 124
 - Gestion des commentaires • 124
 - Commentaires des visiteurs anonymes • 125
 - Champ sujet d'un commentaire • 125
 - Gestion de l'aperçu • 125
 - Emplacement du formulaire de commentaires • 126

Création d'un contenu de type Article • 126

- Le rôle Contributeur • 126
- Formulaire de création d'un article • 127
- Le résumé d'un article • 128
- Décrire votre création • 129

Attacher un fichier • 129	Ajout d'un champ multiple • 185
Aperçu avant sauvegarde • 131	Ajout d'un type de champ • 186
Enregistrement de l'article • 131	Ordonnancement des champs • 187
Nœuds, chemin interne et alias • 132	Affichage des champs • 188
Administration des contenus • 134	Création d'un contenu Référence • 189
Publication de l'article • 134	Les limites de CCK • 190
Retour à la page d'accueil • 135	Créer des listes de contenus • 192
Supprimer un article • 137	Création des vues Articles et Billets • 192
Les commentaires • 137	Fonctionnement de Views • 194
Ajout d'un commentaire • 137	Construction de la liste • 194
Réponse aux commentaires • 139	Tester l'affichage • 197
Administration des commentaires • 140	Duplication de vues • 198
Les formats d'entrée • 142	Les performances de Views • 198
Fonctionnement d'un format d'entrée • 142	Étendre le modèle de droits • 199
Configuration d'un format d'entrée • 144	En résumé • 201
Droits sur un format d'entrée • 146	
Le format des commentaires • 147	9. STRUCTURER LES PAGES 203
Le format d'entrée PHP • 147	Gérer les blocs • 204
Ajouter de nouveaux filtres • 148	Le modèle de présentation • 204
Ajouter de nouveaux formats • 149	La liste des blocs • 205
Mettre en place un éditeur visuel • 150	Les blocs personnalisés • 207
En résumé • 157	Création d'un bloc personnalisé • 207
	Suppression d'un bloc personnalisé • 209
8. CRÉER DES CONTENUS AVANCÉS 159	Gérer la visibilité des blocs • 209
Création d'une FAQ • 160	Des blocs visibles en fonction du chemin • 209
Utiliser les livres • 160	Des blocs visibles par PHP • 213
Création de la FAQ • 161	Visibilité des blocs de modules • 214
Création des sections de FAQ • 162	Gérer les menus • 216
Organisation des pages de livre • 163	Structure des menus • 216
Gestion des révisions • 165	Gestion des chemins • 217
Consultation des révisions • 165	Création d'un élément de menu • 218
Administration des révisions • 166	Ajouter un bloc de menu • 220
Classer les contenus • 167	En résumé • 225
Qu'est-ce que la taxonomie ? • 167	
À quoi sert la taxonomie ? • 168	10. GÉRER LES LANGUES 227
La taxonomie et Drupal • 168	Traduction des contenus et traduction des interfaces • 228
Ajouter un nouveau vocabulaire • 169	Traduction des interfaces • 228
Ajouter des termes • 172	La liste des langues • 228
Associer des contenus à des termes • 173	Négociation de la langue • 229
Recherches par terme • 174	Réglage des préfixes • 230
Ajouter de nouveaux types de contenu • 178	URL spécifiques • 230
Ajout de types simples • 178	Changer de langue • 231
Ajout de types complexes • 180	Ajout d'une nouvelle langue • 232
Le module CCK • 180	Ajout d'une traduction • 234
Création du type de contenu Référence • 181	Traduction des contenus • 237
Traitement des champs Titre et Corps • 182	Page de garde multilingue • 240
Ajout des nouveaux champs • 183	En résumé • 241

11. MODIFIER L'APPARENCE 243

Qu'est-ce qu'un thème ? • 244

Un thème, ça change tout ! • 244

Thème et moteur de thème • 244

Développeur, Administrateur et Thèmeur • 245

Où allons-nous ? • 247

Gérer les thèmes • 247

L'administration des thèmes • 247

Paramétrer son thème • 249

Paramétrages par thème • 251

Paramétrages spécifiques d'un thème • 251

Paramétrage des blocs par thème • 253

Installer un nouveau thème • 254

Changer les pages d'erreur • 257

Régler le thème d'administration • 258

Créer son propre thème • 259

Qu'est-ce qu'un modèle ? • 259

L'imbrication des modèles • 261

Création d'un thème personnalisable • 262

Premier niveau de personnalisation : styles et scripts • 265

Utiliser les feuilles de style • 265

Utiliser les scripts • 268

Second niveau de personnalisation : les modèles • 270

Modifier le modèle de page • 270

Modifier le modèle des blocs • 274

Modifier le modèle des contenus • 275

Modifier le modèle des commentaires • 276

Troisième niveau de personnalisation : la fonction theme() • 278

Pourquoi aller plus loin ? • 278

Vision générale du système de thèmes de Drupal • 278

Utiliser les noms de modèles éligibles • 280

Noms de modèles éligibles pour le hook Page • 280

Noms de modèles éligibles pour le hook Node • 281

Noms de modèles éligibles pour le hook Block • 282

Modifier les modèles invisibles • 282

Personnaliser une fonction de thème • 285

Enrichir les variables d'un modèle • 289

Enrichir les variables de tous les modèles • 292

Le résultat • 292

En résumé • 293

12. AJOUTER DES FONCTIONNALITÉS 297

Ajouter la syndication à votre site • 298

Qu'est-ce qu'un flux RSS ? • 298

Les lecteurs de flux • 298

Flux des contenus • 300

Flux par taxonomie • 302

Flux des commentaires • 303

Créer des flux personnalisés • 303

Ajouter la notification sur les nouveaux commentaires • 305

Ajouter une lettre d'informations • 307

Prendre en charge l'OpenID • 313

Prendre en charge les Gravatars • 316

Ajouter un formulaire de contact • 318

Ajouter un module d'impression

et vos contenus en PDF • 321

Ajouter la notation de vos contenus • 325

Qu'est-ce qu'une notation de contenu ? • 325

Voting API et Fivestar • 325

Les contenus les mieux notés • 327

En résumé • 331

13. AUTOMATISER LES TÂCHES RÉCURRENTES 333

Créer des flux de travaux (workflows) • 334

Actions et déclencheurs • 340

Principe de fonctionnement • 340

Création d'une action simple • 340

Création d'une action avancée • 343

En résumé • 345

14. ÉCRIRE SON PREMIER MODULE 347

Prérequis • 348

Pourquoi faire son propre module ? • 348

Des modules et des hooks • 350

Structure interne d'un module • 352

Organisation générale • 352

Le fichier de description du module • 352

Le code du module • 353

Implémentation d'un hook • 353

Activation du module • 355

En résumé • 355

15. METTRE SON SITE EN PRODUCTION 357

Mettre à jour son installation Drupal • 358

Pourquoi parler de cela maintenant ? • 358

Doit-on toujours tout mettre à jour ? • 358

Le tableau de bord des mises à jour • 359

Mise à jour du code source • 361

Mise à jour de Drupal • 361

Mise à jour d'un module • 363

Mise à jour d'un thème • 363

Mise à jour de la base de données • 363

Sauvegarde de la base de données • 363

Le script update.php • 364

Si tout va mal... • 366

Avez vous bien testé ? • 367	
Les tests fonctionnels • 367	
Les tests techniques • 367	
Validez ! • 368	
De l'intégration à la production • 369	
Choisir son hébergement • 369	
Acheter son nom de domaine • 370	
Le premier déploiement • 371	
Copier les sources du site • 371	
Mettre en place la base de données • 373	
Les déploiements suivants • 375	
Performances • 377	
Vision générale des performances • 377	
Les performances de Drupal • 378	
Améliorer les performances du serveur • 380	
Gestion des caches • 380	
Paramétrage du cache • 382	
Accélérateurs de code PHP • 383	
Le cache de Drupal en mémoire • 384	
Accélérer le journal • 385	
Améliorer les performances du navigateur • 385	
Compression des pages • 385	
Optimisation des feuilles de style • 385	
Optimisation des scripts • 386	
Vidange des caches • 387	
Cron • 388	
Gérer le multisite • 389	
En résumé • 391	
16. UN REGARD SUR DRUPAL 7 393	
Amélioration de l'ergonomie d'utilisation • 394	
Intégration de CCK • 395	
Un nouveau système de base de données • 396	
Autres évolutions notables • 397	
Amélioration des performances • 397	
L'ouverture du moteur de recherche • 397	
Une refonte des droits sur les contenus • 397	
En résumé • 398	
INDEX 399	

Avant-propos

L'information écrite a toujours circulé d'un auteur vers ses lecteurs. À son origine, le Web ne fut qu'un média supplémentaire ajouté à la longue descendance de l'imprimerie de Gutenberg. Mais l'expansion de cet outil à partir des années 2000 l'a très vite conduit à dépasser le cadre des médias traditionnels, par la création de nouvelles possibilités, inexistantes jusqu'alors : pour la première fois, chaque lecteur est aussi un auteur potentiel, et un dialogue direct interactif peut s'instaurer entre un auteur et son lectorat. Cette première phase de ce qu'on appellera plus tard le Web 2.0 n'a cessé par la suite de s'étendre à de nouveaux domaines d'utilisation, dont le plus marquant reste celui des réseaux sociaux. C'est pour prendre en charge ces contenus de plus en plus foisonnants que sont apparus les CMS (*Content Management System*) ou systèmes de gestion des contenus.

Ces outils succédèrent aux technologies traditionnelles HTML ou PHP qui demandaient, pour créer une page web, des compétences techniques pas forcément à la portée de toute personne devant publier sur l'Internet. Grâce aux CMS, les rédacteurs disposent désormais d'un outil « clés en main », leur permettant de s'affranchir des contraintes techniques pour publier leurs contenus en temps voulu, sans intervention systématique d'un informaticien. Qui plus est, de tels outils facilitent le travail collaboratif, ainsi que la gestion des contenus provenant de l'interaction avec les lecteurs (commentaires, etc.). Ainsi le CMS est-il devenu l'outil le plus indiqué pour créer rapidement des sites web de tous usages, du journal en ligne au site institutionnel, en passant par le blog ou l'intranet.

Parmi tous les CMS disponibles sur le marché, il en est un qui se démarque dans la foule : Drupal. Défini par son créateur comme un « assembleur rapide de site web », Drupal est le système de gestion de contenus le plus souple qui soit. Construit sous forme modulaire autour d'un noyau de très petite taille, il délègue en effet chaque fonctionnalité à un module séparé, ce qui signifie, en théorie, que ce logiciel peut être étendu à l'infini. Ainsi, entre outil prêt à l'emploi et plate-forme de développement, il permet de gérer la quasi-totalité des sites web, quel que soit leur type.

Enfin, pour ne rien gâcher, Drupal est un logiciel libre, tout comme ses modules. Il bénéficie du travail de toute une communauté de développeurs, traducteurs, utilisateurs et rédacteurs de documentation, qui contribuent tous à améliorer encore et toujours le logiciel. Ainsi, outre la gratuité, il propose un outil adaptable à tous les besoins, puisque tout un chacun peut modifier son code source pour obtenir un logiciel sur mesure.

Pourquoi ce livre ?

Ce livre a pour objectif de vous apprendre à concevoir, à créer puis à déployer un site web avec Drupal. De l'installation et du paramétrage à la mise en production, sans oublier la méthodologie, tous les aspects sont abordés pour vous permettre de maîtriser l'outil au sein de son contexte – le Web d'aujourd'hui.

Il ne s'adresse cependant pas aux développeurs connaissant déjà bien Drupal et désireux d'étendre son fonctionnement à travers la réalisation de modules. En effet, il ne s'agit pas d'un manuel de développement, mais d'un guide pratique destiné à vous accompagner, quel que soit votre niveau, vers la production d'un site moderne et fonctionnel.

À qui s'adresse-t-il ?

Cet ouvrage s'adresse donc à toute personne cherchant à découvrir ou à approfondir sa connaissance du CMS Drupal :

- le décideur, ou encore le chef de projet, qui aura à gérer un chantier sous Drupal, et qui doit comprendre ce qu'est un CMS, ce qu'est Drupal et comment il fonctionne ;
- le développeur PHP qui doit construire un site avec Drupal et qui trouvera ici les outils et les techniques lui permettant de réaliser sa tâche ;
- toute personne qui cherche à comprendre le fonctionnement interne de Drupal pour construire son propre site.

Structure de l'ouvrage

Le **chapitre 1** de ce livre explique ce qu'est un CMS (*Content Management System*), son origine, sa philosophie et son usage.

Le **chapitre 2** est une transposition de ce que nous aurons appris des CMS sur le système Drupal, en détaillant autant que possible ses lacunes et ses points forts.

Le **chapitre 3** traite de la définition de notre besoin. Il s'agit d'une étape purement méthodologique, dont l'objectif est d'aboutir à une vision claire de ce que nous allons réaliser par la suite. À l'issue de ce travail, nous ne devrions plus nous poser de questions sur le « quoi », afin de nous concentrer pleinement sur le « comment ».

Le **chapitre 4** détaille la phase indispensable de création d'une plateforme de développement dédiée à Drupal. Celle-ci va nous permettre d'expérimenter les notions abordées dans les chapitres suivants, de manière simple et sans risque. Les étapes d'installation d'Apache, MySQL et Drupal y sont détaillées pour les environnements Windows et GNU/Linux.

Le **chapitre 5** est notre premier contact avec Drupal, à travers son système d'administration. Ce chapitre se conclut par l'installation de notre premier module.

Le **chapitre 6** nous permet ensuite d'aller plus en profondeur dans l'administration, pour apprendre à gérer une notion très importante dans un CMS : les utilisateurs. Nous découvrirons ici tous les aspects de la création de comptes utilisateur, ainsi que la mise en place de rôles permettant de donner à chacun des droits particuliers.

Dans le **chapitre 7**, nous arrivons au cœur de ce qu'est un CMS : la création de contenu. Nous commencerons par aborder la notion de *type de contenu*, pour ensuite créer notre premier article et en gérer tous les aspects : modification, commentaires, etc.

Le **chapitre 8** nous emmène un cran plus loin dans le domaine de la création de contenu, en abordant des notions plus complexes de taxonomie, de contenus hiérarchisés et de gestion des historiques de modification. Nous y apprendrons surtout à créer nos propres types de contenu évolués grâce au module CCK, et à les exploiter avec le module Views.

Avec le **chapitre 9**, nous sortons du domaine du contenu pour passer dans celui du contenant, et plus particulièrement de la structuration des informations disponibles sur chaque page de notre site. Nous découvrirons ici comment Drupal se révèle être un puissant outil de présentation, capable d'afficher de manière planifiée n'importe quel type d'information sur n'importe quelle page.

Le **chapitre 10** se place à la croisée des chemins entre contenu et contenant, pour nous permettre de rendre notre site multilingue. Nous étudierons ici comment décliner notre site en plusieurs langues, tant au niveau de l'interface que des contenus eux-mêmes.

Notre site étant maintenant structuré, le **chapitre 11** nous fait entrer dans le domaine du graphisme et des couleurs, en nous permettant de transformer radicalement l'apparence de notre site, de la simple configuration des thèmes existants jusqu'aux techniques de réalisation d'un thème totalement personnalisé.

Le **chapitre 12** traite de l'ajout d'outils transversaux à Drupal. Notre site étant à présent joliment présenté et capable d'intégrer tous les contenus que nous souhaitons, il ne reste donc plus qu'à y ajouter les dizaines de fonctions qu'un visiteur est en droit d'attendre d'un site moderne : PDF, Gravatar, notifications, OpenID, bulletin d'informations (newsletter), etc.

Avec le **chapitre 13**, nous quittons le domaine du contenu pour entrer dans celui de la maintenance d'un site Drupal, en examinant plus particulièrement les possibilités d'automatisation des tâches courantes. Nous y aborderons notamment le travail en équipe à travers la réalisation de flux de travaux (*workflows*).

Le **chapitre 14** offre une ouverture vers d'autres possibilités à travers la réalisation de notre premier module pour Drupal. Un module simple et facile à mettre en œuvre, qui nous permettra de comprendre les mécanismes fondamentaux de Drupal.

Le **chapitre 15** est l'aboutissement de notre travail. Notre site est prêt, il a été testé, et nous apprendrons comment mettre à jour chacun de ses composants pour finalement l'amener à la vie sur un serveur de production. À l'issue de ce chapitre, notre site sera en ligne, prêt à recevoir le succès qu'il mérite.

Le **chapitre 16** est une conclusion en forme d'ouverture à ce que nous promet la prochaine version de Drupal, la version 7.

Remerciements

Je remercie chaleureusement Daniel (Caillibaud) et Régis (Passeron) pour le temps qu'ils ont passé à me relire, et à me relire encore, jusqu'à ce qu'ils finissent par comprendre ce que je cherchais maladroitement à expliquer, pour m'indiquer ensuite, avec beaucoup de tact, ce qu'il me manquait pour être compréhensible.

Merci également à l'équipe des éditions Eyrolles et Muriel Shan Sei Fan, leur editrice, pour son énergie injectée par ligne téléphonique, ce qui n'est pas chose aisée.

Enfin, plus qu'un remerciement, une dédicace... Une dédicace à une Grenouille Qui Parle, à toi Julie, qui m'a botté le derrière pour que j'accepte d'écrire ce livre, et l'a payé cher en de nombreuses soirées passées à décrypter mon charabia pour en faire la prose lisible que vous avez entre les mains. Ceci, comme beaucoup d'autres choses depuis vingt ans, n'aurait pas été possible sans toi à mes côtés.

chapitre 1



Rappels utiles sur les CMS

Les CMS ont révolutionné l'approche de publication sur le Web, s'adaptant en permanence aux besoins créés par l'instantanéité et l'interactivité du Web dit 2.0. Avant de commencer avec Drupal, il est bon de revenir sur les principes de base qui forment un CMS : séparation du fond et de la forme, classification des contenus, gestion des contenus dynamiques et des commentaires, droits et rôles des utilisateurs, etc.

SOMMAIRE

- ▶ Qu'est-ce qu'un CMS ?
- ▶ Qu'apporte un CMS comparé à un site traditionnel ?
- ▶ La séparation fond/présentation
- ▶ Que puis-je faire d'un CMS ?

L'histoire du Web

Le Web n'a pas toujours été un endroit des plus conviviaux, loin de là. Cinq ans après sa naissance en 1990, la Toile était essentiellement composée de volumineux ensembles de fichiers HTML, pimentés de scripts rédigés dans quelques langues absconses. En 1995, l'émergence du couple PHP/MySQL a donné naissance à une seconde génération de sites, certes beaucoup plus riches et dynamiques, mais qui n'en restaient pas moins des machineries construites sur mesure et gardées bien hors de portée des non-spécialistes.

CULTURE La première page web

C'est en 1989 que Tim Berners-Lee¹ alors employé au CERN², propose à la communauté scientifique une approche novatrice du partage de l'information visant à relier entre elles des sources de données distribuées sur l'Internet de manière arbitraire et non hiérarchisée. Le Web et ses hyperliens étaient nés.

Les hyperliens en eux-mêmes ne sont pas une invention de Berners-Lee mais de Ted Nelson, qui en 1965 cherchait lui aussi un moyen non linéaire d'organiser l'information. Pour la petite histoire, le brevet a navigué de société et société jusqu'à aboutir dans le portefeuille de British Telecom qui chercha même à faire un procès pour violation de brevet (!) aux fournisseurs d'accès américains en 2002³.

En 1990, Robert Cailliau arrive dans le projet et collabore à la proposition définitive du « WorldWideWeb: Proposal for a HyperText Project ». Les bases du Web avec des concepts aujourd'hui aussi basiques qu'URL, HTML, ou HTTP sont inventés.

L'équipe du CERN va ensuite écrire le premier service HTTP (poétiquement nommé `nxoc01.cern.ch`) et le navigateur associé, le `WorldWideWeb`, ensuite rebaptisé `Nexus`. La toute première page HTML est finalement mise en ligne le 13 novembre 1990⁴, et en 1991, le projet devient mondial avec la mise en place d'un second serveur au SLAC (accélérateur de Stanford). Un an plus tard, ce sont vingt-six sites qui sont en ligne, en partie grâce à l'arrivée du célèbre service HTTP NCSA.

1. http://fr.wikipedia.org/wiki/Tim_Berners-Lee

2. <http://fr.wikipedia.org/wiki/CERN>

3. <http://www.foruminternet.org/specialistes/veille-juridique/actualites/british-telecom-revendique-des-droits-sur-l-hyperlien.html>

4. <http://www.w3.org/History/19921103-hypertext/hypertext/WWW/Link.html>

À cette époque, le processus *classique* pour modifier *son* site consistait à saisir les textes dans son coin, pour les transmettre ensuite à des développeurs qui se chargeaient, en quelques jours, de mettre le tout en ligne... Et s'il s'agissait de modifier un aspect plus visuel, les jours se transformaient vite en semaines.

Ce mode de travail fonctionna pendant un temps, mais le succès du Web et la complexité des sites allant grandissante, deux problèmes commencent à se poser de manière récurrente :

- Comment modifier la forme d'un site sans systématiquement remettre en cause sa conception ?
- Comment modifier le fond d'un site sans systématiquement passer par des pro du Web ?

C'est dans ce contexte qu'ont été jetées à la fin des années 1980 les premières bases d'un système de gestion des contenus (en anglais CMS pour *Content Management System*). À cette époque, les CMS étaient autant une théorie qu'une série de bonnes pratiques apparues au fil du temps, que les concepteurs d'outils tels que Typo3 tentèrent de mettre en cohérence.

Dans la suite de ce chapitre, nous allons étudier les concepts clés communs à tous les CMS, sans nous attacher à une quelconque implémentation. C'est une étape probablement fastidieuse pour ceux qui sont déjà rompus à ce type d'outillage, aussi peuvent-ils se rendre directement au chapitre suivant. Il est cependant important de poser le plus clairement possible les bases qui nous serviront tout au long de cet ouvrage :

- Qu'est-ce qu'un contenu ?
- Comment et pourquoi est-il séparé de sa forme visuelle ?
- Comment ces contenus sont-ils classés ?
- Quels processus permettent de les modifier ?

Le fond et la forme

Avant les CMS, l'apparence d'une page et ce qu'elle racontait étaient deux choses intimement liées. Ainsi, changer la couleur du titre ou changer le texte du titre était une seule et même opération car tout était confiné en un seul et même document : le fichier HTML ou PHP.

Le principe fondateur des CMS est de considérer qu'un document doit être séparé avec d'un côté un *contenu*, et de l'autre une *présentation de ce contenu*. Pour bien comprendre ces deux concepts, prenons l'exemple d'un article de votre quotidien préféré. Lorsque vous achetez ce journal en kiosque, vous avez sous les yeux une version papier de cet article. Il s'agit à l'origine d'un simple texte fourni par le journaliste qui a été mis en page par toute une équipe de manière à rendre sa lecture la plus agréable possible. Si vous allez sur le site web de ce même journal, vous y retrouverez l'article mais présenté cette fois-ci de sorte que sa lecture soit adaptée à un écran. Les deux articles ne se ressemblent pas dans la forme, mais sont bien issus du même texte écrit par le même journaliste.

Dans le vocabulaire CMS, le journaliste a rédigé un contenu qui a été publié sous deux présentations distinctes : papier et Web.

Pour aller un peu plus loin, un CMS impose une vision structurée de ce contenu. Car ce n'est pas tout de dire qu'un contenu est un simple texte, encore faut-il que le CMS puisse identifier sa structure : il considèrera ainsi notre article comme un assemblage d'éléments tels que titre, sous-titre, illustration, auteur, etc. Cette vision lui permet dès lors d'appliquer une présentation spécifique à chaque élément.

Pour synthétiser ces concepts clefs, nous pouvons donner les définitions suivantes :

- Un *contenu* est un ensemble structuré d'informations (titre, auteur, etc.) sans aucune notion de mise en page ou de mise en forme.
- La *présentation d'un contenu* applique un style visuel à chaque élément du contenu de sorte à en obtenir une version publiable. Un même contenu peut ainsi avoir plusieurs présentations, généralement adaptées à un média donné (papier, Web, etc.).

Sachant cela, il semble évident que selon la nature de notre contenu – article de fond ou simple dépêche –, ce dernier ne contiendra pas les mêmes informations de base. Une dépêche contiendra par exemple une « date de la dépêche » qui n'a pas lieu d'être dans un article de fond. De même, un article de fond contiendra un auteur qui n'aurait que peu de sens dans une dépêche. C'est en réponse à ce constat que le CMS introduit la notion de *type de contenu*.

Un type de contenu définit la liste des informations qui constituent le contenu. Chaque information est caractérisée par son nom (ex. illustration ou encore titre) et son type (ex. image ou texte).

Taxonomie des contenus

Une fois tous ces contenus créés, tout le jeu consiste à pouvoir les retrouver facilement. Tout CMS digne de ce nom intègre le concept de recherche dite *plein texte*, qui fonctionne peu ou prou comme les moteurs de recherche sur l'Internet. Mais cette approche est loin d'être satisfaisante si l'on désire obtenir par exemple l'ensemble des documents traitant de « système d'exploitation ».

Le second concept introduit par les CMS vous permet donc de « ranger » vos contenus par mots-clés libres ou par taxonomie. Trop souvent confondu avec l'idée un peu simple de *catégories*, la taxonomie est en réalité un ensemble de mots-clés (appelés *taxons*) appartenant à un même domaine et reliés entre eux par un lien de hiérarchie.

Dans l'exemple donné ci-dessous, nous avons une (petite) taxonomie des domaines de compétence d'une société de conseil en management. Ainsi, si nous rattachons un contenu au taxon Recrutement, il appartiendra de facto au taxon Ressources Humaines et Domaines de compétence et ressortira du lot si l'on fait une recherche sur l'un de ces trois termes.

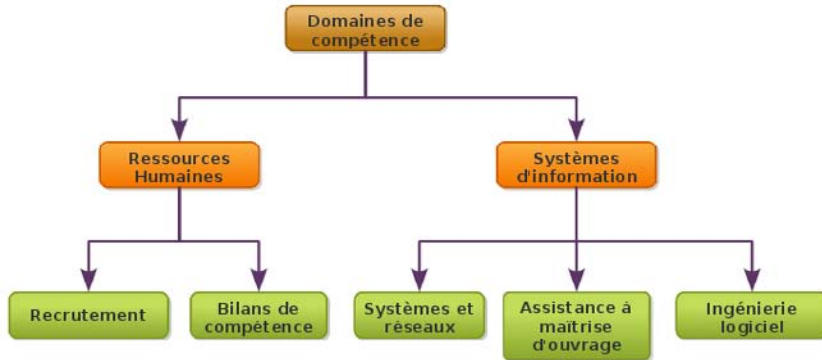


Figure 1-1
Un exemple de taxonomie

Tous ces critères de recherche sont bien sûr combinables, permettant par exemple d'obtenir la liste de tous les contenus de type Article écrits par l'utilisateur Marine, après le 15 janvier 2007 et traitant de Système et réseaux.

Du contenu statique au contenu dynamique

Au cours des deux premières générations de sites web, le changement de la moindre virgule sur la page d'un site vous obligeait à recourir à l'intervention d'un spécialiste. C'était un processus long et frustrant pour les personnes en charge d'écrire les contenus.

CULTURE D'où vient le terme Web 2.0 ?

Le terme « Web 2.0 » est à mettre au crédit de Dale Dougherty de la société O'Reilly Media¹. Lors d'une conférence en 2004, Dale cherche par cette numérotation a posteriori (il n'y a jamais eu de numérotation des versions du Web) à faire prendre conscience à son auditoire de la différence d'approche fondamentale entre le Web d'avant l'explosion de la bulle Internet et le nouveau Web. Une différence qui trouve son point de départ concret en 2001, au lendemain de la crise de l'Internet, avec la naissance de Wikipedia et la prise de pouvoir des utilisateurs qui s'en suivit.

Avec Wikipedia, pour la première fois le contenu « monte ». Les documents ne sont pas écrits par des « têtes connues et reconnues », mais par les internautes eux-mêmes, chacun avec leur savoir, leurs compétences, dans un mode purement collaboratif et ouvert. Pour caractériser cette (r)évolution, le magazine Wired va même jusqu'à inventer un nouveau terme : le « crowdsourcing », littéralement la délégation de la construction du contenu aux internautes devenus d'un coup, pour la première fois, des « utilisateurs ».

1. <http://www.eutech-ssii.com/ressources/1>

Les CMS ont radicalement changé cette approche en permettant aux utilisateurs de modifier dynamiquement leurs contenus et surtout de publier directement ceux-ci sur le Web.

Le CMS vous permet d'un simple clic d'éditer le contenu qui se cache derrière la page que vous êtes en train de visualiser. Dès que vous validez vos modifications, le contenu dans sa nouvelle présentation vous est alors renvoyé.



Figure 1–2

La base d'un CMS, le cycle de vie d'un contenu

Lorsque ce contenu vous semble correct, il est alors possible de le publier sur l'Internet d'un simple clic.

Cette approche simple et intuitive de modification du contenu est un changement significatif dans le rapport entre l'auteur et le média. Enfin déchargé de la relation utilisateur/technicien, il est alors libre de produire à son rythme. Cette mutation a eu pour résultat l'apparition d'une troisième classe d'acteurs du Web : les *contributeurs*. Cette nouvelle catégorie d'intervenants a notamment fait le succès d'un CMS mondialement connu, Wikipédia, et donc du Web 2.0.

Des droits et des rôles

Cependant, cette nouvelle classe d'acteur amène aussi de nouveaux besoins en matière de gestion des droits. Il est en effet rarement souhaitable de laisser à n'importe quel visiteur la possibilité de modifier le contenu d'un site. Les CMS disposent donc d'une approche complète de la gestion des droits, combinant permissions et rôles.

- Une *permission* est un droit simple portant sur une opération du CMS.
- Un *rôle* est quant à lui un ensemble nommé de permissions.

Par exemple, le rôle nommé Écrivain contiendra la permission « créer un nouveau contenu de type article » mais pas forcément la permission « rendre un contenu visible ». Un rôle nommé Correcteur, quant à lui, ne

possédera pas « créer un contenu » mais seulement « modifier un contenu existant », etc.

Une fois ces rôles définis, le CMS permet de les attribuer à des utilisateurs qui devront s'identifier pour en bénéficier.

Pour compléter cette approche collaborative de création de contenu, un CMS autorise la définition d'états par type de contenu, permettant ainsi la mise en place de flux de travaux (en anglais, *workflow*). Un contenu peut ainsi passer de l'état Nouveau à l'état En attente de correction, puis Corrigé, puis Publié. Il est dès lors possible de définir des rôles ayant la permission de faire passer un contenu d'un état à un autre.

Les CMS les plus aboutis protègent leur contenu de toute altération accidentelle ou maligne en mettant en place un système d'historique, chaque modification du contenu induisant la création d'une nouvelle version. Il sera ainsi possible de remonter dans le temps pour revenir à une version antérieure.

Les commentaires

Les CMS proposent une série de services liés au contenu. La majorité d'entre eux consiste à prévenir l'utilisateur d'une nouveauté sur le site par courriel, ou encore par flux RSS.

CULTURE Les commentaires et le Web 2.0

Le *crowdsourcing*, animé par des utilisateurs disposant d'une « identité numérique », carburant du phénomène Wikipédia, est devenu réellement populaire avec les fameux *commentaires*. Les utilisateurs parlent aux utilisateurs, fournissant ainsi de manière incidente aux propriétaires d'un site, un contenu volumineux et surtout... gratuit. Les utilisateurs s'expriment, communiquent et forment ainsi le troisième et sûrement le plus important concept clé du Web 2.0 : les *réseaux sociaux*.

Cette dénomination un tantinet ronflante de réseau social décrit néanmoins assez bien une nouvelle forme de communication transversale. Non seulement les utilisateurs ne reçoivent plus l'information d'en haut, mais en plus ils la produisent et la partagent. Le circuit de l'information devient ainsi énormément plus complexe que pour n'importe quel média préexistant : verticalement (*bottom up*) et horizontalement. Et un exemple plus qu'évocateur de ces réseaux sociaux, de cette communication transversale : les blogs.

Parmi ces services satellites, la possibilité laissée aux visiteurs de publier leurs commentaires est de loin la plus révolutionnaire.

En effet, dans les premières périodes du Web, l'interaction auteur/lecteur fonctionnait comme pour les médias traditionnels. La communication émanait toujours d'une source unique vers de multiples destinataires : du journaliste vers les lecteurs, du webmestre vers les visiteurs, etc.

Avec ses commentaires, le lecteur emprunte un canal de communication inédit en direction de l'auteur, c'est-à-dire de la source de l'information. Par une démarche censée lui permettre uniquement d'exprimer son opinion, il vient en réalité enrichir le contenu initial et par conséquent, nourrir son auteur.

Pris dans la théorie des CMS, les commentaires ne sont finalement qu'une manière de modifier dynamiquement un contenu parent en venant s'y accrocher. Ces commentaires sont donc des contenus à part entière qu'il est possible d'exploiter comme n'importe quel autre type de contenu. Ils auront donc eux aussi un fond (le texte du commentaire) et une forme (leur apparence visuelle), la possibilité d'être édité dynamiquement et d'apparaître sous différentes formes.

Un CMS, mais pour quoi faire ?

Les CMS résultent de l'expérience cumulée des sites de précédentes générations. Il n'est donc pas surprenant qu'ils apparaissent aujourd'hui comme des outils de création rapide de sites, convenant à peu près à tous les usages.

- Un journal en ligne, car le CMS dispose de tous les outils pour permettre la mise en ligne d'une activité journalistique classique : ce type de site exploitera particulièrement la taxonomie, la multiplication des vues par contenu, la création de types de contenu spécifiques (dépêches, articles, dossier, etc.).
- Le blog et le site personnel sont eux aussi de bons candidats pour les CMS, dont ils utilisent la simplicité de mise en œuvre, la rapidité d'édition et de mise en ligne et les divers services permettant la syndication.
- Les forums utilisent depuis longtemps des CMS spécialisés (ex. phpBB), mais les CMS plus généralistes se prêtent également bien à l'exercice.
- Les sites Intranet exploitent massivement les outils de collaboration offerts par les CMS.
- Les sites institutionnels reprennent nombre d'avantages des CMS, avec un accent particulier sur la séparation fond/forme, permettant

une mise en œuvre ou un changement rapide d'une charte graphique, sans mettre en péril la structure et le contenu.

Ce ne sont là que quelques exemples, car en réalité, il serait plus simple d'établir la liste des sites qui n'ont rien à gagner à utiliser un CMS. Généralement, il s'agit plus d'applications que de sites web. Par exemple, si vous cherchez à créer pour votre société un service de messagerie avec une interface accessible par le Web, l'utilisation d'un CMS ne présente que peu d'intérêt.

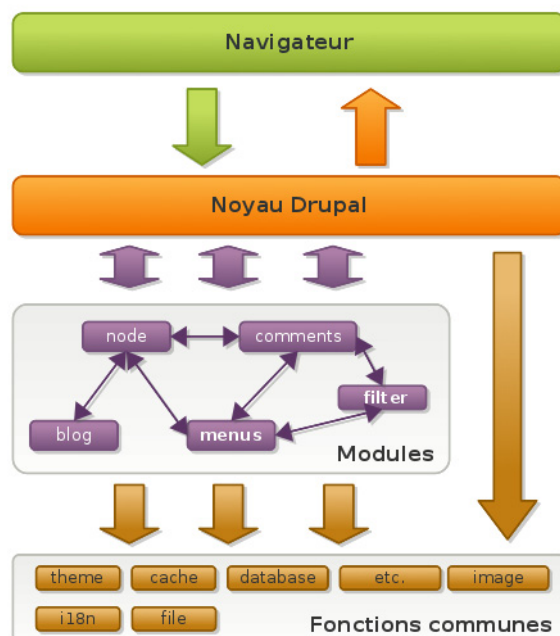
La diversité des CMS existant sur le marché est bien à l'image de cette quasi-universalité d'usage. À titre d'exemple, si l'on se limite aux logiciels libres toutes plate-formes confondues, il existe plus d'une centaine de CMS dont 80 en PHP. Ce nombre tombe à 35 pour les CMS dits généralistes, c'est-à-dire ceux qui ne sont pas conçus spécifiquement pour des forums, des annuaires, du réseau social, du *groupware*, etc.

Mais si vous lisez ce livre, c'est que vous avez déjà une petite idée de celui qui vous intéresse, et nous allons maintenant voir en détail ce que Drupal – ce CMS hors normes – a de si particulier.

En résumé

- CMS veut dire *Content Management System*. En français, cela se traduit par système de gestion des contenus ou SGC.
- Pour un CMS, tout est contenu : article, page, commentaire, etc.
- Un CMS permet de créer tout type de site web dès lors qu'il y a manipulation de contenu (blog, catalogue de produits, etc.)
- Un CMS permet de faire une séparation stricte entre les informations d'un contenu et leur présentation visuelle.
- Un CMS permet un travail collaboratif sur le contenu qu'il gère.

chapitre 2



Drupal, un CMS extensible

Drupal n'est pas un CMS comme les autres. Partant d'un système somme toute traditionnel, il n'a cessé d'évoluer pendant près de dix ans, avec pour seule idée de ne jamais fermer aucune possibilité tout en recherchant une efficacité maximale. De ce paradoxe a découlé une architecture atypique et performante, à mi-chemin entre le logiciel « clés en main » et la plate-forme de développement, permettant ainsi de construire tout type de site web, depuis le simple blog personnel jusqu'au plus moderne des réseaux sociaux.

SOMMAIRE

- ▶ L'architecture atypique de Drupal
- ▶ Comment Drupal interprète les fondamentaux d'un CMS
- ▶ Comment Drupal utilise les bases de données
- ▶ Les performances et la « dimensionnabilité » de Drupal
- ▶ La licence de Drupal

L'architecture de Drupal

Drupal fait partie de la famille des CMS libres à vocation généraliste écrits en PHP. Mais présenté ainsi, il n'est au fond qu'un parmi trente-cinq de ses frères et sœurs. Ce qui amène inévitablement à la question suivante : *Qu'a donc Drupal de si particulier ?*

Dries, l'initiateur de ce logiciel, définit Drupal comme un « assembleur rapide de site web ». Cette image peut sembler un peu rapide, mais définit cependant sa capacité à créer rapidement des sites très variés (portail, blog, site institutionnel) tout en ne faisant – et c'est important de le préciser – que bien peu de concessions par rapport aux des besoins d'origine. Le secret de cette souplesse réside dans le mot « assemblage » donné dans l'expression de Dries, qui nous renvoie directement à l'architecture ultra-modulaire de Drupal.

CULTURE L'histoire de Drupal

L'histoire de Drupal commence en 1999 lorsque Dries Buytaert, alors étudiant à l'université d'Anvers, cherche à créer un réseau sans fil pour partager sa connexion ADSL (un luxe pour l'époque) avec ses compagnons de chambrée. Vint alors le besoin d'un site web pour partager le statut de ce réseau.

Lorsque Dries est diplômé, le site est transféré sur l'Internet sous le nom `drop.org`. Ce drôle de nom est simplement le fruit d'une erreur de frappe de Dries qui voulait en fait acheter le domaine `dorp.org` (« village » en néerlandais). L'erreur ne posa pas de problème à la communauté de passionnés des technologies du Web qui commençait à se créer autour de ce site pour l'expérimenter et le faire évoluer. Il y travaillèrent jusqu'à ce que les sources soient libérées en 2001 sous le nom de « Drupal » (le mot « drop », en néerlandais, se prononçant « druppel », était trop compliqué à prononcer pour des anglophones).

En octobre 2004, la version 4.5 de Drupal est livrée et forme les contours « définitifs » de ce que nous en connaissons aujourd'hui.

Modularité ne signifie pas simplement extensibilité. La majorité des CMS modernes sont extensibles. Généralement formés d'un corps dense intégrant la gestion du contenu, des commentaires, du rendu visuel, des flux, etc., il est toujours possible de leur adjoindre quelques greffons (ou *plug-ins*, ou encore composants) dont le but sera d'amener telle ou telle nouvelle fonctionnalité. Dans cette approche, le CMS, qui forme un ensemble monolithique, ne peut être étendu que par des « ouvertures » qui auront été préalablement pratiquées.

Il est donc délicat d'étendre ce genre de produit de manière organique. En effet, que se passe-t-il lorsqu'un développeur veut changer un aspect auquel ne correspond aucune ouverture ? Il attend généralement avec impatience que sorte une nouvelle version du produit, disposant des nouvelles portes dont il a besoin. Ou pire, il perce lui même ses propres portes, fragilisant ainsi dangereusement l'édifice.

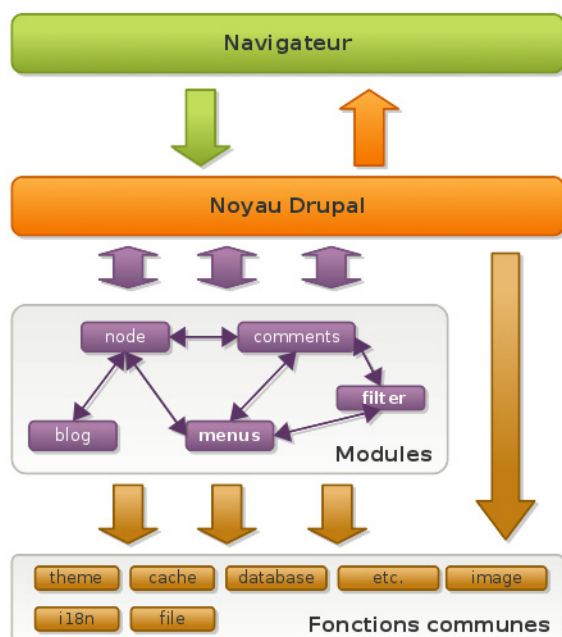


Figure 2-1
L'architecture de Drupal

Dès son origine, Drupal a pris une direction radicalement différente. Son architecture interne consiste en un noyau de très petite taille, ne contenant que les fonctionnalités essentielles (gestion des requêtes HTTP, du cache, etc.), et déléguant l'ensemble des traitements fondamentaux à des modules. Les modules vont à leur tour déléguer certains traitements à d'autres modules, et ainsi de suite jusqu'à remonter un résultat au noyau. Pour les aider dans cette tâche, Drupal leur fournit dans ce noyau une bibliothèque très complète de fonctions regroupées par thématique appelées API (FormAPI, SchemaAPI, MenuAPI, etc.). Ce dernier point permet aux modules d'être eux aussi de très petite taille mais surtout de contenir le moins de bogues possible.

C'est ainsi que pour Drupal, chacune des fonctions fondamentales d'un CMS – comme la gestion du contenu, des commentaires ou encore des fichiers attachés – est prise en charge par un module autonome. La conséquence directe de cette approche est un système extrêmement robuste qu'il est difficile de déstabiliser.

L'autre implication de cette architecture est que chaque fonctionnalité de Drupal a été conçue dès l'origine pour pouvoir être modifiée ou remplacée par un module. Partie intégrante de tous les traitements, le champ d'action du module est donc beaucoup plus large que celui d'un simple greffon, ce qui explique en grande partie la multitude de modules Drupal disponibles aujourd'hui. Vous désirez authentifier vos utilisateurs

sur un service LDAP ? Il existe un module pour cela. Vous voulez accéder à votre contenu via le protocole webDAV ? Il existe également un module. Vous désirez importer un contenu à partir d'un fichier CSV, monter une boutique en ligne, intégrer un géolocalisation, ajouter des recherches par facette ? Des modules existent pour toutes ces tâches.

Au moment de la rédaction de ce livre, et rien que pour la version 6 de Drupal, le dépôt drupal.org comptait plus de 3 400 modules libres et donc gratuits, répartis sur une trentaine de domaines fonctionnels. Dans les rares cas où vous ne trouveriez pas votre bonheur, la création d'un nouveau module est d'une grande simplicité pour qui possède quelques connaissances basiques en PHP.

En résumé, cette architecture modulaire est l'atout majeur de Drupal, lui octroyant une grande robustesse alliée à une extrême souplesse, et lui permettant de coller à tous les usages. Drupal est donc effectivement une machine à assembler rapidement des sites web.

Cet atout n'est cependant pas sans danger. Nous y reviendrons plus en détail par la suite, mais sachez que tous ces modules ne sont pas – loin de là – de qualité et de maturité égales. Il est donc primordial de choisir avec grand soin les modules que vous intégrerez à votre site. Cela consiste généralement à exclure les modules qui ne sont pas encore déclarés comme stables, à étudier avec intérêt la liste des anomalies non résolues, et à consulter les statistiques fournies sur le site drupal.org qui indique pour chaque module le nombre de sites l'utilisant effectivement, ce qui constitue un bon indicateur de son niveau d'utilité.

Un contenu au centre de tout

Comme vous pouvez l'imaginer, Drupal évolue très vite avec une telle architecture, ce qui peut, à juste titre, inquiéter celui qui se préoccupe de la pérennité de son site, et particulièrement du contenu qu'il contient, sa véritable richesse.

En réalité, ce risque a été circonscrit depuis le début par un principe garantissant la conservation systématique des données d'une version à l'autre de Drupal. Les contenus saisis avec Drupal 4 continuent d'être « lisibles » avec Drupal 6 et le seront avec le futur Drupal 7. Votre investissement est ainsi protégé.

En contrepartie, aucune compatibilité ascendante n'est assurée pour les modules eux-mêmes. En d'autres termes, un module qui fonctionne avec Drupal 6 n'est pas assuré de fonctionner avec Drupal 7. C'est donc aux

développeurs des modules qu'incombe la responsabilité de la migration de leur projet d'une version à l'autre.

Cette approche permet à Drupal d'évoluer très rapidement dans le sens de la technologie tout en assurant la continuité de ce qui vous est le plus cher : vos données.

Maintenant rassuré sur la pérennité des données, vous pouvez légitimement vous demander ce qui se passera pour votre site si les développeurs de modules ne suivent pas le rythme des versions de Drupal. C'est en effet un risque à garder en tête, mais qui lui aussi a été prévu.

En effet, la communauté Drupal fournit des mises à jour régulières pour les deux dernières versions majeures du CMS : la version courante, et la version précédente. Au rythme d'une version annuelle, cette organisation laisse donc un an pour permettre aux développeurs de migrer leurs modules, ce qui dans la pratique est largement suffisant. Vous concernant, cela implique qu'à partir du moment où vous avez mis en œuvre Drupal, vous avez deux ans devant vous pour envisager sereinement la migration.

Les caractéristiques majeures de Drupal

La liberté des formats d'entrée

Comme nous l'avons vu, un CMS se caractérise par la séparation entre le fond et la forme. Drupal applique ce principe à la lettre en n'imposant à aucun moment le format des contenus que vous saisissez.

Pour y parvenir, l'outil se repose sur un concept de « chaîne de transformations » appelé « format d'entrée » qui va, par traitements successifs, produire un contenu au format XHTML exploitable sur le Web.

Vous avez ainsi la possibilité de définir autant de formats d'entrée que vous le désirez, ainsi que chacune de leurs étapes de transformation appelées « filtres ». Les filtres étant fournis par des modules, vous pouvez soit vous contenter de ceux présents en standard, soit en ajouter de nouveaux – vous permettant par exemple de prendre en charge des syntaxes plus exotiques (syntaxe Wiki ou phpBB), des outils de correction de syntaxe HTML, des filtres maquillant les adresses de courriel pour éviter les pourriels, etc.

Les chaînes de filtres ainsi constituées peuvent ensuite être appliquées selon les droits des utilisateurs et les types de contenus rédigés.

Un moteur de thème performant

Une fois le contenu formaté au standard XHTML, le moteur de thème de Drupal intervient. S'appuyant sur un jeu de modèles (templates PHP), ce module d'une grande souplesse rend possible toutes les fantaisies visuelles, comme faire varier l'apparence du site selon la section où l'on se trouve, le type de contenu affiché, le type de liste de résultats, ou encore un schéma spécifique d'URL.

Au-delà de sa puissance, le point intéressant de ce moteur est qu'il n'a pas été pensé pour le développeur PHP, mais pour l'infographiste, que l'on appelle le « thèmeur ». Il est donc très simple à mettre en œuvre sans pour autant connaître les structures internes de Drupal, l'essentiel du paramétrage se faisant par la nomenclature des modèles.

Le code des pages produit par Drupal est particulièrement propre et optimisé. Globalement, sans paramétrage et avec les thèmes standards, Drupal fabrique des pages parfaitement compatibles avec les standards XHTML/CSS, avec en prime une excellente accessibilité.

Enfin, il est important de noter que les URL de Drupal sont facilement indexables par les moteurs de recherche et sont totalement paramétrables. Ce point intéressera notamment tout public sensibilisé aux problématiques de référencement.

Une taxonomie native

Contrairement à d'autres CMS populaires qui limitent la catégorisation des contenus à seulement deux niveaux, Drupal intègre un moteur de taxonomie permettant des classements et des recherches sur un nombre illimité de critères, reprenant donc mot pour mot le concept de taxonomie que nous avons abordé au chapitre précédent.

Les taxonomies sont appelées « vocabulaires » et les termes (ou taxons) qu'ils contiennent peuvent être hiérarchisés les uns avec les autres. Un vocabulaire peut être rendu obligatoire pour une série définie de types de contenus.

Drupal permet également l'étiquetage libre (ou *free tagging*) qui consiste à associer une série de mots-clés, sans contrainte, à un contenu.

Cette capacité de catégorisation de contenu est ensuite utilisable pour les recherches, l'affichage par section, le croisement des critères, ou encore par des modules tiers, comme pour l'affichage de nuages d'étiquettes.

Internationalisation

Comme beaucoup de projets libres, Drupal est un produit développé depuis près de huit ans par des développeurs venant des quatre coins de la planète. C'est donc sans surprise que l'on disposera d'un produit pouvant s'afficher, selon les choix du visiteur ou les réglages de son navigateur, en plus de trente langues différentes, y compris asiatiques et arabes.

Mais l'aspect international de Drupal ne s'arrête pas à l'interface : il intègre en standard la capacité de gérer un même contenu en autant de langues que désiré, avec un système collaboratif de traduction des contenus.

Gestion des droits

Comme pour la taxonomie, la gestion des droits Drupal est poussée très loin pour obtenir un niveau de contrôle que l'on a davantage l'habitude de voir sur un système d'exploitation.

En effet, là où d'autres CMS se contentent d'une vision administrateurs/abonnés rendant délicat l'octroi de droits particuliers à certains abonnés, Drupal adopte la classique vue permissions/rôles/utilisateurs. Il est ainsi possible de manière très simple de donner à un utilisateur le droit de corriger des contenus sans lui permettre d'en créer, ou encore de modérer des commentaires sans qu'il lui soit possible d'aller plus loin.

L'inconvénient (tout relatif) de cette approche est qu'avec Drupal, la gestion des droits est une tâche à part entière qui doit être pensée au moment de la conception du site.

Ouverture aux bases de données

Beaucoup de CMS, surtout en PHP, se limitent à la simple prise en charge de la base de données MySQL. Sur cet aspect, Drupal adopte un positionnement agréablement agnostique (même si pour l'instant, seules deux bases de données sont effectivement prises en charge). Les modules se reposent en effet sur une batterie de fonctions qui leur cachent complètement la nature de la base de données sous-jacente, y compris le point délicat de la création des tables.

En résulte un CMS qui peut fonctionner nativement aussi bien sous MySQL que sous PostgreSQL. Nous ne parlons là que de la prise en charge par défaut, d'autres projets s'occupant par exemple de la base de données Oracle (<http://archimedes.bitfine.com/dms/oradруп>) ou encore Microsoft SQL Server.

Performances

Pour doper ses performances, Drupal utilise un système de cache très avancé. Pour ceux qui découvrent cette technique, un cache n'est rien d'autre qu'une zone dans laquelle Drupal va stocker une page, un bloc ou un contenu précalculé pour ne faire les traitements qu'une seule fois. Nativement, Drupal stocke son cache dans la base de données, mais comme toujours, cet aspect est modifiable par des modules pour permettre du cache en mémoire (extrêmement rapide) ou encore sur le système de fichiers.

Ceci dit, se reposer totalement sur ce système de cache pour garantir les performances d'un site à forte charge n'est pas forcément très sage. En effet, Drupal ne sait pour l'heure que mettre en cache les pages des visiteurs anonymes. Dès que l'utilisateur est identifié, une partie de ce système de cache est désactivée, ce qui implique une baisse des performances.

Ce problème ne touchera que peu de monde, car rares sont les sites dont la majorité de visiteurs est identifiée. Mais il faut malgré tout garder à l'esprit ce travers dans les performances de Drupal lors de la conception de son site, en évitant par exemple de le surcharger de modules inutiles.

Dimensionnabilité

S'agissant de sites à fort trafic, la capacité à monter en charge (*scalability*) devient un point crucial. Pour rappel, nous parlons ici de la possibilité d'augmenter facilement l'envergure du site (nombre d'utilisateurs, trafic, nombre de contenus, etc.) sans pour autant perdre en performances. Cette augmentation peut être ponctuelle (pic de charge) ou structurelle (succès du site, extension d'un département pilote à l'entreprise en général, etc.).

Dans ce dernier cas, la dimensionnabilité d'un outil se traduit généralement par sa capacité à fonctionner sur plusieurs machines, ce qui dans notre cas est plus le problème de la base de données que du CMS en lui-même. Malgré tout, Drupal apporte là aussi une flexibilité supplémentaire en acceptant l'utilisation de PostgreSQL qui se révèle beaucoup plus linéaire dans sa montée en charge que MySQL. Enfin, Drupal 7 améliorera la prise en charge des architectures redondantes, déjà présente partiellement dans Drupal 6.

Pour ce qui est d'absorber une charge ponctuelle, Drupal exploite son architecture modulaire par un mécanisme de balance de charge. Il sera ainsi possible de paramétrer certains modules et même certains contenus (les blocs) pour se désactiver automatiquement en cas de pics de trafic. Le site fonctionnera ainsi en mode « dégradé » mais sans rupture de service.

La licence GPL v2 de Drupal

Drupal est un logiciel libre, et c'est cette liberté qui rend possible le travail d'une communauté entière de développeurs, de traducteurs, d'auteurs d'articles et de documentation, produisant jour après jour un outil encore meilleur.

Ses sources, comme celles de l'ensemble des modules, sont disponibles sous Licence GPL v2. Vous avez donc le droit de copier le code sans payer de royalties, de mettre en place Drupal où vous le désirez, et de modifier le code qu'il contient comme bon vous semble.

En échange de cette liberté qui ne vous coûte rien, votre seule obligation consiste à rendre public le code que vous aurez modifié. Ainsi, si vous améliorez un module pour votre site, vous devez absolument proposer ces modifications à la communauté. Libre à elle, ensuite, de les accepter ou non.

Ce type de licence a fait naître une légende entretenue par nombre de personnes plus ou moins bien intentionnées qui voient d'un mauvais œil que l'on puisse mettre à disposition des ressources de qualité, alors qu'eux gagnent leur vie en faisant la même chose. Cette légende prétend qu'utiliser un logiciel libre comme Drupal vous obligerait à fournir à la communauté tout le code source de votre site. C'est absolument faux.

En effet, il y a une différence entre modifier une source existante et créer son propre code. Si pour votre site, vous fabriquez des thèmes et des modules, rien ne vous oblige à les placer sous licence GPL. Ce serait évidemment fortement apprécié, mais c'est votre droit de ne pas le faire.

Avec le libre, vous avez donc le choix au sens large du terme.

CULTURE Libre et gratuit, la différence

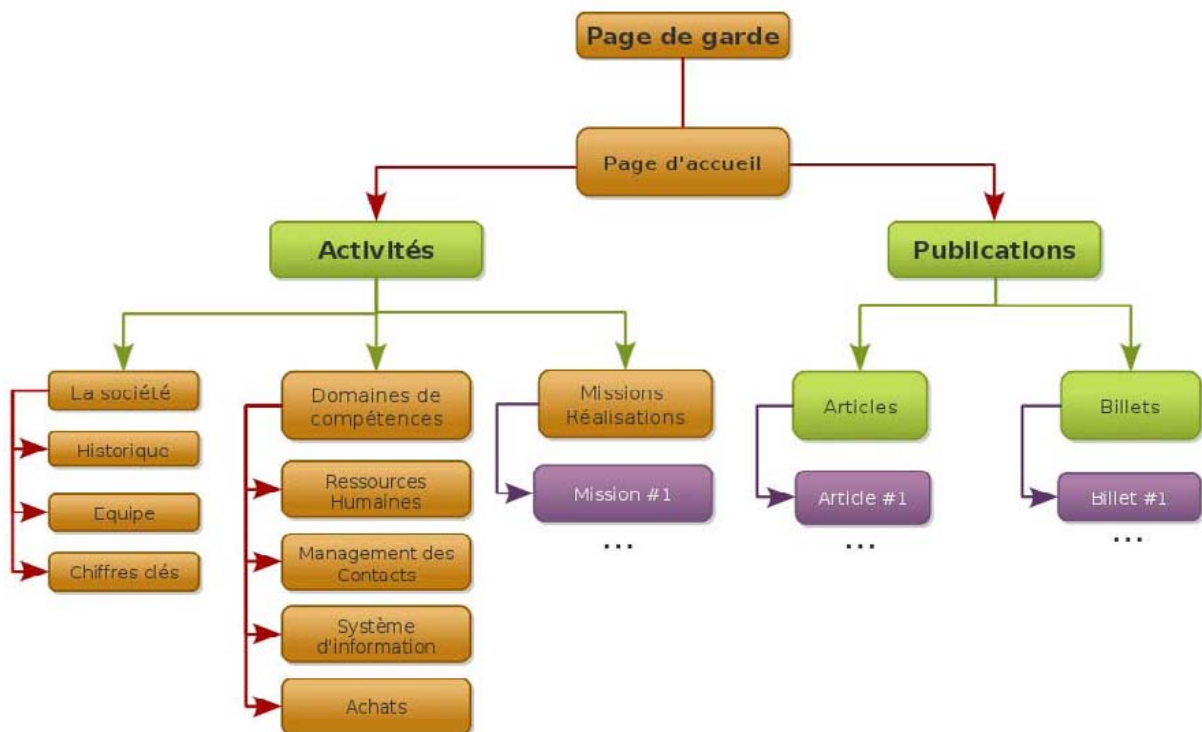
La confusion entre « libre » et « gratuit » qu'entretiennent souvent journaux et médias mérite d'être levée. Le lecteur de PDF fourni par Adobe est totalement gratuit, certainement pas libre. Vous ne pouvez en avoir le code source pour l'auditer par exemple, vous ne pouvez pas le traduire, ni en faire votre propre version et même son utilisation est soumise à conditions (conditions en général acceptées négligemment lors de l'installation). Un logiciel libre est l'inverse de tout cela. Vous disposez des sources, vous pouvez les lire, les modifier, les publier sous une autre forme. Et tout cela, plus que gratuitement : sans restriction aucune.

En résumé

- Drupal est un outil libre sous licence GPL v2, de même que l'ensemble des modules que nous utiliserons dans cet ouvrage.
- L'architecture de Drupal est modulaire et peut être étendue dans toutes les directions imaginables.
- Drupal implémente la quasi-totalité des concepts d'un CMS.
- De par sa nature complexe, il est important de garder en tête l'aspect performance qui peut devenir le point faible d'un site Drupal.

3

chapitre



Bien définir son projet

Quel que soit l'usage que vous réservez à Drupal, il faut se poser plusieurs questions avant de commencer le moindre travail. Car s'il est possible de faire tout ou presque avec Drupal, il est aussi possible de faire à peu près n'importe quoi. Mal définir le besoin, mais aussi le contexte technique du projet, c'est prendre le risque d'obtenir un site trop lent, non intégré avec l'environnement ou pire encore, purement et simplement rejeté par les internautes.

SOMMAIRE

- ▶ Définir l'objectif de son site
- ▶ Les bons réflexes pour formaliser ses idées
- ▶ Les étapes pour définir les fonctionnalités
- ▶ Penser aux contraintes techniques

Quel est votre objectif ?

Formaliser l'objectif de notre projet en quelques phrases simples permet d'avancer en ayant les idées claires, mais également de faire comprendre à d'autres ce que nous cherchons à réaliser. Aucun projet ne se réussit seul dans son coin, aussi n'hésitez jamais à en discuter avec vos collègues ou vos proches.

Dans ce livre, nous chercherons à réaliser le site web de la société (fictive) de conseil en management Colchique, dont l'objectif sera : « La présentation des domaines d'activité de la société et de ses références les plus prestigieuses. Il devra aussi permettre la publication d'articles techniques traitant de divers sujets liés à son cœur de métier. Le site proposera en outre une série de services transversaux comme l'abonnement à des lettres d'informations permettant à la société de communiquer sur son actualité et celle de son domaine. »

De l'objectif au storyboard

Avant de poursuivre, il convient de faire la distinction entre deux concepts, l'ergonomie d'un site, et son apparence.

L'ergonomie consiste à étudier la meilleure manière d'architecturer les pages de votre site pour correspondre parfaitement aux usages et aux besoins de vos visiteurs. Il s'agit principalement de déterminer ici la manière de naviguer, la disposition des contenus et des liens, les éléments mis en avant ou en arrière-plan, etc.

Ce processus se fait en gardant à l'esprit le public envisagé. Comment s'attend-il à trouver l'information ? Comment voudrait-il que le contenu soit organisé ? Quelles sont ses habitudes de travail, quelles choses souhaiterait-il trouver au premier regard ? Etc.

L'apparence de notre site correspond à l'image et à l'impression que vous cherchez à produire : choix de la palette de couleurs, du type de graphisme, etc. Lorsqu'il s'agit d'un site d'entreprise, nous pourrions nous appuyer sur sa charte graphique.

De manière générale, il convient déjà de séparer, ici aussi, le fond et la forme. L'ergonomie choisie pour un site a, certes, des incidences sur les couleurs ou les graphismes, mais il s'agit surtout de déterminer la manière dont il va être organisé, ce que contient chaque grande section, et la navigation qui va lier l'ensemble.

L'ergonomie en général est un bien vaste sujet, même réduit au seul domaine des interfaces humain-machine, et plus précisément aux applica-

tions web. C'est sans doute pour cela qu'il existe un métier d'ergonome à part entière. Si vous réalisez un site professionnel, il sera inévitable de faire appel à un tel spécialiste, ou plus généralement à un graphiste ayant de solides connaissances sur votre type de métier et sur l'ergonomie du Web.

Si vous êtes à l'aise avec ces notions ou que le site que vous comptez réaliser est suffisamment simple, vous pouvez faire ce travail vous-même. Dans tous les cas, le passage difficilement contournable reste la rédaction d'un storyboard.

RÉFÉRENCE Ergonomie web

A. Boucher, *Ergonomie Web – Pour des sites web efficaces*, 2^e édition, Eyrolles, 2009

Mettre en place le scénario (storyboard)

Un *storyboard* est un mot emprunté au vocabulaire cinématographique. Il s'agit de la représentation illustrée d'un film avant sa réalisation dans le but de planifier les plans principaux et leurs enchaînements. Pour un site web, le principe du storyboard est le même, à la différence que les plans seront plutôt les différentes sections, rubriques et autres pages spécifiques.

Il n'existe pas réellement de méthode stricte pour créer son storyboard. Il doit juste faire ressortir clairement les différentes sections du site et les enchaînements entre ses sections. Cela peut être fait de manière très simple par un organigramme fonctionnel.



Figure 3-1 Organigramme fonctionnel de notre site

Rien ne sert ici d'aller dans le détail des liens. Il s'agit de la navigation générale. Dans le schéma qui précède, il est important de bien montrer que :

- le site a deux facettes, une partie institutionnelle (*Activités*) et une partie rédactionnelle (*Publication*) ;
- le site présente d'abord une page de garde (qui peut être une animation, à utiliser avec parcimonie, car cela peut incommoder le visiteur), puis une page d'accueil qui permettra d'aiguiller les visiteurs vers l'une des sections.

Se plonger brièvement dans les détails

Il est possible et même souhaitable d'aller encore plus loin dans le détail en décrivant précisément chacune des présentations types. L'exemple ci-dessous correspondrait à la présentation de la page d'accueil.

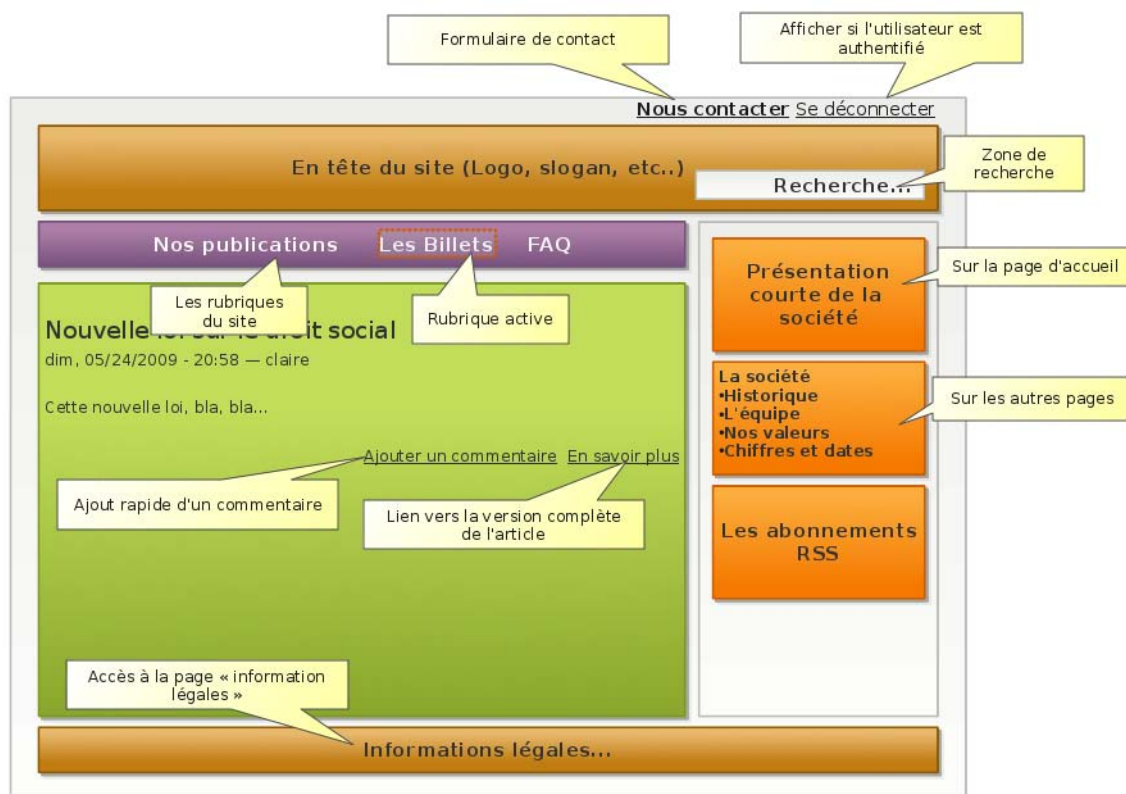


Figure 3-2 Le storyboard de la page d'accueil de notre site

Plus vous serez précis dans cette phase de conception, plus la suite se passera simplement pour vous et pour les personnes avec qui vous travaillerez.

Quels sont vos types de contenu ?

Une fois le storyboard terminé, nous avons une vision plus claire des types de contenu qui nous seront nécessaires, ainsi que des éléments les constituant :

- un contenu de type Référence : une fiche décrivant de manière synthétique une réalisation effectuée par la société, avec les domaines de compétences techniques mis en œuvre pour cette réalisation ;
- un contenu de type Article : une publication écrite par l'un des collaborateurs et traitant d'un sujet donné ;
- un contenu de type Billet : comme un article, mais plus court.

Pour nous assurer de l'exhaustivité de cette liste, nous pouvons la formaliser de manière plus graphique :

Articles	Billet	Référence
→ Titre	→ Titre	→ Objet de la mission
→ Auteur	→ Auteur	→ Date de la mission
→ Date de création	→ Date de création	→ Prestation réalisée
→ Domaines de compétence	→ Domaines de compétence	→ Résultats obtenus
→ Corps	→ Corps	→ Domaine de compétence
		→ Acteurs

Figure 3–3
Les types de contenu à implémenter

Les rôles de vos utilisateurs

Le CMS est un outil collaboratif qui permet à tous les utilisateurs autorisés de créer, modifier, détruire et publier (c'est-à-dire rendre public) des contenus.

À l'exclusion du cas d'un site personnel, la définition des rôles, et donc des permissions de chacun dans ce processus d'animation, est une étape aussi importante que généralement oubliée. Il serait pourtant dommage que des informations confidentielles soient rendues publiques par accident...

Comme nous l'avons déjà vu, un rôle dans un CMS correspond à une liste de permissions pouvant être attribuées à un utilisateur. Du point de vue organisationnel, il s'agit de la fonction que prend un collaborateur à un moment t donné pour effectuer une tâche spécifique.

Pour le site de la société Colchique, nous déciderons que tous les contenus de type Article (les publications) passeront par quatre états avant leur publication : Brouillon, À corriger, Corrigé et Validé. Cet ensemble d'états forme les étapes de ce qui est appelé un flux de travaux (en anglais

workflow). Chaque passage d'étape est assuré par des utilisateurs ayant le rôle approprié :

- l'auteur (aussi appelé contributeur), qui est la personne rédigeant des contenus. C'est lui qui fera passer le contenu de l'état Brouillon à À corriger ;
- le correcteur, qui est la personne relisant et corrigeant les contenus dans l'état À corriger, pour finalement les faire évoluer vers l'état Corrigé ;
- le valideur, qui valide les contenus marqués Corrigé et décide de leur passage à l'état Validé, condition nécessaire à leur publication ;
- le visiteur anonyme, qui consulte les contenus publiés sur le site.

Ainsi, un document publié sur le site et visible par tout public sur l'Internet aura forcément suivi nos étapes de traitement, garantissant qu'il ne contient ni coquille, ni information sensible.

Vous pouvez encore une fois décrire ces flux de travaux de manière plus visuelle :

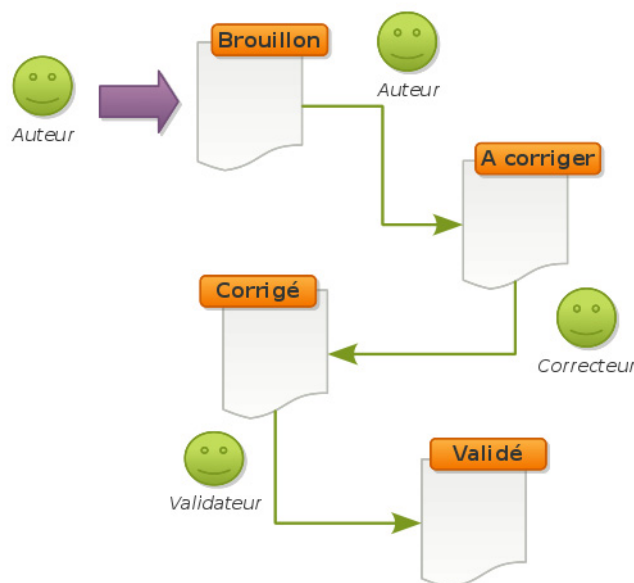


Figure 3-4
Le flux de travaux

Prévoir vos contraintes techniques

À ce stade, nous avons une vision claire des informations que nous manipulerons, et avons défini les rôles et les types de contenu. Il nous reste à penser aux contraintes que nous allons imposer à notre nouveau projet.

Le système d'information

Si votre objectif est de créer un site personnel, ceci vous touchera peu, mais si, comme dans notre exemple, vous œuvrez pour une société, il est important de prendre conscience que votre site web n'est pas une île isolée. Il s'intègre dans un ensemble plus vaste que l'on appelle communément « le système d'information » de la société. Il devra sans doute puiser des informations dans des bases de données externes à Drupal (par exemple les noms et mots de passe des collaborateurs). Il pourra aussi avoir à s'interfacer avec d'autres systèmes pour en retirer des données ou en injecter (par exemple un système de paiement en ligne, un système de comptabilité, etc.).

Tous ces éléments doivent impérativement être listés dès à présent pour éviter toute mauvaise surprise au moment de la mise en œuvre.

Le contexte technique

Dans le même style d'idées, vous devez penser à l'endroit où le site sera hébergé. Est-ce un serveur dédié ou une machine partagée avec d'autres applications ? Dans le second cas, la base de données vous sera sans doute imposée, ou encore le système d'exploitation. Il est important de le savoir dès le début, pour ne pas démarrer le travail en utilisant des éléments incompatibles avec l'infrastructure finale. Un exemple classique est de construire puis tester votre site sur une base de données (par exemple MySQL), pour découvrir que la seule machine que l'on vous propose pour l'hébergement utilise PostgreSQL. Ce n'est pas dramatique, mais cela vous obligerait à retester tout votre site pour vous assurer de sa bonne compatibilité.

Votre public et la fréquentation escomptée

Le dernier aspect à évaluer est la fréquentation que nous attendons sur ce site. En effet, un site qui reçoit une centaine d'utilisateurs par jour n'aura pas les mêmes contraintes qu'un autre qui en recevra autant... par seconde. La manière de construire le site ne sera pas exactement la même dans l'un ou l'autre cas. Vous devez donc estimer aussi précisément que possible la volumétrie. Dans ce calcul, pensez à séparer la volumétrie des utilisateurs anonymes de celle des utilisateurs authentifiés. Drupal ne les gère, en effet, pas de la même manière, et si votre site est majoritairement utilisé par des utilisateurs connus du système, certaines options seront plus pertinentes que d'autres.

BON À SAVOIR Les résolutions

Les résolutions d'écran à prendre en compte lors de la conception d'un site web sont à nuancer en fonction du panel envisagé de vos utilisateurs. En effet, un public fortement technophile sera sans nul doute mieux équipé qu'un public plus large. Notez qu'en juin 2009 (source LibStat), les résolutions les plus utilisées sont :

- 1024 × 768 (33 %) ;
- 1280 × 800 (16 %) ;
- 1280 × 1024 (12,58 %) ;
- 1400 × 900 (11,5 %).

BON À SAVOIR Les navigateurs et les moteurs de rendu

Il n'est pas forcément évident de s'y retrouver dans la jungle des navigateurs, sauf si l'on raisonne à partir de leur moteur de rendu. En effet, chaque navigateur utilise une bibliothèque particulière, qui a pour rôle de transformer le code HTML/CSS en une représentation visuelle. Cette bibliothèque s'appelle un *moteur de rendu*.

Il existe beaucoup moins de moteurs de rendu que de navigateurs, ou en d'autres termes, plusieurs navigateurs utilisent le même moteur de rendu :

- **Gecko** est le moteur de rendu d'un très grand nombre de navigateurs : Netscape, Firefox, Iceweasel, Camino (Mac OS), Flock, Epiphany et Galeo (Gnome), etc. ;
- **Trident** est le moteur de Microsoft, utilisé dans les navigateurs Internet Explorer (à partir de la version 4), MyIE (aka Maxthon), Avant Browser ou encore AOL Explorer ;
- **WebKit** (issu du moteur KHTML) est utilisé dans les navigateurs Epiphany et Midori (Gnome), Google Chrome, Konqueror, OmniWeb, Dafari, Shiira, etc. ;
- **Presto** est le moteur d'Opera.

Sachant cela, vous pouvez spécifier comme contrainte technique le moteur de rendu et ainsi vous assurer (dans une certaine mesure) de la compatibilité avec les navigateurs qui l'utilisent.

En juin 2009 (source Libstat), les navigateurs les plus utilisés sont :

- Microsoft Internet Explorer 7 (37,96 %) ;
- Firefox 3 (24,4 %) ;
- Microsoft Internet Explorer 6 (16,86 %).

En résumé

- L'étape de définition d'un projet n'est pas seulement utile, elle est indispensable pour réaliser un travail propre et dans les délais fixés.
- Le site que vous concevez n'est pas là pour répondre à vos besoins, mais à ceux de vos visiteurs et utilisateurs.
- La conception d'un site passe par la définition des acteurs (rôles), des fonctionnalités attendues et de leur organisation (navigation).
- Faites le tri entre les fonctions fondamentales (ex. créer un article) et les fonctions transversales et utilitaires (ex. générer un PDF).
- N'omettez jamais de définir précisément la plate-forme technique cible et la fréquentation attendue.

4

chapitre



The screenshot shows the Drupal 7 administrator interface. At the top, there is a blue header with the Drupal logo and the site name 'Colchique'. Below the header, on the left, is a sidebar menu for the 'administrateur' (administrator) role, containing links for 'Mon compte', 'Créer un contenu', 'Administrer', and 'Se déconnecter'. The main content area displays a 'Bienvenue sur votre nouveau site Drupal !' (Welcome to your new Drupal site!) message. It includes a thank-you note and a four-step guide for configuring and using the site.

Bienvenue sur votre nouveau site Drupal !

Merci de suivre ces étapes pour configurer et commencer à l'utiliser :

- 1. Configurez votre site web**
Une fois connecté, visitez la [section d'administration](#), où vous pouvez [perso](#)
- 2. Activez des fonctions supplémentaires**
Ensuite, visitez la [liste des modules](#) et activez les fonctionnalités qui convie modules dans la [section de téléchargement de modules de Drupal](#).
- 3. Personnalisez l'apparence de votre site web**
Pour changer l'apparence et le comportement de votre site web, visitez la [se](#) ou télécharger d'autres thèmes dans la [section de téléchargement des thème](#)
- 4. Commencez à contribuer**
Enfin, vous pouvez [créer du contenu](#) pour votre site web. Ce message disp: d'accueil.

Mettre en place son environnement de développement

Comme n'importe quel projet informatique, un site web nécessite la mise en place d'un environnement de développement sur une machine de travail. Plus cette plate-forme sera propre et efficace, moins vous serez tenté de travailler directement sur le serveur de production.

En prenant l'habitude de travailler et de tester sur cet environnement dédié, vous contribuerez ainsi à augmenter la robustesse de votre site.

SOMMAIRE

- ▶ Apache et PHP, les moteurs de Drupal
- ▶ Mettre en place un environnement WAMP et LAMP
- ▶ Créer votre base de données
- ▶ Résoudre les difficultés techniques

Les technologies mises en œuvre

Le service HTTP

Un site web fonctionne peu ou prou toujours de la même manière. Le navigateur web (Firefox, Safari, Opera, Internet Explorer, etc.) se connecte à un service HTTP en lui fournissant une requête portant sur une URL donnée (ex. `http://mon_site.com/ma_page.html`). Le service HTTP va alors chercher s'il existe un fichier nommé `ma_page.html`, et si c'est le cas, le renvoyer au navigateur qui va simplement en afficher le contenu.

Si le fichier HTML contenait le code suivant :

```
<HTML>
  <BODY>
    Bonjour !
  </BODY>
</HTML>
```

Le navigateur afficherait alors le texte « Bonjour ! » en noir sur fond blanc.

JARGON Serveur et service

Le mot *serveur* est un élément du jargon désignant la machine qui va répondre à une *requête* par le biais d'un logiciel, le *service*. Ce vocabulaire est tiré de la vie réelle où nous avons le garçon de café (le serveur) qui va prendre notre commande (la requête) et nous apporter un expresso (le service).

Le problème est que le mot « serveur » est très souvent utilisé dans deux contextes qui n'ont pourtant rien à voir. Parfois il désigne un service, c'est-à-dire un logiciel (ex. service HTTP, serveur SMTP) qui est installé parmi d'autres services sur une même machine. Parfois il désigne la machine elle-même (ex. le serveur de mise en production). La raison de cette confusion est que dans beaucoup de cas, un serveur (la machine, cette fois) est utilisé pour un et un seul service (ex. serveur de messagerie). Mais dans la réalité, il est tout aussi réducteur d'assimiler un serveur à un service, que de réduire le garçon de café à la seule capacité de servir des expressos.

Dans la mesure du possible, pour éviter toute confusion, nous essaierons dans ce livre d'utiliser plutôt le mot « service » lorsqu'il s'agira du logiciel et « serveur » pour la machine.

Si la page contient un lien hypertexte et que l'utilisateur clique dessus, le navigateur fera à nouveau une requête au service HTTP portant sur l'URL du lien, et récupérera une nouvelle page. Et ainsi de suite.

Le service HTTP est donc le logiciel qui va répondre à une requête du navigateur web. Il existe de nombreux services HTTP différents : LightHTTP, Apache ou encore IIS. Drupal ayant été conçu pour être le plus souple possible, il est techniquement possible de le faire tourner sur la majorité d'entre eux. Cependant, Apache reste le service HTTP de loin le plus populaire. C'est donc lui que nous utiliserons pour construire notre plate-forme de développement.

Le langage PHP

PHP (pour *Hypertext PreProcessor*) est un langage de programmation spécialement conçu pour réaliser des applications web. La différence entre application web et site web vient cette fois du fait que le contenu des pages n'est plus statique, mais construit dynamiquement à chaque requête. L'objectif de PHP est ici de modifier le contenu HTML en lui ajoutant des instructions spéciales. Si nous voulions ajouter l'affichage de l'heure à notre exemple précédent, nous ajouterions le code PHP suivant :

```
<HTML>
  <BODY>
    Bonjour, il est <?php print date() ?> !
  </BODY>
</HTML>
```

Comme vous le voyez, l'inclusion d'instructions PHP dans du code HTML est aussi simple que directe. C'est d'ailleurs cette caractéristique qui en a fait le langage le plus utilisé au monde.

Drupal étant écrit en PHP, ce dernier lui est absolument nécessaire. La version 4 de PHP suffit pour travailler avec Drupal jusqu'à sa version 6 ; en revanche, PHP 5 sera nécessaire à Drupal 7. Il est donc conseillé d'utiliser dès maintenant PHP 5.

La base de données

Drupal utilise donc PHP pour construire ses pages « à la volée », mais il ne serait rien s'il n'avait pas une base de données pour le seconder. En effet, toute la configuration du site, le thème utilisé, la position des éléments dans la page, et bien évidemment, les contenus eux-mêmes sont autant d'informations stockées dans cette base.

Nous l'avons vu, un des points forts de Drupal est sa souplesse vis-à-vis des bases de données. Techniquement, Drupal 6 pourrait tourner avec n'importe laquelle d'entre elles, sous réserve que le greffon correspondant existe. Dans les faits, les deux bases de données les plus utilisées avec ce CMS sont MySQL et PostgreSQL. Le choix de l'une ou l'autre

HISTOIRE **Personal Home Page ou Hypertext Preprocessor ?**

Lorsqu'en 1995, Rasmus Lerdorf libéra le code source de PHP, l'acronyme signifiait *Personal Home Page*. En 1997, Andi Gutmans et Zeev Suraski redéveloppèrent complètement PHP pour donner naissance un an plus tard à Zend Engine, la version 3 du cœur de PHP, qui signifiait cette fois Hypertext Preprocessor.

de ces deux bases vous appartient donc. Concernant MySQL, sachez seulement que Drupal ne fonctionnera pas en dessous de la version 4.1.

B.A.-BA Les bases de données

Une base de données est un logiciel dont le but est de stocker et de manipuler des données. On parle aussi très souvent de SGBD pour Système de gestion de bases de données.

Dans une base de données dite *relationnelle* (ou SGBD-R), les informations sont stockées dans des tables, composées de colonnes appelées *champs*, et de lignes contenant les valeurs de ces champs. Chaque ligne est appelée enregistrement ou *tuple*. L'aspect relationnel tient à ce que les tables sont généralement reliées entre elles. Par exemple, une table « clients » a une relation avec une table « sociétés », qui est « la société à laquelle appartient le client ».

À une certaine époque, les SGBD-R étaient totalement autonomes et contenaient des applications complètes. Aujourd'hui, dans la plupart des cas, elles constituent avant tout l'espace de stockage utilisé par des applications plus évoluées comme Drupal.

Pour discuter avec un SGBD-R, humains et applications utilisent le même langage, le SQL. Ce langage est utilisé sous la forme de requêtes à la base de données pour créer de nouvelles tables ([CREATE TABLE](#)), insérer des enregistrements ([INSERT](#)), les mettre à jour ([UPDATE](#)) et faire des recherches ([SELECT](#)).

Il existe de très nombreux SGBD-R (Oracle, Sybase, Microsoft SQL Server, etc.) dont certains sont des logiciels libres (MySQL, PostgreSQL, BerkeleyDB, etc.).

CULTURE Qu'est ce que LAMP ?

LAMP est un acronyme trouvé par Michael Kunze dans le magazine Allemand *Computertechnik* (C't). L'objectif de l'article était de prouver qu'un assemblage de logiciels libres (en l'occurrence Linux, Apache, MySQL et PHP) pouvait sans problème concurrencer les offres commerciales présentes alors sur le marché.

Le principe de l'environnement LAMP a plus tard été décliné en WAMP (Windows Apache MySQL PHP), puis MAMP (MacOS Apache MySQL PHP), puis avec la démocratisation de l'excellente base de données PostgreSQL, sont arrivés les architectures LAPP (Linux Apache PostgreSQL PHP).

Le système d'exploitation

Dans ce domaine, vous êtes absolument libre. Les trois briques que nous venons de décrire fonctionnent aussi bien sous GNU/Linux que sous Solaris, Mac OS ou encore Windows. Il est donc possible de combiner tout cela sans problème et de monter une plate-forme WAMP (Windows Apache MySQL PHP), MAMP (MacOS Apache MySQL PHP), LAPP (Linux Apache PostgreSQL PHP), etc.

Pour la majorité de systèmes d'exploitation, il existe de nombreux paquets prêts à l'emploi qui installent l'ensemble (Apache, MySQL, PHP, etc.) en une seule passe.

Puisqu'il faut faire des choix, nous allons décrire ici l'installation de deux types de plates-formes : une WAMP et une LAMP.

Création de la plate-forme de développement

Création d'un environnement WAMP

Installation de WampServer

Pour installer Apache, PHP et MySQL sous Windows, nous allons choisir, quelque peu arbitrairement, le paquet WampServer (<http://www.wampserver.com>), mais il y en a beaucoup d'autres qui fonctionnent tous à peu près sur le même principe.

Pour commencer, vous devez télécharger le paquet à installer à l'adresse <http://www.wampserver.com/download.php>. Nous utilisons ici la version 2.0. En cliquant sur le lien de téléchargement, vous êtes détourné sur le site sourceforge.net qui héberge les binaires du projet.

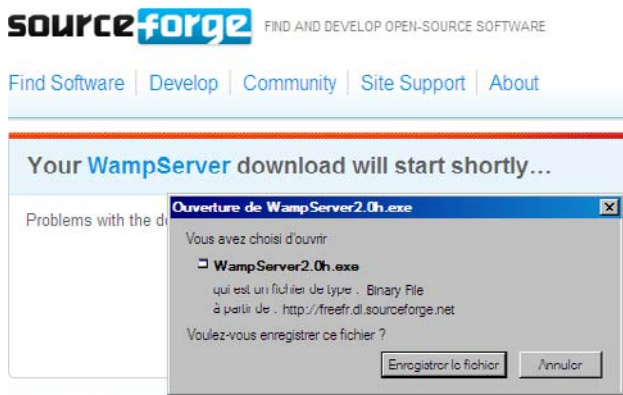


Figure 4-1
Téléchargement de l'archive de WampServer 2.0
à partir des serveurs de sourceforge

Une fois le fichier `WampServer2.0h.exe` récupéré, vous pouvez double-cliquer dessus pour l'installer.

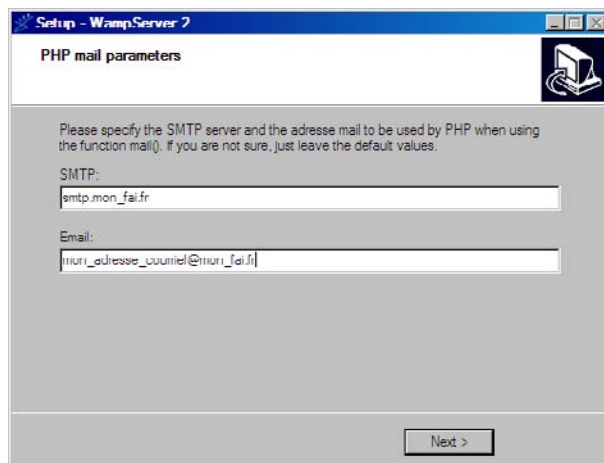


Figure 4-2
Choix du service SMTP

JARGON Service SMTP

SMTP veut dire *Simple Mail Transfer Protocol* (littéralement, protocole simple de transfert de courrier). Il s'agit du mode de communication standard utilisé par deux machines pour s'envoyer du courrier.

Lorsque vous cliquez sur le bouton *Envoyer* de votre client de messagerie, ce dernier utilise la plupart du temps ce protocole pour transférer votre message au service SMTP de votre fournisseur d'accès à Internet (FAI). Une fois que ce dernier a reçu un message, il va à son tour contacter le service SMTP de votre correspondant (le champ *à* de votre message) pour lui remettre à son tour le courrier.

Dans notre cas, le service SMTP a une grande importance, car c'est lui qui recevra tous les courriels de confirmation, notifications et autres messages d'information qui vont être envoyés par Drupal tout au long de notre exploration.

Au cours du processus d'installation vous seront demandés les éléments suivants :

- le dossier d'installation : laissez les valeurs par défaut `c:\wamp` ;
- si vous voulez utiliser la page de garde de WampServer, répondez *oui* ;
- *PHP Mail parameters* : dans le champ *SMTP*, fournissez l'adresse du service SMTP qui vous a été communiquée par votre fournisseur d'accès à l'Internet (FAI, voir figure page précédente). Dans le champ *E-mail*, vous pouvez indiquer votre adresse de courriel qui sera utilisée par défaut dans le champ *de* des messages envoyés.

Une fois installé, WampServer propose de lancer l'ensemble de ses services (MySQL et Apache) directement, ce que nous acceptons.

Test de la plate-forme

Si tout s'est bien passé, nous devrions voir apparaître une nouvelle icône dans la boîte à miniatures (sur la droite de la barre de tâches, à côté de l'heure).

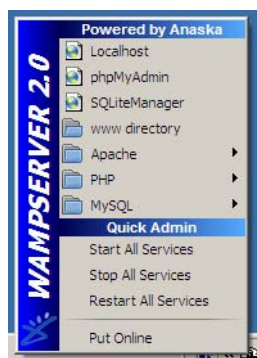


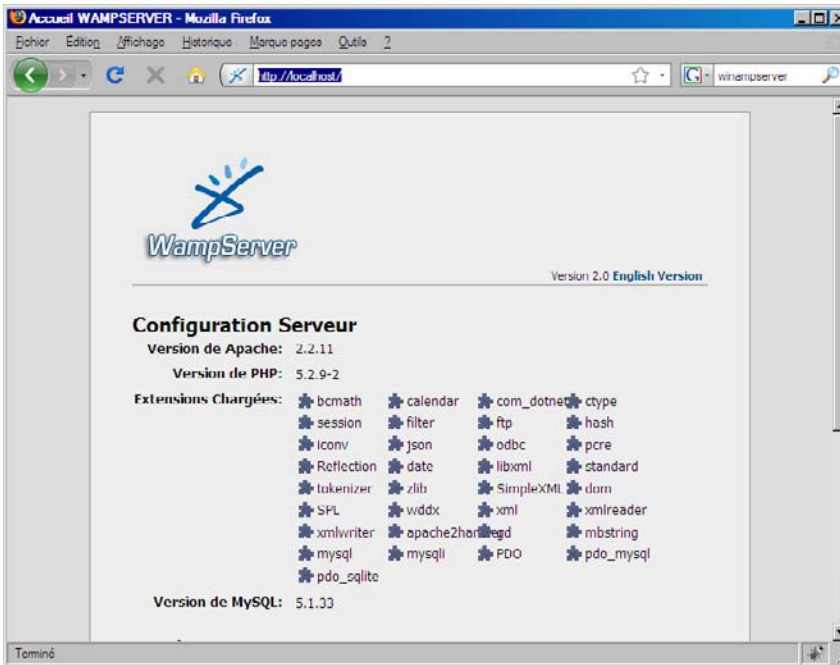
Figure 4-3
Le menu de WampServer

Vous disposez en bas de ce menu des options d'arrêt des services Apache et MySQL (*Stop All Services*), de lancement (*Start All Services*) et de redémarrage (*Restart All Services*).

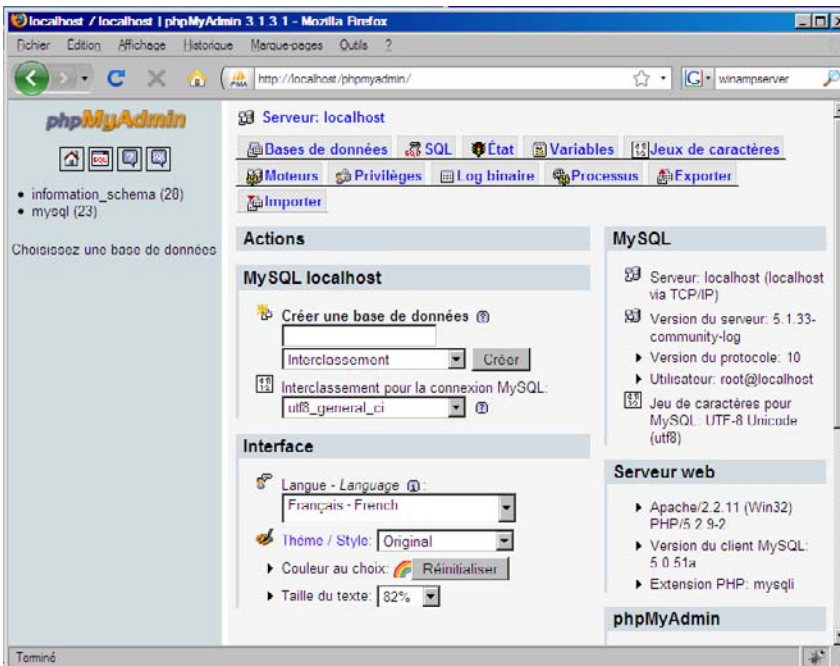
Nous commencerons donc par démarrer ces services et vérifier leur bon fonctionnement en tapant `http://localhost` dans la barre d'adresse de notre navigateur. La page de garde de WampServer devrait alors apparaître.

Dans ce paquet, vous avez aussi accès à phpMyAdmin, un outil simplifié d'administration de la base de données MySQL, à l'adresse `http://localhost/phpmyadmin`.

Comme la page de phpMyAdmin s'affiche correctement, nous pouvons en déduire que la base de données MySQL est opérationnelle. Notez au passage que par défaut, le service MySQL de PhpMyAdmin est configuré sans mot de passe. Il n'est donc pas nécessaire de s'authentifier pour s'y connecter, comme ce sera le cas pour notre installation LAMP.

**Figure 4–4**

La page de démarrage de WampServeur : tout fonctionne...

**Figure 4–5**

L'outil indispensable à MySQL, phpMyAdmin



Figure 4-6
La page par défaut de WampServer

Repérer les fichiers source

Plus prosaïquement, la page de garde disponible à l'adresse <http://localhost> est un fichier `index.php` stocké en local dans le dossier `c:\wamp\www`. Vous pouvez y accéder directement par le menu WampServer *www directory*.

Tous les fichiers qui seront posés à cet endroit seront donc accessibles par le biais du navigateur web. Nous pouvons en faire l'expérience en ajoutant ici un fichier `test.php`. Ce fichier peut être créé avec Notepad et contenir par exemple le code PHP suivant :

```
<html>
  <head>
    <title>Bonjour le monde!</title>
  </head>
  <body>
    <?php
      print "Bonjour, il est ".date("H:m:s");
    ?>
  </body>
</html>
```

Une fois notre page de test sauvegardée, nous pouvons la tester en saisissant l'adresse <http://localhost/test.php> dans le navigateur. Nous obtenons alors la page de statut du moteur PHP.

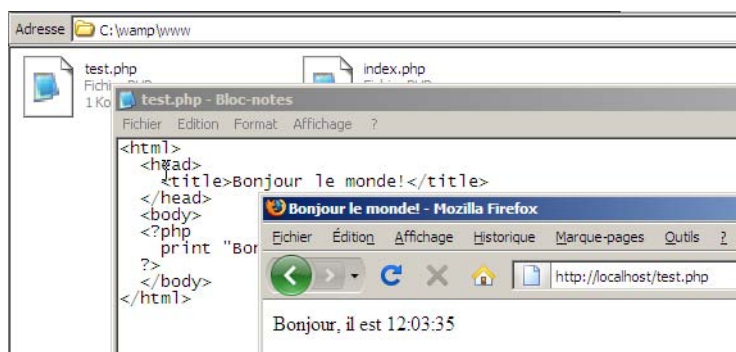


Figure 4-7
WampServer fonctionne donc bien.

Création d'un environnement LAMP

Considérations sur les distributions

Dans la suite de ce paragraphe, nous partons du principe que vous êtes déjà familiarisé avec le système d'exploitation Linux et plus particulièrement avec la notion de paquets d'installation.

Pour ce qui est de la distribution, nous utilisons dans cet exemple Mandriva, mais la transposition à une autre distribution est assez directe,

les noms des paquets étant toujours plus ou moins les mêmes. Seules les commandes utilisées pour les installer diffèrent.

Ici nous allons utiliser uniquement la ligne de commande au risque d'entretenir le mythe d'un Linux peu ergonomique. En réalité, toutes les distributions permettent aujourd'hui de faire ce travail aidé d'une interface graphique. Malgré tout, l'approche en ligne de commande reste de loin la plus facile à transposer d'une distribution à l'autre.

Installation d'Apache

La première étape de l'installation se fait à partir d'une console ouverte sur l'utilisateur `root`. Ceci est nécessaire pour avoir le droit d'installer des paquets et de modifier les configurations.

Nous commençons par le service Apache, dont il existe plusieurs variantes qui diffèrent surtout par la manière de gérer les accès concurrents. Pour PHP, il est conseillé d'utiliser la variante `mpm-prefork`.

```
root# urpmi apache-mpm-prefork
```

Répondez par oui à toutes les demandes d'installation qui vont suivre après le lancement de cette commande, il s'agit des dépendances nécessaires à la bonne marche d'Apache.

Une fois le processus terminé, nous pouvons démarrer le service HTTPD (Apache) de la manière suivante :

```
root# /etc/init.d/httpd start
Starting http: [ OK ]
```

Nous pouvons valider cette étape à l'aide d'un navigateur web en allant à l'adresse `http://localhost` :



BON À SAVOIR Task-lamp

Sous Mandriva, nous disposons aussi d'un méta-paquet nommé `task-lamp`, qui comme son nom l'indique, provoque l'installation directe de tout l'environnement LAMP.

Figure 4–8
Test du bon fonctionnement d'Apache

Repérer les fichiers source

Par défaut, Apache va chercher ses données dans le dossier `/var/www/html`. S'agissant d'une installation de développement, nous allons nous approprier ce dossier pour pouvoir le manipuler dans l'avenir sans avoir à utiliser le compte root. Si nous partons du principe que l'utilisateur sera Gaston, nous lui donnerons les droits sur le dossier de la manière suivante :

```
root# chown gaston:gaston /var/www -R
```

À présent, l'utilisateur Gaston est le propriétaire officiel de ce dossier et Apache garde la possibilité de le lire.

Comme pour notre installation WAMP, nous pouvons rédiger un petit test pour vérifier que PHP fonctionne correctement. Cette fois, il n'est plus nécessaire d'être root, Gaston peut ouvrir sa propre console pour effectuer cette tâche :

```
gaston$ cat > /var/www/html/test.php
<html>
  <head>
    <title>Bonjour le monde!</title>
  </head>
  <body>
    <?php
      print "Bonjour, il est ".date("H:m:s");
    ?>
  </body>
</html>
CTRL + D
gaston$
```

Maintenant que le fichier est créé, nous allons pouvoir l'afficher en saisissant l'adresse `http://localhost/test.php` dans le navigateur :

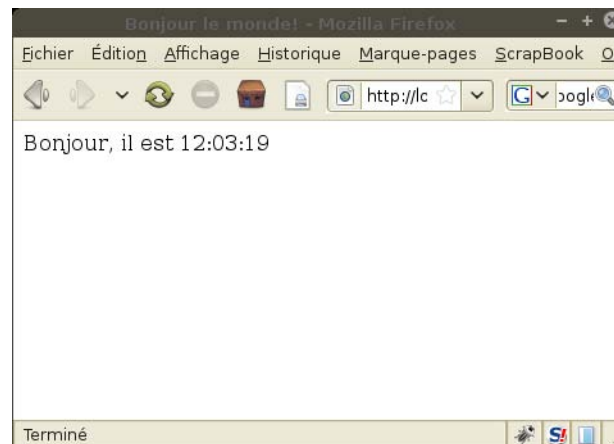


Figure 4–9
Test du bon fonctionnement de PHP

Installation de MySQL

L'installation de MySQL se passe de la même manière que pour Apache, en tant qu'utilisateur root :

```
root# urpmi mysql
```

Lançons le service pour nous assurer que tout fonctionne :

```
root# /etc/init.d/mysqld start
Starting MySQL: [ OK ]
```

Une fois MySQL démarré, nous pouvons utiliser une procédure pour configurer rapidement les mots de passe et la sécurité :

```
root# mysql_secure_installation
NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MySQL
SERVERS IN PRODUCTION USE! PLEASE READ EACH STEP CAREFULLY!
```

In order to log into MySQL to secure it, we'll need the current password for the root user. If you've just installed MySQL, and you haven't set the root password yet, the password will be blank, so you should just press enter here.

Enter current password for root (enter for none):
OK, successfully used password, moving on...

Setting the root password ensures that nobody can log into the MySQL root user without the proper authorisation.

Set root password? [Y/n]

À la première question, entrez *y* (oui) et saisissez votre mot de passe. Conservez soigneusement ce mot de passe, sinon vous ne pourrez plus vous connecter à votre base de données.

Aux questions suivantes, répondez comme suit :

```
remove anonymous users ? y
disallow root login remotely ? y
remove test database and access to it ? y
reload privilege tables now ? y
```

Pour valider la bonne marche de MySQL, nous utilisons le client en ligne de commande en prenant l'utilisateur MySQL root et le mot de passe que nous venons de donner :

```
root# mysql -u root -p
mot de passe: secret
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 14
```

```
Server version: 5.0.67 Mandriva Linux - MySQL Standard Edition
(GPL)

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> show databases ;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| test |
+-----+
3 rows in set (0.00 sec)

mysql>quit
root#
```

Installation de phpMyAdmin

Le client en ligne de commande est très utile, mais il est appréciable de disposer comme sous Windows de phpMyAdmin. Pour cela, il nous suffit d'installer le paquet correspondant :

```
root# urpmi phpmyadmin
```

Le service MySQL a redémarré automatiquement et il ne reste plus qu'à tester cela dans le navigateur à l'adresse <http://localhost/phpmyadmin>.

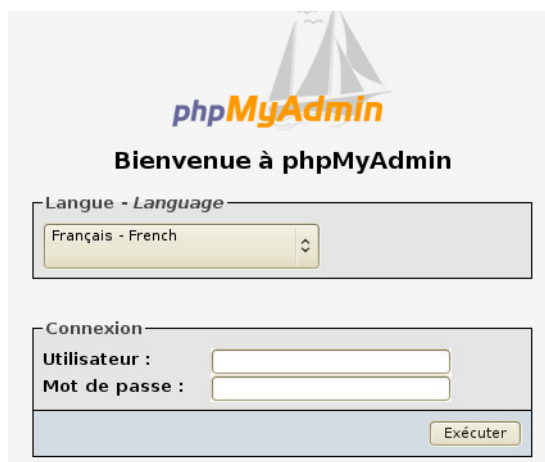


Figure 4-10

La page d'authentification de phpMyAdmin

Nous disposons à présent de l'environnement complet pour installer Drupal.

Téléchargement de Drupal

Nous avons maintenant, sous Windows ou sous Linux, une plate-forme de développement opérationnelle à laquelle ne manque que Drupal. À partir de maintenant, que vous soyez sous GNU/Linux, Mac OS ou Windows, nous n'aurons plus besoin que d'un simple navigateur.

Site de téléchargement

Pour télécharger la dernière version de Drupal, il suffit d'aller sur la page d'accueil du projet Drupal à l'adresse <http://drupal.org>. Toutes les ressources téléchargeables sont disponibles en cliquant sur l'onglet *Download*.

Sur cette page se trouvent tous les types de ressources disponibles :

- *Drupal project* : les sources de Drupal ;
- *Installation profiles* : les profils d'installation spécialisés pour certaines tâches particulières ;
- *Modules* : tous les modules que l'on peut ajouter à Drupal ;
- *Themes* : les thèmes libres que nous pourrions utiliser pour changer l'apparence de Drupal ;
- *Theme engines* : ce sont des moteurs de thème qui pourront éventuellement être utilisés dans de rares cas où le moteur standard ne conviendrait pas ;
- *Translations* : les traductions de Drupal en diverses langues.

Pour l'instant, le type de ressource qui nous intéresse est *Drupal project*.

Project types

- **Drupal project**

Get started by downloading the official Drupal core files. These official releases come bundled with a variety of modules and themes to give you a good starting point to help build your site. Drupal core includes basic community features like blogging, forums, and contact forms, and can be easily extended by downloading other contributed modules and themes.

Une fois que vous avez cliqué sur ce lien, la liste des versions courantes de Drupal apparaît. En vert, nous avons les versions stables et en rouge les versions en cours de développement.

Nous allons ici choisir la ligne verte (stable) de la version la plus récente de Drupal (6.10). Une fois la ligne repérée, cliquez sur le lien *Download* correspondant.

Figure 4–11

La section de téléchargement du projet Drupal

ATTENTION

Les versions instables des modules

Sauf si vous désirez tester des versions expérimentales de Drupal, il est fortement conseillé de ne prendre que des versions stables.

Figure 4–12
Les deux versions de Drupal

Releases				
Official releases	Date	Size	Links	Status
6.10	2009-Feb-26	1.03 MB	Download · Release notes	Recommended for 6.x ✓
5.16	2009-Feb-26	748.29 KB	Download · Release notes	Recommended for 5.x ✓

Le navigateur nous propose alors de sauvegarder une archive ayant pour extension `.tar.gz`, par exemple sur votre bureau.

BON À SAVOIR Les outils de décompression

Pour les *windowsiens*, les fichiers `.tag.gz` (ou `.tgz`) sont des archives Unix compressées, des fichiers équivalents aux fichiers `.zip`. En standard, vous n'avez pas les outils pour pouvoir prendre en charge ce type d'archive. Pour pouvoir l'utiliser, il faudra télécharger et installer un outil de décompression comme WinRAR, WinZip ou 7-Zip. Une fois l'un de ces outils installé, vous pourrez cliquer sur l'icône de l'archive téléchargée pour en voir le contenu. Généralement, vous disposez aussi d'un menu par clic droit proposant l'option *extraire ici* (ou *extract here*) qui nous servira un peu plus loin.

Téléchargement de la traduction française de Drupal

La toute dernière version stable de *Drupal* est maintenant sur votre bureau mais malheureusement il s'agit d'une version anglaise. L'étape suivante consiste donc à récupérer le paquet des traductions pour disposer d'une version francisée. Notez cependant que les traductions de Drupal sont l'œuvre de bénévoles et qu'elle ne sont pas parfaites. De plus, ces traductions ne prennent pas en compte les modules que nous allons installer. Chaque auteur de module est libre de fournir les traductions dans les langues qui lui ont été fournies là aussi par des bénévoles. Vous trouverez plus d'informations à ce sujet dans le chapitre 10 « Gérer les langues ».

• **Translations**

Figure 4–13
La section des traductions de drupal.org

Drupal uses English by default, but may be translated to many other languages. To install these translations, unzip them and import the `.po` file through Drupal's administration interface for localization. You will need to turn on the locale module if it's not already enabled. For up to date information on where the translation infrastructure is going, please see [the localization server project](#).

Pour cela, revenons à la page précédente (section *Download*) et sélectionnons le type de ressource *Translations*. Nous trouvons sur cette page les nombreuses traductions disponibles de Drupal. Pour l'instant, concentrons-nous sur la version française (*French*) qui se trouve sur la troisième page.

Une fois repérée, comme pour *Drupal Core*, nous avons en vert les versions stables des traductions, et en rouge celles qui ne le sont pas. Préférez les versions stables en prenant soin de sélectionner la traduction adaptée à la version de Drupal que vous utilisez (ici la version 6). Nous sauvegarderons cette archive de traduction au même endroit que celle de Drupal.

BON À SAVOIR La stabilité des langues

Si vous n'avez pas de version stable pour une langue, vous pouvez prendre une version de développement sans trop de risques : au pire, il ne vous manquera que certaines traductions.

Installation des sources de Drupal

Décompression des archives

Drupal étant une application PHP, son installation ne se passe pas comme pour une application traditionnelle. Il s'agira d'abord d'installer le code source de Drupal, ce qui correspond à décompresser les archives que nous avons téléchargées, puis dans un second temps à installer les données de Drupal dans une base de données MySQL.

L'installation des sources consiste donc simplement à décompresser le fichier `drupal-6.10.tar.gz`, ce qui nous donne un dossier `drupal-6.10`. Recopiez tout le contenu de ce dossier (n'oubliez pas le fichier `.htaccess`) dans le dossier web (`c:\wamp\www` pour Windows, et `/var/lib/www/html` pour Linux). Les sources de Drupal sont maintenant en place. Nous allons y ajouter la traduction française en recopiant l'archive `fr-6.x-1.0.tar.gz` à la racine de notre dossier web et en la décompressant à cet endroit. Vous pouvez ensuite supprimer l'archive.

Comprendre la structure des dossiers

Nous avons à présent dans notre dossier web une série de dossiers dont il est important de comprendre l'usage. Même si vous n'avez aucunement l'intention d'aller modifier le fonctionnement interne de Drupal, connaître cette structure permet de ne pas poser ses fichiers (modules, thèmes, etc.) n'importe où, ce qui risquerait d'aboutir à une installation difficile à maintenir et à faire évoluer.

- `misc` : ce dossier contient divers éléments standards de Drupal comme les images, les fichiers JavaScript, etc.
- `modules` : ce dossier contient les modules par défaut de Drupal (Node, Blog, Book, Comment, etc). Ils forment le cœur des fonctionnalités Drupal.
- `themes` : ici se trouvent les quatre thèmes par défaut de Drupal, que nous étudierons un peu plus loin.

- **profils** : ce dossier contient les profils de configuration utilisés lors des installations. C'est par exemple ici qu'une partie de notre archive de traduction a été décompressée.
- **includes** : il s'agit là de l'ensemble des fonctionnalités fondamentales de Drupal (gestion du cycle de création des pages, des thèmes, des bases de données, etc.).
- **scripts** : vous trouverez ici des scripts utilitaires que nous étudierons un peu plus loin.
- **sites** : il s'agit du seul endroit où vous stockerez tout fichier ne faisant pas partie de la distribution standard de Drupal.

Le dossier **sites** est pour nous le plus important car, répétons-le, il s'agit (sans compter **profils**) du seul emplacement dans lequel nous serons amenés à travailler. Considérez d'ores et déjà tout autre dossier, y compris la racine, comme interdit de modification. Cette bonne pratique vous évitera de nombreux maux de crâne lorsque vous devrez mettre à jour votre Drupal. Si vous avez scrupuleusement respecté cette règle, vous n'aurez qu'à sauvegarder le dossier **sites**, écraser tout le reste avec la nouvelle version, et remettre **sites** à sa place pour que tout fonctionne de nouveau.

Comme nous le verrons en détail dans les prochains chapitres, dans ce dossier **sites** nous pourrions stocker :

- nos modules ;
- nos thèmes ;
- nos configurations ;
- les fichiers utilisateur attachés aux contenus.

Création du paramétrage initial

Pour l'instant, vous ne trouvez à la racine du dossier **sites** que deux répertoires : **all** et **default**. Laissons provisoirement **all** de côté pour nous concentrer sur **default**.

Ce dernier contient un seul fichier **default.settings.php**. Il s'agit de la configuration de base de notre site Drupal. Ce fichier est juste un modèle que nous allons dupliquer au même endroit sous le nom **settings.php**.

Une fois le fichier **settings.php** créé, nous devons nous assurer que le dossier **sites** et son contenu possèdent les droits d'écriture pour le service Apache, faute de quoi l'installation de Drupal ne pourra pas effectuer sa configuration automatique pour nous.

Sous Windows XP, le changement de droits se fait par un clic droit sur le dossier **sites**, en choisissant *Propriétés*. Dans l'onglet *Général*, décochez *Lecture seule* puis cliquez sur *Appliquer*.

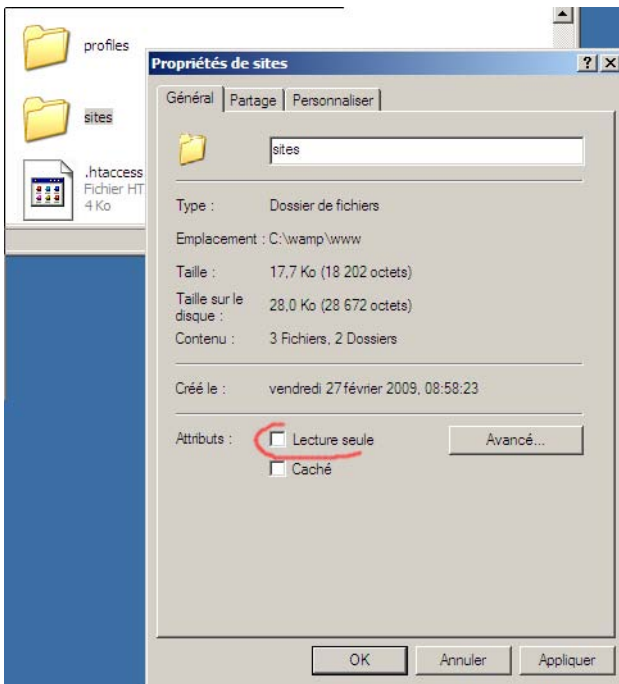


Figure 4-14
Changement des droits sous Microsoft Windows

Une confirmation vous est alors demandée pour *appliquer les modifications à ce dossier et à tous les sous-dossiers et fichiers*. Confirmez en cliquant sur **OK**, puis **OK** à nouveau.

Sous GNU/Linux, vous pouvez, dans une console, lancer la commande :

```
chmod 777 /var/www/html/sites -Rc
```

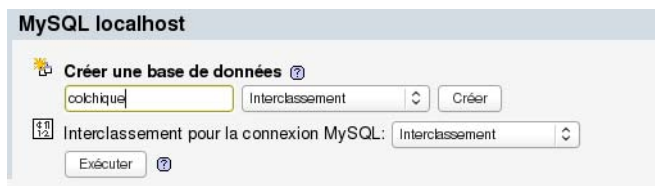
Création de la base de données

Avant d'aller plus loin, il nous faut créer la base de données qui hébergera les données de notre site. Pour cela, créez un nouvel onglet dans votre navigateur (généralement **CTRL+T**) et allez à l'adresse <http://localhost/phpmyadmin>.

Ici se trouve l'outil phpMyAdmin qui permet de créer, de visualiser et de paramétrer notre service de base de données. Si un mot de passe vous est demandé, saisissez l'identifiant root et le mot de passe que vous avez configuré plus haut.

Une fois sur la page d'accueil de phpMyAdmin, nous avons la possibilité de créer une nouvelle base de données, dans le cadre *MySQL localhost*.

Figure 4–15
Création de la base de données pour Drupal



Nous saisissons ici le nom de notre base de données dédiée à Drupal, *colchique*, et appuyons sur *Créer*. La nouvelle base de données apparaîtra sur le côté gauche de la page. Vous pouvez la sélectionner mais elle sera en toute logique vide.

Initialisation de l'installation Drupal

Les sources de Drupal étant installées, il nous reste maintenant à faire de même avec sa base de données. À partir de maintenant, tout se fait à partir d'un simple navigateur web en allant à l'adresse <http://localhost/install.php>.

Sélection de la langue

Notre installation commence par le choix de la langue d'installation qui sera aussi celle par défaut de notre site Drupal. Sélectionnez *French (Français)* et appuyez sur *Select language* pour passer à l'étape suivante.

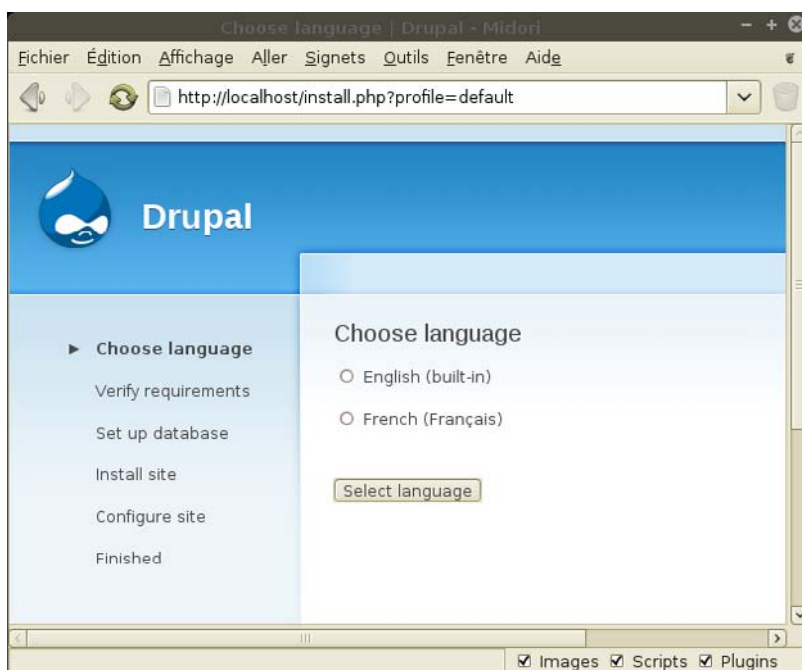


Figure 4–16
Installation de Drupal : choix de la langue

Si tout s'est bien passé, l'installation devrait sauter l'étape de vérification des prérequis. Sinon, notez l'erreur rencontrée par Drupal et reportez-vous à la section « Résolution des problèmes techniques », page 60. Lorsque le problème a été corrigé, pressez **F5** pour vérifier que Drupal ne trouve pas d'autres erreurs. Si aucune erreur ne subsiste, vous passerez automatiquement à l'étape suivante.

Configuration de la base de données

Ceci fait, appuyez sur **F5** dans le navigateur pour voir si Drupal a d'autres avertissements à nous donner. Normalement, aucun problème ne devrait être détecté, et le processus d'installation devrait pouvoir avancer à l'étape suivante.



Figure 4-17
Installation de Drupal :
configuration de la base de données

Ici, nous allons donner à Drupal les informations lui permettant de contacter la base de données. Tout d'abord, il nous faut choisir le *Type de base de données*, en l'occurrence *MySQL*.

Ensuite nous saisissons le *Nom de la base de données*, à savoir *colchique*, et enfin le *Nom d'utilisateur* et le *Mot de Passe* de l'utilisateur qui sera

Figure 4–18
Installation de Drupal :
application des traductions

autorisé à se connecter à la base de données. L'utilisateur concerné ici est root. Pour une installation WAMP, vous n'avez pas à saisir de mot de passe. Pour une installation LAMP, entrez le mot de passe que vous avez choisi lors de l'installation de MySQL.

Ceci fait, cliquez sur *Sauvegarder et poursuivre*. Une barre de progression de l'installation s'affiche alors pour vous indiquer les opérations effectuées par Drupal.



Configuration du site

Un avertissement devrait apparaître en tête de cette nouvelle page, vous indiquant que le fichier `settings.php` a bien été modifié et qu'il nous faut maintenant retirer les droits d'écriture. Il s'agit ici d'effectuer l'opération inverse de celle réalisée plus haut, en ôtant à *tout le monde* les droits sur le dossier `sites` et ses éléments enfants.

Informations sur le site

Une fois cette précaution prise, nous procédons au paramétrage du site en lui-même. Notez que rien de ce qui est fait ici n'est définitif et que tout pourra être changé plus tard en passant par l'administration de Drupal.

Figure 4–19
Installation de Drupal :
les informations sur le site

ATTENTION Choix des adresses courriel

Faites attention, toutes les adresses courriel indiquées dans cet ouvrage le sont à titre d'exemple. Lors de vos tests, vous devrez à chaque fois utiliser une adresse réelle pour pouvoir ainsi recevoir les courriels de test envoyé par Drupal. Cela peut être votre adresse personnelle ou une adresse que vous créerez pour l'occasion.

Nous commençons par les informations de base du site : le *Nom du site* ainsi que l'*Adresse e-mail du site*. Le *Nom du site* sera utilisé plus tard par Drupal en tant que titre du site, en en-tête de toute page.

Compte administrateur

Une fois le titre fixé, nous allons créer le compte de l'utilisateur privilégié de notre site Drupal.

Compte administrateur

Le compte administrateur a un accès complet au site ; il aura automatiquement toutes les permissions et pourra effectuer n'importe quelle tâche. Ce compte sera le seul à pouvoir exécuter certaines tâches, n'en abusez donc pas.

Nom d'utilisateur : *

Les espaces sont autorisés, mais pas les signes de ponctuation à l'exception des points, traits d'union et caractères de soulignement.

Adresse électronique : *

Tous les messages de ce système seront envoyés à cette adresse. Cette adresse ne sera pas rendue publique, et ne sera utilisée que si vous désirez obtenir un nouveau mot de passe ou recevoir personnellement certaines nouvelles ou avertissements.

Mot de passe : *

 Sécurité du mot de passe : **Haute**

Confirmer le mot de passe : *

 Concordance des mots de passe : **Oui**

Figure 4–20
Installation de Drupal :
réglage du compte de l'administrateur

Il n'y a aucun problème à utiliser un mot de passe simpliste et facile à retenir pour une machine de développement, mais pour un serveur public, ayez conscience que ce compte aura absolument tous les droits. Il conviendra donc d'utiliser un mot de passe complexe. Lors de la saisie, un indicateur de solidité est là pour vous aider : vous devez obtenir une sécurité de mot de passe « haute » en mélangeant majuscules, minuscules, espaces, symboles et chiffres sur 10 à 12 caractères.

Tous les paramètres du compte administrateur pourront être changés ultérieurement. Il suffira pour cela de se connecter en tant qu'administrateur, et d'aller dans *Mon compte*, puis dans l'onglet *Modifier*.

Concernant l'adresse électronique, utilisez votre propre adresse courriel. En effet, dès que l'installation sera achevée, Drupal vous enverra un courrier de confirmation indiquant que le compte a été correctement créé. Cette adresse doit donc être valide.

Fuseau horaire

Paramètres du serveur

Fuseau horaire par défaut :

Par défaut, les dates et les heures sur ce site seront affichées selon ce fuseau horaire.

Figure 4–21
Installation de Drupal : paramétrage du fuseau horaire

Le fuseau horaire est utilisé par Drupal pour dater les nouveaux contenus, les commentaires, les mises à jour, etc. Le choix du fuseau se fait ici simplement en sélectionnant dans la liste celui que vous désirez utiliser.

BON À SAVOIR Réinitialiser le mot de passe administrateur

Si vous avez perdu des identifiants, sachez que tous les comptes utilisateur sont stockés dans la table `users`.

Dans cette table, à chaque compte correspond un `id` unique (colonne `UID`). Pour Drupal, les utilisateurs anonymes et privilégiés sont systématiquement les deux premières lignes de cette table, et ont pour UID respectif 0 et 1.

Pour changer identifiant et mot de passe, il suffit d'exécuter la requête SQL suivante avec la console MySQL ou PhpMyAdmin :

```
update users set
  name='identifiant_administrateur',
  pass=md5('mot_de_passe')
where
  uid=1;
```

BON À SAVOIR

Les URL simplifiées vues par Google

Le moins qu'on puisse dire est que les URL non simplifiées ne sont pas des plus élégantes. Mais leur vrai problème nous est donné dans le guide à l'usage des webmasters rédigé par Google :

« *If you decide to use dynamic pages (i.e., the URL contains a "?" character), be aware that not every search engine spider crawls dynamic pages as well as static pages. It helps to keep the parameters short and the number of them few.* »

En somme, en utilisant ce format d'URL, nous risquons d'avoir des pages non référencées par certains automates de recherche. Ce format n'est donc pas très conseillé pour augmenter sa visibilité sur le Web.

Figure 4–22

Installation de Drupal : URL simplifiées

URL simplifiées

Comme vous le constatez dans la barre d'adresse du navigateur, Drupal utilise par défaut des adresses web de la forme `http://www.colchique-conseil.fr?q=/section/page`.

Ce qui se trouve après le `?q=` – dans notre exemple : `/section/page` –, représente le chemin interne et désigne la ressource Drupal à afficher. Le mode *URL simplifiées* permet de se passer du `?q=` pour obtenir une adresse de la forme `http://www.colchique-conseil.fr/section/page`.

Vous pouvez donc ici activer la simplification des URL. Si Drupal vous indique que ce n'est pas possible, reportez-vous à la section « Résolution des problèmes techniques », page 60.

URLs simplifiées :

☐ Désactivé

☒ Activé

Cette option indique à Drupal d'émettre des URLs "simplifiées" (c'est à dire sans le `?q=` dans l'URL).

Le test a réussi. Votre serveur supporte cette fonctionnalité.

Notification des mises à jour

Cette option permet à Drupal de vérifier régulièrement si une mise à jour a été publiée pour Drupal, ou pour l'un des modules que vous avez installés. Cette option est très importante, et sur un site de production, il ne faudra jamais oublier de l'activer. Cependant, dans le but d'apprentissage qu'est le nôtre, nous allons la décocher. Cela va nous permettre un peu plus loin de vérifier que Drupal nous prévient correctement de cette erreur de paramétrage.

Notification des mises à jour :

☐ Vérifier automatiquement les mises à jour

Figure 4–23

Installation de Drupal : désactivation des notifications de mises à jour

Fin de l'installation

Nous avons terminé cette page de paramétrages, et pouvons cliquer sur le bouton *Sauvegarder et poursuivre*.

Installation de Drupal terminée

Félicitations, Drupal est maintenant correctement installé.

Vous pouvez maintenant vous rendre sur [votre nouveau site](#).

Figure 4–24

Installation de Drupal enfin terminée !

Comme Drupal nous l'indique, notre site est maintenant opérationnel. Nous pouvons cliquer sur le lien *Votre nouveau site* pour nous en convaincre :



Si tout s'est bien passé, vous devriez recevoir le courrier de confirmation de la création du compte administrateur. Si vous ne le recevez pas, vous avez sûrement un problème dans votre chaîne d'envoi de courrier, reportez-vous à la section « Résolution des problèmes techniques » page suivante pour plus d'informations.

ASTUCE Et si on recommençait depuis le début ?

Question saugrenue de prime abord mais qui n'est néanmoins pas dénuée d'intérêt. En effet, vous avez peut-être simplement envie de vérifier ce que vous venez d'apprendre depuis le début ?

Pour recommencer l'installation, vous devez effectuer les étapes suivantes :

1. Écraser le fichier `settings.php` avec la version qui vous a été fournie par défaut `settings.default.php`.
2. Supprimer la base de données avec phpMyAdmin, en sélectionnant la base de données `colchique` et en cliquant sur l'onglet *Supprimer*.

C'est tout ce qu'il y a à faire pour recommencer l'installation de Drupal à partir de l'adresse `http://localhost/install.php`.

Figure 4–25

La page d'accueil de Drupal

ASTUCE Paramétrage du nom de domaine

Cette étape n'est en soi pas nécessaire mais pour la suite de cet ouvrage, nous n'utiliserons plus l'URL `http://localhost/` mais celle-ci : `http://www.colchique-conseil.fr`. Si vous aussi vous désirez utiliser cette seconde URL, c'est possible en ajoutant la ligne suivante dans votre fichier `hosts`.

```
www.colchique-conseil.fr127.0.0.1
```

Sous Linux, le fichier `hosts` se trouve dans `/etc/hosts`, sous Windows, `C:\WINDOWS\system32\drivers\etc\hosts`. Il suffit d'ajouter cette ligne et d'enregistrer le fichier pour que le nouveau nom soit accessible dans votre navigateur.

Résolution des problèmes techniques

Dans cette section, nous allons aborder divers problèmes techniques que nous pouvons rencontrer au cours du processus d'installation.

Problèmes de droits

Lors de la phase de vérification des prérequis, si vous obtenez le message suivant :

Figure 4-26
Premier message d'erreur lié aux droits lors de l'installation

Changer les droits des fichiers pour qu'ils soient autorisés en écriture par le serveur web. Si vous ne savez pas comment attribuer les droits, consultez le [manuel en ligne](#).

ou encore :

Figure 4-27
Second message d'erreur lié aux droits lors de l'installation

• Le répertoire `sites/default/files` n'existe pas. L'essai de création automatique de ce répertoire a échoué, probablement du fait d'un problème lié aux permissions. Pour procéder à l'installation, vous pouvez soit créer un répertoire et régler ses permissions manuellement, ou assurez-vous que l'installateur possède les permissions nécessaires pour le créer automatiquement. Pour plus d'information, consultez le fichier `INSTALL.txt` ou le [manuel en ligne](#).

... vous n'avez probablement pas correctement paramétré les droits d'écriture à Apache pour le dossier `sites`. Reportez-vous à la section « Initialisation de l'installation », rubrique « Création du paramétrage initial » pour plus d'informations.

Mauvais fichier de configuration

Si vous obtenez lors de la phase de validation des prérequis le message d'erreur suivant :

Figure 4-28
Message d'erreur lié à l'absence de `settings.php`

Copier le fichier `./sites/default/default.settings.php` en `./sites/default/settings.php`.

... vous avez sans doute oublié de recopier le modèle configuration vers une configuration par défaut. Reportez-vous à la section « Initialisation de l'installation », à la rubrique « Création du paramétrage initial » pour plus d'informations.

Impossible d'activer les URL simplifiées

Il est possible que Drupal vous interdise le choix *Activé* pour cette option, ce qui se traduit par le message suivant :

La configuration de votre système ne supporte pas cette fonctionnalité. La [page du manuel sur les URLs simplifiées](#) apporte une aide supplémentaire.

Figure 4–29

Message d'erreur portant sur l'impossibilité d'activer les URL simplifiées

La raison est généralement une incompatibilité entre cette option et le paramétrage du service Apache. En effet, le mode URL simplifiées utilise le module Apache `mod_rewrite` pour réécrire les requêtes avant de les transmettre à Drupal. Le principe est le suivant : `mod_rewrite` transforme toute requête de la forme `http://www.conchique-conseil.fr/a/b/c` en `http://www.conchique-conseil.fr?q=a/b/c`, permettant ainsi à Drupal de retomber sur son mode natif (avec `?q=`) .

Les règles de réécriture de requêtes sont stockées dans le fichier `.htaccess` qui se trouve à la racine du site Drupal. Si le mode n'est pas activable, soit le module Apache `mod_rewrite` n'est pas installé, soit il est installé mais l'utilisation des fichiers `.htaccess` est interdite par votre configuration Apache.

Dans les deux cas, la solution se trouve dans le fichier `httpd.conf` (situé dans `/etc/httpd/conf` pour GNU/Linux, et dans `Apache/httpd.conf` pour l'environnement WAMP). Il s'agit d'y vérifier que le module `mod_rewrite` est bien chargé :

```
LoadModule rewrite_module modules/mod_rewrite.so
```

Ensuite, cherchez un peu plus loin dans la configuration la directive `Directory` portant sur `c:\wamp\www` et vérifiez que vous avez bien une commande `AllowOverride All`.

```
<Directory "/var/www/html">
...
AllowOverride All
...
</Directory>
```

Lorsque vous aurez effectué les modifications, vous pourrez redémarrer le service Apache et rafraîchir la page de l'installation Drupal sur votre navigateur. Vous devriez maintenant avoir la possibilité d'activer cette option. Si Drupal vous affiche une erreur indiquant que le site est déjà installé, placez le curseur de la souris à la fin de l'adresse courante dans la barre d'adresse du navigateur, et appuyez sur *Entrée* pour recharger la page.

Figure 4–30
Message d'erreur lié à
une base de données non accessible

Problème de base de données

Lorsque vous avez saisi le nom de votre base de données ainsi que vos identifiant et mot de passe, si vous obtenez l'erreur suivante :

Impossible de sélectionner votre base de données sur le serveur MySQL, ce qui signifie que le nom d'utilisateur et le mot de passe sont bons, mais qu'il y a un problème pour accéder à vos données. MySQL a répondu : *Unknown database 'site_xyq'*.

... vous devez vérifiez dans l'ordre :

- 1 que le service de base de données est bien démarré. Pour cela, allez à l'URL <http://localhost/phpmyadmin> ;
- 2 que le type de base de données choisi est bien le bon (*mysql*) ;
- 3 que vous avez bien créé votre base de données. Pour cela se reporter à la section « Initialisation de l'installation », rubrique « Création de la base de données » ;
- 4 que le nom de l'utilisateur est bien *root* et que le mot de passe est bien celui que vous avez configuré. Si vous n'avez configuré aucun mot de passe (cas d'une installation Windows), ce champ doit être vide.

Courrier de confirmation non reçu

Si vous ne recevez pas le courrier de confirmation de votre compte administrateur, il s'agit sûrement d'un problème de paramétrage dans l'envoi des courriers du côté de PHP.

Sous GNU/Linux, ceci ne pose pas de problème, le courrier émis par PHP étant pris en charge par la commande *sendmail*. Si votre GNU/Linux est correctement configuré, par exemple par l'installation d'un service PostFix local, les messages devraient partir correctement. Vous avez cependant la possibilité d'utiliser un service de courrier externe (SMTP) en utilisant l'outil *smtp*. Un tutoriel sur cette technique est disponible à l'adresse <http://artisan.karma-lab.net/node/1171>.

Sous Windows, en revanche, il est important d'indiquer à PHP quel est le service de courrier utilisé. Vous utiliserez pour cela l'adresse du service qui vous aura été donnée par votre fournisseur d'accès à l'Internet : allez dans le menu WAMP et sélectionnez *PHP/php.ini*. Notepad devrait s'ouvrir avec la configuration de PHP. Recherchez la section *mail function* et modifiez comme suit :

```
[mail function]
; For Win32 only.
SMTP = adresse_SMTP_FAI
smtp_port = 25
```

Il suffit maintenant de quitter Notepad en sauvegardant la configuration de PHP, puis de redémarrer les services WAMP.


En résumé

- Pour fonctionner, Drupal a besoin de PHP 5.0 et d'un service HTTP (Apache ou IIS feront l'affaire).
- Toutes les ressources pour Drupal (Drupal lui-même, les traductions, les modules, les thèmes) sont disponibles sur le site <http://drupal.org>.
- Si vous avez des questions à propos de l'installation, n'hésitez pas à les poser sur le forum en français dédié à Drupal, à l'adresse <http://drupalfr.org/forum> dans la rubrique *Avant installation*.

BON À SAVOIR Comment poser une question sur un forum ?

- Commencez par faire une recherche sur votre moteur préféré (Google, Yahoo!, etc.). La plupart des informations sont en anglais, donc si vous le pouvez, formulez votre recherche en français, puis en anglais.
- Si vous ne trouvez rien, posez votre question sur le forum (après avoir vérifié qu'aucune autre question similaire n'a déjà été posée). Soyez certain qu'il n'en existe pas de mauvaise. Poser une question, c'est aussi une manière de contribuer au projet Drupal car les questions (ainsi que leurs réponses) seront un jour indexées dans les moteurs de recherche.
- Soyez précis dans vos formulations. Rédigez des phrases complètes, en français (bannissez tout langage abrégé, SMS, etc.) et compréhensibles par tous. Donnez toujours le contexte de votre problème (version exacte de Drupal, nom et version des modules, etc.).
- Lorsqu'un contributeur vous aura donné la bonne réponse, remerciez-le et modifiez le sujet de votre question pour faire précéder le titre de *[Résolu]*.

chapitre 5



Colchique

Accueil • Administrer

administrateur

- Mon compte
- ▷ Créer un contenu
- ▽ Administrer
 - ▷ Construction du site
 - ▷ Gestion du contenu
 - ▷ Configuration du site
 - ▷ Gestion des utilisateurs
 - ▷ Rapports
 - Aide
- Se déconnecter

Administrer

Par tâchePar module

Cette page liste les tâches d'administration disponibles pour chaque module.

Block

Contrôle l'affichage des boîtes apparaissant en marge du contenu principal.

- Configurer les droits d'accès
- Blocs
- Aide

Database logging

Consigne les événements systèmes dans la base de données.

- Entrées récentes du journal
- Journalisation en base de données
- Principales erreurs de type 'accès refusé'
- Principales erreurs de type 'page non trouvée'
- Aide

Comment

Permet aux utilisateurs de commenter et de discuter le contenu publié.

- Configurer les droits d'accès
- Commentaires
- Aide

Filter

Gère le filtrage du contenu avant son affichage.

- Configurer les droits d'accès
- Formats d'entrée
- Aide

Menu

Permet aux administrateurs de personnaliser le menu de navigation du site.

Administrer Drupal et ses modules

La page de bienvenue de Drupal nous propose de commencer le voyage par la configuration de notre site web. Cela peut paraître étonnant mais ne manque pourtant pas de sens. Nous allons en effet passer beaucoup de temps dans cette zone du produit pour créer du contenu, modifier les réglages prédéfinis à l'installation et bâtir la structure des pages. Autant se familiariser le plus rapidement possible avec l'interface d'administration, et en appliquer les principes à une tâche qui reviendra souvent : l'installation de nouveaux modules.

SOMMAIRE

- Premiers regards sur l'administration de Drupal
- Les modules : en trouver et en ajouter
- Simuler le cron de Drupal

Premier regard sur l'administration

Accéder à l'administration

Le reproche le plus fréquemment adressé à Drupal est lié à son interface d'administration, jugée moins accessible que celle de Joomla. C'est assez difficile à nier, mais il est important de comprendre que ce paramétrage complexe au premier abord est la rançon de possibilités étendues.

Pour accéder aux sections d'administration, il suffit simplement de cliquer sur le lien *Administrer* à gauche de l'écran, dans la zone marquée *administrateur*.



Figure 5-1
Le menu des actions réalisables
par l'administrateur

La page s'affichant à présent est dense, aussi, pour bien se repérer, est-il primordial d'en comprendre la structure. Celle-ci se compose de cinq sections ayant chacune un rôle précis.

La construction du site



Figure 5-2
La section de l'administration dédiée
à l'ajout de fonctionnalités

Comme son nom le laisse présager, cette section regroupe tout ce qui permet d'ajouter des fonctionnalités au site. C'est ici que seront gérés les thèmes visuels, les traductions de l'interface, les nouveaux menus, les blocs à disposer autour d'un contenu, et enfin les modules.

Même si nous y reviendrons rapidement, retenons pour le moment qu'un module est un greffon (plug-in) pour Drupal, le dotant de nouvelles fonctionnalités.

La configuration du site

De manière générale, vous passerez beaucoup de temps dans cette section lors de la phase de construction de votre site (d'où le titre), et beaucoup moins une fois que ce dernier sera en production.

Configuration du site
Ajuster la configuration de base du site.

Figure 5-3
La section dédiée au paramétrage
des fonctionnalités

Contrairement à *Construction du site* où s'effectue l'ajout de nouvelles fonctionnalités visuelles ou non, cette section permet de gérer tous les réglages des fonctions existantes. Ici nous n'ajoutons donc rien, nous ne faisons que paramétrer.

Cela comprend un choix d'options comme le format des date et heure, le titre du site, l'activation ou la désactivation des URL simplifiées, etc. Vous y retrouverez donc beaucoup de réglages que vous avez définis lors de l'installation de Drupal.

Ici aussi, la majorité du temps passé le sera au moment de la fabrication de votre site. Plus tard, vous y reviendrez surtout lorsque vous devrez configurer de nouveaux modules que vous aurez ajoutés.

La gestion des utilisateurs

Gestion des utilisateurs
Gérer les utilisateurs de votre site, les groupes et l'accès aux fonctionnalités.

Figure 5-4
La section dédiée à la gestion des utilisateurs

Ici pas de mystère : vous allez pouvoir gérer vos utilisateurs (noms, mots de passe, etc.), ainsi que les permissions qui leur seront données sur les différentes fonctionnalités du site. Notez seulement que nous parlons bien d'utilisateurs et non de contenus générés par les utilisateurs (articles, commentaires, etc.), qui eux sont pris en charge par la section suivante.

Mis à part la définition des rôles généralement réalisée pendant la phase de développement, nous sommes ici dans une section qui est davantage utilisée pendant la phase d'exploitation de votre site, une fois qu'il aura été mis en production.

La gestion du contenu

Gestion du contenu
Gérer le contenu de votre site.

Figure 5-5
La section dédiée à la gestion des contenus

Vous trouverez ici tout ce qui concerne le contenu au sens CMS du terme, c'est-à-dire la création de nouveaux types de contenu, la gestion des commentaires et des contenus déjà écrits, la taxonomie, etc.

Figure 5–6
La section permettant d’accéder à tous les rapports sur le fonctionnement de Drupal

Comme pour la section *Gestion des utilisateurs*, une fois vos types de contenu et vos taxonomies réglés, vous passerez beaucoup de temps dans cette section après la mise en ligne de votre site pour gérer les contenus existants et les commentaires associés.

Les rapports

Rapports
Voir les rapports des journaux du système et d'autres informations.

Cette section vous donne accès à une série de pages permettant de connaître l’état de Drupal. Vous y trouverez, par exemple, la liste des pages en erreur, le journal des erreurs émises par les modules, ou plus important, le tableau de bord vous permettant d’avoir une vision générale de la bonne marche du système.

Terminer l’installation de Drupal

Utilisation du tableau de bord

Intéressons-nous à présent au message qui se trouve en haut de la page d’administration et qui devrait ressembler à ceci :

Figure 5–7
Message prévenant l’administrateur d’un mauvais paramétrage

Un ou plusieurs problèmes ont été détectés dans votre installation de Drupal. Vérifiez le [tableau de bord](#) pour plus d’information.

Par ce message, Drupal nous indique qu’il a détecté des problèmes d’installation. Pour savoir de quoi il en retourne, nous pouvons soit cliquer sur le lien *Tableau de bord* du message, soit – ce qui revient strictement au même – sur le lien *Tableau de bord* de la section *Rapports*.

Sur cette page apparaît une liste de points que Drupal a cherchés à vérifier. Ici nous avons deux erreurs (fond rouge) et une alerte (fond jaune). Le premier point porte sur le fait que nous avons oublié d’enlever à Apache les droits en écriture sur le dossier `sites/default` et le fichier `sites/default/settings.php`. Nous pouvons corriger cela très facilement comme nous l’avons vu au chapitre précédent. Il suffit alors de rafraîchir la page pour vérifier que la modification est bien validée par Drupal.

Drupal	6.13
✓ Accès au fichier update.php	Protégé
✓ Base de données PostgreSQL	8.3.7
✗ Fichier de configuration	Non protégé. Le répertoire <code>sites/default</code> n'est pas protégé en écriture. Cela pourrait poser un problème de sécurité, vous devez donc changer les permissions du fichier pour qu'il ne soit pas accessible en écriture. Le fichier <code>sites/default/settings.php</code> n'est pas protégé en écriture et cela pose un problème de sécurité. Vous devez changer les permissions du fichier pour qu'il ne soit pas accessible en écriture.
✓ Librairie GD	2.0 or higher
✓ Librairie Unicode	Extension PHP mbstring
✓ Mise à jour de la base de données	À jour
⚠ Notification des mises à jour	Non activé Les notifications de mise à jour ne sont pas activées. Il est fortement recommandé d'activer le module <code>update status</code> depuis la page d'administration des modules pour rester informé des mises à jour. Pour plus d'information veuillez lire la page Update status du manuel.
✓ PHP	5.2.9
✓ PHP 'register globals'	Désactivé
✓ Plafond mémoire de PHP	256M
✓ Serveur web	Apache/2.2.11 (Mandriva Linux/PREFORK-10.1mdv2009.1)
✓ Système de fichiers	Inscriptible (méthode de téléchargement <i>publique</i>)
✗ Tâches de maintenance cron	Dernière exécution, il y a 4 semaines 2 jours Cron n'a pas été lancé récemment. Pour plus d'informations reportez-vous à l'entrée sur la configuration du Cron du manuel en ligne. Vous pouvez lancer le cron manuellement .

✓ Fichier de configuration	Protégé
----------------------------	---------

Figure 5–9
Les fichiers sont maintenant protégés.

Figure 5–8
La liste des points vérifiés par Drupal, et des erreurs détectées

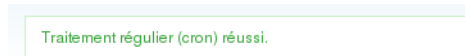
La tâche de maintenance cron

La deuxième erreur détectée par Drupal porte sur « cron » qui n'aurait jamais été exécuté. Un peut déroutant tout de même si l'on ne sait pas ce qu'est ce cron...

Cron est un mécanisme de Drupal lui permettant d'effectuer des actions à intervalles de temps réguliers, comme envoyer des bulletins d'informations ou des courriels de notification, nettoyer la base des messages d'erreur, vérifier si votre système est à jour, etc. En bref, cron est indispensable au bon fonctionnement de Drupal, ce qui explique que son dysfonctionnement détecté ici soit considéré comme une erreur grave.

De manière plus pragmatique, cron est un simple script PHP appelé `cron.php` qui se trouve à la racine, à côté d'`index.php`. Pour lancer les

Figure 5–10
Cron s’est correctement exécuté.



Et très logiquement, le message d’erreur du tableau de bord a disparu, vous indiquant la dernière fois que cron a été exécuté et vous redonnant le lien pour le lancer à nouveau.

Figure 5–11
L’erreur cron a maintenant disparu.



HISTOIRE D’où vient cron ?

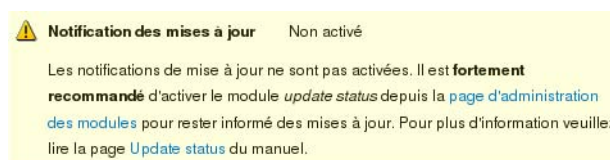
Cron est à l’origine un programme Unix qui fonctionne en permanence (ce genre de programme est appelé un démon, *daemon* en anglais) et qui exécute des tâches programmées à intervalles de temps réguliers. C’est l’équivalent du planificateur de tâches sous Microsoft Windows. Le nom cron en lui-même vient de Chronos, qui n’est autre que le dieu grec personnifiant le temps.

Bien évidemment, lancer le cron manuellement n’est pas une option valable lorsque notre site sera en production. Nous verrons donc au chapitre « Mettre son site en production » comment automatiser ce processus. Mais pour notre plate-forme de développement, cette méthode manuelle convient parfaitement. Elle nous permet en effet de tester les tâches de maintenance sans avoir à attendre qu’elles s’exécutent toutes seules.

Ainsi, tout au long de cet ouvrage, et à chaque fois que cela sera nécessaire, nous reviendrons sur le tableau de bord pour lancer nous-même la tâche cron.

La gestion des modules

Figure 5–12
Une demande insistante pour activer le module Update status



Il nous reste encore une erreur, ou plutôt un avertissement, dans notre tableau de bord, nous enjoignant d’activer un certain module Update status. Voyons maintenant comment faire une telle chose.

Activer et désactiver un module

Les modules sont la composante fondamentale de Drupal, il est donc important de comprendre comment les utiliser.

Dans le message d'erreur sur la *Notification des mises à jour*, le tableau de bord nous propose un lien direct vers la page d'*Administration des modules*. Nous pourrions l'utiliser mais il est plus intéressant d'apprendre le chemin normal qui mène à ce lieu où nous allons passer beaucoup de temps.

L'administration des modules fait partie de la section *Construction du site*. Nous pouvons y accéder en utilisant à nouveau le menu *Administrateur* sur le côté droit et en cliquant sur *Construction du site*. Le menu se déroule alors pour faire apparaître les différents aspects de la construction, dont la section *Modules* sur laquelle nous cliquons.

Sur cette nouvelle page apparaît la liste des modules installés par défaut avec Drupal. Comme vous le voyez, ils sont regroupés en deux catégories : *Core – facultatif* et *Core – obligatoire*.

ERGONOMIE Les zones dépliantes

Les catégories sont des zones qu'il est possible de déplier et de replier en cliquant sur le triangle à gauche du libellé. Cette ergonomie est utilisée un peu partout dans Drupal. Par défaut, certaines zones sont dépliées et d'autres non. Il s'agit généralement de différencier les options avancées (repliées) des standards (dépliées).



Activé	Nom	Version	Description
<input checked="" type="checkbox"/>	Block	6.13	Contrôle l'affichage des boîtes apparaissant en marge du contenu principal.
<input checked="" type="checkbox"/>	Filter	6.13	Gère le filtrage du contenu avant son affichage.
<input checked="" type="checkbox"/>	Node	6.13	Permet de soumettre du contenu au site et de l'afficher sur les pages.
<input checked="" type="checkbox"/>	System	6.13	Permet aux administrateurs de modifier la configuration générale du site.
<input checked="" type="checkbox"/>	User	6.13	Gère l'inscription des utilisateurs et le système d'identification.

Figure 5-13
Les modules du cœur de Drupal

Core (en français, cœur) fait référence aux fonctionnalités de base de Drupal. Nous avons ainsi les modules de base obligatoires, gérant les fonctions fondamentales d'un CMS.

Block : nous allons bientôt découvrir ce module, les blocs ayant un rôle important dans Drupal pour créer la présentation de notre site. Ce module est responsable de leur affichage.

Filter : ce module sera étudié plus en détail dans le chapitre 7, section « Ajouter de nouveaux filtres ». Son rôle est de permettre la prise en charge de syntaxes particulières (HTML, Wiki, etc.) pour le corps des contenus et des commentaires.

Node : un module absolument central car il est chargé de la gestion des contenus eux-mêmes. En vocabulaire Drupal, un contenu s'appelle d'ailleurs un *node* (nœud).

User : le module qui gérera les utilisateurs authentifiés, c'est-à-dire ceux qui sont connus du site et qui s'y connectent par un identifiant et un mot de passe.

La case à cocher qui précède chaque module indique si ce dernier doit être activé ou désactivé. Comme vous le voyez, beaucoup de modules

BON À SAVOIR Les modules et le français

Chaque module est fourni avec un certain nombre de traductions disponibles. Lorsque l'on active un module, Drupal y cherche une langue correspondant à la langue courante, ici le français. S'il la trouve, il l'installe. Le module activé sera donc disponible en français. Cependant, beaucoup de modules ne sont pas traduits, nous verrons un peu plus loin comment effectuer cette traduction nous-même.

Figure 5-14

Nous avons la confirmation que notre Drupal est bien à jour.

facultatifs sont désactivés. Évidemment, ceux qui sont dans la catégorie *Core – obligatoire* sont tous activés.

Dans la catégorie *Core – facultatif*, nous trouvons vers la fin de la liste le module *Update status*. C'est ce module que le tableau de bord nous reprochait de ne pas avoir activé. Cochez donc la case qui le précède et en bas du formulaire, appuyez sur le bouton *Enregistrer la configuration*.

Lorsque la page se rafraîchit, le module Update status reste coché, il est donc activé. Nous venons d'activer notre premier module et si nous retournons sur le tableau de bord, ce dernier nous en remercie en nous affichant un nouvel avertissement, cette fois pour indiquer qu'il nous faut relancer cron. En effet, le module Update status ajoute à cron une nouvelle tâche de maintenance consistant à chercher sur le site drupal.org la dernière version de Drupal pour vérifier que nous possédons bien la toute dernière version. Lancez donc à nouveau le cron manuellement, et cette fois, tous les indicateurs doivent être au vert.



Vous voyez ainsi le premier intérêt de ce module que nous venons d'activer : nous tenir en permanence informé des nouvelles versions de Drupal.

Trier l'administration par modules

Le problème lorsque l'on active un nouveau module est qu'il n'est pas aisé de savoir comment le paramétrer. Fort heureusement, il existe une solution élégante à ce problème qui consiste à trier l'administration non plus par grandes sections, mais par modules. Pour ce faire, retournez sur la page d'administration. En haut de la page, vous trouverez deux onglets. Par défaut, et c'est le mode que nous utilisons jusqu'alors, l'onglet sélectionné est *Par tâche*. Cliquez maintenant sur l'onglet *Par module*.



Figure 5-15

Tri de l'administration par modules

Sur cette nouvelle page, nous trouvons une liste des modules disposés en sections sous lesquelles sont disponibles les actions d'administration spécifiques à chaque module. Il est maintenant aisé de chercher le module que nous venons d'activer pour découvrir les entrées ajoutées.

Update status

Vérifie la disponibilité de mises à jour de Drupal et de vos modules et thèmes installés.

- [Mises à jour disponibles](#)
- [Aide](#)

Figure 5-16

La section d'administration dédiée au module Update status

Nous avons ici deux menus ajoutés, *Mises à jour disponibles* et *Aide*. Le second est disponible pour chacun des modules, et en cliquant dessus, vous pouvez savoir ce que chacun fait exactement.

Update status

Le module Update status vérifie périodiquement la disponibilité de nouvelles version pour le logiciel de votre site (incluant les modules tiers et les thèmes) et vous alerte des nouvelles mises à jour.

Figure 5-17

Information sur le module Update status

Cette page contient de précieux renseignements et fournit des liens vers les différentes fonctions du module. À la fin de cette page, une section reprend la liste des fonctions d'administration publiées par le module. Nous y trouvons le lien *Mise à jour disponibles* sur lequel nous allons cliquer.

ERGONOMIE De l'aide pour l'administration

Dans le menu d'administrateur, l'aide est aussi disponible si vous dépliez le menu *Administrer*.

Accueil • Administrer • Rapports

Mises à jour disponibles

Liste Paramètres

Ici vous pourrez trouver de l'information à propos des mises à jour disponibles pour vos modules et thèmes installés. Veuillez noter que chaque module ou thème fait partie d'un "projet", lequel peut ou non avoir le même nom et peut inclure plusieurs modules ou thèmes.

Afin d'étendre les fonctionnalités ou de changer l'apparence de votre site, un certain nombre de [modules](#) tiers et [thèmes](#) sont disponibles.

Dernière vérification : il y a 16 min 12 sec ([Vérifier manuellement](#))

Drupal core

Drupal 6.13

À jour ✓

Contient : *Block, Book, Color, Comment, Contact, Database logging, Filter, Garland, Help, Locale, Menu, Node, OpenID, PHP filter, Path, Search, System, Taxonomy, Trigger, Update status, Upload, User*

Figure 5-18 Le rapport du module Update status

Nous voyons ici, dans la zone en haut de la page appelée « fil d'Ariane » (*breadcrumb* en anglais), que nous sommes dans la section *Rapports* de l'administration.

Les modules standards

Avant d'aller plus loin et voir comment ajouter de nouveaux modules, il est intéressant d'apprendre à connaître ceux que nous avons en standard. Nous allons profiter de cette exploration pour activer un certain nombre

À NOTER Les versions de ce livre

Ne vous inquiétez pas si les versions des modules ne sont pas les mêmes sur votre écran : Drupal est un produit très vivant qui évolue rapidement et il y aura donc sûrement eu des mises à jour depuis la rédaction de ce livre, comme il y en a eu pendant sa rédaction.

Figure 5–19
Les modules standards de Drupal,
d’Aggregator à Contact

<input type="checkbox"/>	Aggregator	6.13	Agrège du contenu syndiqué (flux RSS, RDF et Atom).
<input type="checkbox"/>	Blog	6.13	Permet de maintenir des pages web d'utilisateurs ou des blogs tenus à jour facilement et régulièrement.
<input type="checkbox"/>	Blog API	6.13	Permet aux utilisateurs de poster en utilisant les applications qui supportent les API XML-RPC.
<input checked="" type="checkbox"/>	Book	6.13	Permet aux utilisateurs de structurer les contenus du site de manière hiérarchique.
<input checked="" type="checkbox"/>	Color	6.13	Permet à l'utilisateur de modifier le schéma de couleur de certains thèmes.
<input checked="" type="checkbox"/>	Comment	6.13	Permet aux utilisateurs de commenter et de discuter le contenu publié. Requis par : Forum (disabled), Tracker (disabled)
<input checked="" type="checkbox"/>	Contact	6.13	Permet l'utilisation de formulaires de contact à la fois pour l'ensemble du site et pour chaque utilisateur.

Aggregator : ce module nous permettra de combiner des flux RSS provenant d'autres sites que le nôtre, et de disposer les résultats sous la forme de listes navigables. À bien des égards, ce module permet de fabriquer ce que l'on appelle des « planets », c'est-à-dire des sites dont l'objectif est de synthétiser l'activité d'une myriade d'autres sites axés sur un sujet donné, comme le fait par exemple <http://planet-libre.org> ou encore le planet de Drupal France, <http://planete.drupalfr.org/>.

Blog et **Blog API** : ces modules permettent de transformer Drupal en un système de blog permettant d'accueillir plusieurs bloggeurs comme le ferait un site tel que <http://overblog.com>.

Book : ce module permet de créer des contenus de type « page d'un livre ». Ces contenus ont la caractéristique de pouvoir être organisés les uns par rapport aux autres sous la forme de chapitres, préfaces, etc. C'est le module idéal pour créer une documentation collaborative.

Color : ce module qui doit être activé fournit des fonctionnalités au moteur de thème que nous étudierons dans le chapitre 11 « Modifier l'apparence ».

Comment : il s'agit là d'un module quasiment obligatoire permettant aux utilisateurs de commenter les contenus.

Contact : ce module va ajouter des formulaires de contact à notre site ainsi qu'aux utilisateurs. Il faut l'activer.

Content translation et **Locale** : nous l'étudierons un peu plus loin dans ce livre, ces modules permettent de disposer d'un même contenu et d'une interface en plusieurs langues.

Database logging : ce module, activé par défaut, prend en charge les traces d'exécution laissées par les autres modules.

Forum : ce module ajoute à Drupal la capacité d'accueillir un système de forum complet comme le proposerait phpBB.

À NOTER Modules magiques

Le module Forum de Drupal est par défaut assez simple. Notez cependant qu'il peut être étendu considérablement par d'autres modules tels que *Advanced forum* (http://drupal.org/project/advanced_forum) qui vise à le faire approcher du célèbre phpBB.

Menu : un autre module presque obligatoire vous permettant de créer vos propres menus semblables au menu administrateur.

<input type="checkbox"/>	Content translation	6.13	Permet au contenu d'être traduit dans plusieurs langues. Dépend de : Locale (enabled)
<input checked="" type="checkbox"/>	Database logging	6.13	Consigne les événements systèmes dans la base de données.
<input type="checkbox"/>	Forum	6.13	Rend possible les fils de discussion à propos de sujets généraux. Dépend de : Taxonomy (enabled), Comment (enabled)
<input checked="" type="checkbox"/>	Help	6.13	Permet de gérer l'affichage de l'aide en ligne.
<input checked="" type="checkbox"/>	Locale	6.13	Ajoute des fonctionnalités de gestion des langues et permet la traduction de l'interface utilisateur. Requis par : Content translation (disabled)
<input checked="" type="checkbox"/>	Menu	6.13	Permet aux administrateurs de personnaliser le menu de navigation du site.

Figure 5–20
Les modules standards de Drupal, de Content translation à Menu

<input type="checkbox"/>	OpenID	6.13	Permet aux utilisateurs de s'identifier sur votre site avec leur OpenID.
<input checked="" type="checkbox"/>	Path	6.13	Permet aux utilisateurs de renommer les url.
<input checked="" type="checkbox"/>	PHP filter	6.13	Permet à du code PHP d'être évalué à l'intérieur des contenus.
<input type="checkbox"/>	Ping	6.13	Alerte d'autres sites quand votre site a été modifié.
<input type="checkbox"/>	Poll	6.13	Permet à votre site de collecter des votes sur différents sujets sous la forme de sondages.
<input type="checkbox"/>	Profile	6.13	Gère des profils utilisateurs configurables.
<input checked="" type="checkbox"/>	Search	6.13	Permet la recherche par mots-clés dans l'ensemble du site.
<input type="checkbox"/>	Statistics	6.13	Enregistre les statistiques d'accès à votre site.
<input type="checkbox"/>	Syslog	6.13	Enregistre les événements systèmes dans le syslog.

Figure 5–21
Les modules standards de Drupal, de OpenID à Syslog

OpenID : ce module permet de s'authentifier via OpenID. Il sera abordé en détail dans le chapitre 12 « Ajouter des fonctionnalités ». Laissez-le désactivé pour l'instant.

Path : ce module ajoute à Drupal la capacité de renommer les URL des contenus pour les rendre plus agréables et faciles à lire. Il doit être activé.

PHP filter : ce module permet de saisir du code PHP dans certains contenus. Il sera abordé dans le chapitre 7, section « Le format d'entrée PHP ». Laissez-le désactivé.

Ping : ce module permet de créer des réseaux de serveurs qui se notifient les uns les autres des nouveaux contenus présent.

Poll : permet la création de questionnaires en ligne.

Profile : ce module ajoute des champs supplémentaires à la fiche d'un utilisateur comme son site web, son adresse, etc.

Search : un module très important à activer, qui permet de proposer une recherche dans les contenus.

Figure 5–22
Les modules standards de Drupal,
de Taxonomy à Upload

<input checked="" type="checkbox"/>	Taxonomy	6.13	Permet la classification du contenu. Requis par : Forum (disabled)
<input type="checkbox"/>	Throttle	6.13	Assure un mécanisme d'autorégulation permettant de contrôler l'encombrement du site.
<input type="checkbox"/>	Tracker	6.13	Permet le suivi des dernières contributions des utilisateurs. Dépend de : Comment (enabled)
<input checked="" type="checkbox"/>	Trigger	6.13	Permet à des actions d'être lancées suite à certains événements systèmes, par exemple q contenu est créé.
<input checked="" type="checkbox"/>	Update status	6.13	Vérifie la disponibilité de mises à jour de Drupal et de vos modules et thèmes installés.
<input checked="" type="checkbox"/>	Upload	6.13	Permet aux utilisateurs d'envoyer des fichiers et de les attacher au contenu.

BON À SAVOIR Modules et dépendances

Les modules de Drupal peuvent dépendre les uns des autres. Chaque module peut ainsi déclarer de quels modules il est dépendant, et sans lesquels il ne peut fonctionner.

Si le module A dépend du module B, Drupal affichera en dessous de la description du module A *Dépend de : A* et sous la description du module B *Requis par A*. Si le module dépendant ou requis n'est pas activé, ce sera indiqué par (*disabled*) ou par (*enabled*) dans le cas contraire, et par (*missing*) s'il n'est pas installé.

Si vous activez un module qui dépend d'autres modules, Drupal vous proposera systématiquement d'activer tous les modules dont votre module a besoin et qui ne le sont pas encore. En revanche, si votre module dépend d'un module qui n'est pas installé, Drupal désactivera la case à cocher permettant de l'activer.

Taxonomy : comme son nom l'indique, ce module est responsable de la gestion des taxons, du vocabulaire associé et de leurs interactions avec les contenus.

Throttle : ce module permet de désactiver d'autres modules si la fréquentation sur le site est trop forte. Nous étudierons ce module au chapitre 15, « Mettre son site en production ».

Tracker : ce module permet de suivre les dernières contributions (commentaires, contenus, etc.) sur le site.

Trigger : le principe de ce module est de vous permettre de déclencher des actions à partir d'événements. Nous étudierons ce module en détail dans le chapitre 13, « Automatiser les tâches récurrentes ». Pour l'heure, activez-le.

Update status : nous ne présentons plus ce module qui doit évidemment être activé.

Upload : un module très important qu'il convient d'activer afin de pouvoir ajouter des fichiers aux contenus (attachements).

Téléchargement d'un nouveau module

Les modules disponibles en standard dans Drupal sont bien loin d'être les seuls que vous pourrez utiliser. Pour notre première installation de modules, nous nous intéresserons au module Administration menu. Ce module permet de simplifier énormément l'administration d'un site Drupal en dupliquant l'ensemble de la section *Administration* sous la forme d'un menu déroulant qui restera en permanence en haut de chaque page.

Si vous retournez sur la page par laquelle vous avez obtenu les sources de Drupal (<http://drupal.org/project>) mais en cliquant cette fois-ci sur le lien *Modules*, vous aurez une idée des possibilités dans ce domaine.

• Modules

Contributed (contrib) modules are plugins for Drupal that extend, build or enhance Drupal core functionality. Use matching **versions** (modules released for Drupal 5.x will not work for Drupal 6.x). Contributed modules are not part of Drupal core releases and may or may not have optimized code/functionality. If a module solves your needs please consider **joining forces** and helping the maintainer.

Figure 5-23 La section de drupal.org dédiée aux modules

Chaque lien sur cette page mène à la page du projet d'un module. Si nous nous déplaçons vers *Administration menu* et que nous cliquons sur le titre, nous arrivons sur la page du projet de ce module, http://drupal.org/project/admin_menu.

Ces pages se composent toujours de la même manière avec tout d'abord un texte explicatif (en anglais), suivi des liens de téléchargement des différentes versions, puis d'une série de ressources spécifiques au projet (documentation, démonstration, anomalies, statistiques d'usage, etc.).

Lorsque vous cherchez à installer un module, choisissez de préférence les liens de téléchargement sur fond vert (stables) en accord avec la version de votre Drupal. Pour *Administration menu*, nous disposons de la version stable pour Drupal 6 : 6.x-1.3. En cliquant sur le lien *Download*, nous téléchargeons un fichier *tar.gz*, le même type d'archive que pour les sources de Drupal, que nous stockons pour l'instant sur le bureau.

BON À SAVOIR Recherche

des modules Drupal avec Google

Une astuce bien pratique pour rechercher un module particulier consiste à utiliser la syntaxe avancée de Google. En effet, les URL des pages projet des modules Drupal ont toujours la même structure, http://drupal.org/project/nom_projet. Pour trouver un projet qui traite du sujet « menu d'administration », il suffit donc de taper dans la zone de recherche de google :

```
site:drupal.org
inurl:project administration menu
```

Choisir vos modules

Avant de poursuivre, un avertissement : comme vous le voyez, la richesse des modules est un des grands atouts de Drupal. Mais elle présente aussi une certaine forme de danger consistant à les voir se multiplier dans votre système dans une joyeuse anarchie.

Lorsque vous décidez d'utiliser un module, surtout lorsque vous construisez un site professionnel, sachez évaluer sa qualité en vous posant les bonnes questions.

Ai-je vraiment besoin de ce module ?

L'utilisation des modules n'est pas sans conséquence sur les performances. PHP est fait pour que chaque module activé implique l'inclusion de fichiers de code supplémentaires, et ce, à chaque fois qu'une page est servie. Ce temps de chargement tend à se réduire au fur et à mesure des versions de Drupal mais reste néanmoins non négligeable.

BONNE PRATIQUE **Faites le ménage !**

Si vous ne cherchez qu'à tester un module, pensez à noter son nom sur une liste à part de sorte à ne pas oublier de le supprimer plus tard. Ne laissez jamais des modules inutilisés installés et encore moins activés.

Lorsque l'on active un module, il est donc important, a fortiori pour un site à forte fréquentation, de se poser la question de son utilité.

Dans le même esprit, il existe souvent plusieurs modules pour la même fonction, dont les champs d'action sont plus ou moins étendus. Par exemple, les modules Filebrowser et WebFM permettent tout deux d'afficher des listes de fichiers à la manière d'un service FTP. Filebrowser est cependant plus simple que WebFM qui lui est très complet. Si vous n'avez besoin que de permettre le téléchargement de fichiers, Filebrowser sera donc plus indiqué.

Enfin, la force modulaire de Drupal tient autant à la profusion des modules disponibles qu'au fait qu'il est très simple de réaliser ses propres modules, ou encore de simplifier des modules existants. Cette possibilité est totalement accessible à qui possède des compétences basiques en PHP¹.

Dans notre exemple, nous allons chercher à installer le module Administration menu dont le but sera de simplifier énormément l'administration d'un site Drupal. Le besoin est d'autant plus justifié qu'une fois activé, ce module sera chargé uniquement pour l'administrateur du site.

Ce module est-il stable ?

Une fois que votre choix s'est porté sur un module vient la question de sa stabilité. Tous les modules fournis pour Drupal sont l'œuvre de gens comme vous et moi qui ne passent pas 100 % de leur temps sur ce travail. Il convient donc de prendre cet aspect en compte pour n'avoir que les avantages de ces développements libres, sans les inconvénients.

Ne mettez ainsi jamais sur un site un module en développement (liens sur fond rouge) ou en version RC ou Alpha. Même si l'on peut transiger sur la seconde catégorie, ces modules ne sont, par définition, pas stables et ne doivent en aucun cas finir sur un serveur de production.

Les modules Drupal suivent un nommage de version uniformisé. Une version *6.x-2.0-dev* indique un module conçu pour Drupal 6, ayant pour version 2.0, et qui est en phase de développement.

Ce module survivra-t-il ?

Comme nous l'avons vu dans la description générale de Drupal, les modules ne bénéficient jamais d'une compatibilité ascendante. Un module écrit pour une version 6 de Drupal ne fonctionnera pas en l'état dans la version 7. Chaque changement de version implique pour les développeurs des modules un travail de « portage » de leur œuvre dans la version supérieure. Or, si le module est peu soutenu par une commu-

1. Un tutoriel pour fabriquer ses propres modules : <http://www.arnumeral.fr/node/14>

nauté d'utilisateurs, il sera peut-être abandonné par son auteur, vous obligeant alors à faire ou à faire faire ce portage s'il s'agit d'une fonction trop ancrée sur votre site pour être remplacée.

Pour évaluer au mieux l'avenir d'un module, il est donc important d'étudier l'écosystème qui gravite autour, à commencer par ses utilisateurs. Pour vous aider dans cette étude, deux outils se trouvent sur la page d'un module : les anomalies et les statistiques d'utilisation.

Les anomalies sont disponibles via le lien [View all issues](#) dans la page de projet d'un module. Elle vous permettent de voir le niveau d'activité sur le module. Si les dernières anomalies non corrigées datent de plus d'un an, c'est très mauvais signe.



Figure 5–24 Liste des anomalies remontées par les utilisateurs pour le module Administration menu

Dans l'exemple en copie d'écran, nous avons les anomalies du module Administration menu, et nous constatons que la dernière correction date de moins de deux jours, ce qui est très bon signe.

Les statistiques d'utilisation permettent quant à elles de se faire une idée du nombre de sites utilisant ce module à travers le monde. Pour y accéder, dans la page de projet du module, cliquez sur le lien [View usage statistics](#).

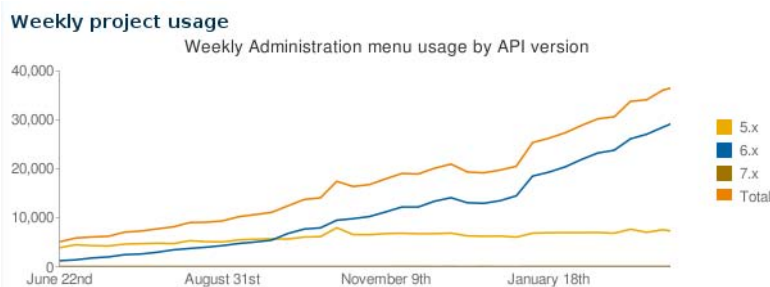


Figure 5–25
Statistiques d'utilisation
du module Administration menu

BON À SAVOIR La source d'Update status

Pour ceux qui se demanderaient d'où viennent ces chiffres, il s'agit tout simplement d'informations indirectement données par le module Update status que nous venons d'activer. En effet, le rôle de ce module est de vérifier que tous les modules activés sont à jour et ne présentent pas de failles de sécurité. Pour arriver à ses fins, il doit donc se connecter sur le site drupal.org pour comparer les numéros de version. Le site dispose ainsi de statistiques précises des modules activés.

BONNE PRATIQUE Une place pour chaque chose

Nous ne le répéterons jamais assez, le seul endroit où vous devez installer vos modules (mais aussi, nous le verrons, vos thèmes) est le dossier `sites`. N'installez jamais quoi que ce soit hors de ce dossier. Pour bien comprendre ce principe, vous devez pouvoir à tout moment décompresser une nouvelle version de Drupal et y déplacer le dossier `sites` pour que tout fonctionne à nouveau comme avant.

Ici, nous voyons que l'utilisation de la version pour Drupal 6 de ce module est en constante hausse ce qui, là aussi, est un bon signe.

Installer un module

L'installation de l'archive `tar.gz` d'un module est pour l'heure une opération manuelle assez similaire à l'installation de Drupal que nous avons effectuée. Il s'agira simplement de décompresser l'archive à un endroit précis de l'arborescence.

Comme nous l'avons vu dans la section « Comprendre la structure des dossiers » au chapitre précédent, les modules liés à l'installation de Drupal sont placés dans le dossier `modules`. Il ne faut donc en aucun cas ajouter à cet endroit vos propres modules. Cela fonctionnerait, mais vous poserait aussi beaucoup de problèmes au moment des mises à jour de votre installation quand vous devrez faire le tri entre vos modules et ceux de Drupal.

La bonne place pour mettre des modules est le dossier `sites`. Dans ce dossier, vous avez pour l'heure deux sous-dossiers `all` et `default`. C'est dans le sous-dossier `all` que devront être copiés tous les nouveaux modules que vous désirez ajouter à Drupal.

Quels que soient l'arborescence et le nombre de sous-dossiers que vous ajouterez dans ce dossier, Drupal retrouvera votre module. Ce n'est cependant pas une raison pour manquer d'organisation. La bonne pratique consiste à créer d'abord un sous-dossier nommé `modules` qui contiendra tous les modules que vous utiliserez. Peut-être réaliserez-vous dans l'avenir des modules personnels, qui cohabiteront avec d'autres venant de drupal.org. Pour éviter toute confusion, anticipons cette possibilité et créons encore deux sous-dossiers, `mes_modules` et `contrib`.

Il ne nous reste maintenant plus qu'à décompresser l'archive `admin_menu_6.x-1.3.tar.gz` dans le dossier `sites/all/modules/contrib`, donnant lieu à un nouveau sous-dossier `admin_menu`.

BON À SAVOIR wget pour une installation instantanée

Pour les *unixiens*, un moyen très simple d'installer un module Drupal consiste à utiliser la commande `wget`. Il suffit pour cela de copier l'adresse du lien de téléchargement dans la page de projet du module et de la recopier comme suit dans un terminal :

```
mkdir -p sites/all/modules/contrib
cd sites/all/modules/contrib
wget -qO - http://ftp.drupal.org/files/projects/admin_menu-6.x-1.3.tar.gz | tar -zxvf -
```

La commande `wget` lit l'archive à travers l'Internet et en restitue le contenu dans la sortie standard. La commande `tar` lit cette sortie et effectue la décompression à la volée. Le résultat sera un module installé directement sans fichiers temporaires.

La mise en place de notre module étant faite, retournons à présent dans l'administration des modules pour y trouver Administration menu. Nous remarquons au passage l'apparition d'une nouvelle catégorie *Administration*. Il ne reste alors qu'à cocher son activation, sans oublier de cliquer ensuite sur *Enregistrer la configuration*.



Figure 5–26
Activation du module Administration menu dans le panneau d'administration des modules

Une fois le module activé, vous devriez voir apparaître en haut de la page un nouveau menu déroulant regroupant l'ensemble de l'administration de Drupal. Où que vous soyez, vous pourrez donc utiliser ce menu pour paramétrer votre site.

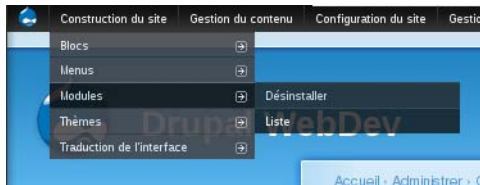


Figure 5–27
Le module Administration menu en action

BON À SAVOIR Réorganiser vos modules

Si par hasard vous avez besoin de réorganiser vos modules dans différents dossiers, il se peut que Drupal n'apprécie pas cette action et que vous aboutissiez sur une page blanche. Pas de panique, la procédure de mise à jour que nous aborderons un peu plus loin est capable de régler le problème. Il suffira de la lancer jusqu'au bout pour que votre site soit à nouveau fonctionnel.

Désinstaller un module

Lorsqu'un module est décompressé à partir de son archive puis activé, il va potentiellement créer des tables dans la base de données, des variables dans le système Drupal, etc. Il est donc hors de question pour le désinstaller de simplement détruire le dossier d'un module.

Vous devez avant toute chose retourner dans l'administration des modules pour désactiver le module que vous désirez supprimer. Ceci fait, en haut de la page, vous trouverez un onglet *Désinstaller*. Il faut comprendre ici « Désinstaller les données installées par le module ». Cette option ne détruira en aucun cas des fichiers.

Sur cette nouvelle page, vous retrouverez la liste des modules qui ont été désactivés et en attente d'une possible désinstallation. Cochez les modules que vous souhaitez supprimer et cliquez sur *Désinstaller*.



Figure 5–28
Liste des modules susceptibles
d’être désinstallés

Drupal vous renvoie alors sur une page de confirmation où vous pouvez encore [Annuler](#) ou confirmer l’action de [Désinstaller](#). Ceci fait, vous pourrez supprimer physiquement le dossier du module.

Soumettre une anomalie

Comme Drupal lui-même, les modules sont des logiciels libres développés par des gens comme vous et moi. Chacun peut participer à l’amélioration de l’ensemble et ainsi faire avancer la communauté. Il peut s’agir de code bien sûr, mais aussi de traduction ou encore plus simplement de prendre la peine de signaler une erreur de fonctionnement à l’auteur d’un module. Vous serez alors surpris de la vitesse de réaction de certains auteurs qui ne mettront que quelques jours à corriger une anomalie.

Pour déclarer une anomalie, il vous faudra tout d’abord créer un compte sur [Drupal.org](#). Cela se fait tout simplement en cliquant sur le lien [Create new account](#) dans la boîte *User I* du site.



Figure 5–29
Créer un compte sur drupal.org

Une fois vos identifiants reçus, il suffit d’aller sur la page de projet du module et de cliquer sur [View all issues](#) pour déclarer l’anomalie. Si votre anomalie n’a pas encore été trouvée (ce que vous pouvez vérifier en faisant des recherches), cliquez sur le lien [Create a new issue](#) et décrivez aussi précisément que possible le problème rencontré sans oublier de

spécifier la version du module, le type d'anomalie (critique, mineur, etc.) et de laisser le statut à *active*. Plus vous serez précis dans la description, plus l'auteur pourra résoudre rapidement votre problème.

Project information

Project: Administration menu **Version: *** 6.x-1.3 **Component: *** Code

Issue information

Category: * bug report **Priority:** minor **Assigned:** Unassigned **Status:** active

Descriptions of the **Priority** and **Status** values can be found in the [Issue queue handbook](#).

Issue details

Title: *

Description: *

Figure 5–30 Saisir une nouvelle anomalie

Une fois validée, votre anomalie apparaîtra dans la liste *View all issues*. Vous pourrez alors revenir plus tard pour prendre connaissance des réponses données.

Le paramétrage général de Drupal

Comme nous l'avons vu, chaque module a son rôle à jouer dans l'administration de Drupal. Il est donc grand temps de faire un tour dans la configuration du site pour régler toutes les fonctionnalités qui sont maintenant activées.

Comme nous l'avons vu au début de ce chapitre, tout ce qui touche à la configuration de Drupal se trouve dans la section *Configuration du site*. Cliquons sur ce lien pour parcourir ensemble les principaux réglages qui nous sont offerts.

Boîte à outils image

La boîte à outils GD2 est installée et fonctionne correctement.

Qualité JPEG :

75 %

Définir la qualité d'image pour les conversions en JPEG, sur une échelle de 0 à 100.

Figure 5–31

Réglage de la qualité des images JPEG

Dans cette section, vous pouvez régler la qualité de compression des images qui sera appliquée par le processeur d'images de Drupal. Ce der-

nier peut utiliser la bibliothèque `php-GD2` (conseillé) ou `php-imagick`. Un message vous indique au début de la page quel système est activé ou vous avertit le cas échéant qu'aucun système n'a été installé.

Lorsque vous avez terminé le paramétrage, vous pouvez cliquer sur le bouton *Enregistrer la configuration*.

Date et Heure

Cette section s'occupe de tout ce qui touche à la gestion du temps dans Drupal. Cela commence par le choix du fuseau horaire. Pour régler ce champ, faites correspondre l'une des heures données dans la liste déroulante avec l'heure réelle.

Langues et paramètres régionaux

Fuseau horaire par défaut :
 mercredi, 8 juillet 2009 - 1:35am +0200
 Choisissez le fuseau horaire par défaut du site.

Fuseaux horaires personnalisables :
☐ Désactivé
☒ Activé
 Si activé, les utilisateurs peuvent définir leur propre fuseaux horaires

Premier jour de la semaine :
 lundi
 Le premier jour de la semaine, pour les vues de type calendrier.

Figure 5-32
Réglage du fuseau horaire

Vous pouvez en outre activer les fuseaux personnalisables qui permettront à vos utilisateurs de choisir leur propre fuseau horaire à travers la configuration de leur profil.

Enfin, vous devez définir dans la liste déroulante suivante le premier jour de la semaine qui servira pour les vues de type « calendrier ». Il s'agit généralement de sélectionner « lundi ».

Formats

Format court des dates :
 08/07/2009 - 01:37
 Le format court d'affichage des dates et heures.

Format moyen des dates :
 8 juillet 2009 - 1:37am
 Le format moyen d'affichage des dates et heures.

Format long des dates :
 mercredi, 8 juillet 2009 - 1:37am
 Le format d'affichage des dates et heures le plus détaillé.

Figure 5-33
Choix des formats pour les dates

Ensuite vient le format d’affichage des dates selon les besoins : court (date numérique et heure), moyen (jour en clair, date numérique et heure) et long (jour et mois en clair, année et heure). Nous allons choisir un format francisé avec le jour qui précède le mois.

Lorsque vous avez terminé le paramétrage, vous pouvez appuyer sur le bouton *Enregistrer la configuration*.

Informations

Informations

Nom : *

 Le nom de ce site.

Adresse électronique : *

 Le champ *De* des emails automatiques envoyés pendant la procédure d’enregistrement, les demandes les autres notifications.

Slogan :

 La devise de votre site, slogan, ou une phrase accrocheuse (souvent affichée à côté du titre du site).

Figure 5–34

Nous retrouvons ici les informations saisies lors de l’installation.

Dans cette section, vous retrouverez une partie des éléments que nous avons définis lors de l’installation :

- *Nom* du site : normalement vous devez y retrouver « Colchique » ;
- *Slogan* du site : une devise ou une accroche qui pourra apparaître au côté du titre du site dans l’en-tête. Pour l’exemple, saisissez ici une devise comme « Car chaque brin d’herbe est unique... » ;
- *Objectifs* du site : qui apparaîtront en haut de la page d’accueil. Il est possible d’utiliser ici la syntaxe XHTML. Laissons ce champ vide pour nos tests ;
- *Pied de page* du site : qui apparaîtra à la fin de chacune des pages. Là aussi, la syntaxe XHTML est autorisée. Nous allons utiliser un meilleur moyen pour gérer les pieds de page, laissons donc ce champ vide lui aussi ;
- *Nom de l'utilisateur anonyme* : cet identifiant sera le nom officiel des utilisateurs qui ne sont pas connus du système. Il sera utilisé lorsqu’un visiteur commentera un billet sans donner son nom ;
- *Page de garde par défaut* : Ce champ est très important car il détermine ce que la page d’accueil contiendra. La valeur *node* indique à Drupal que la page d’accueil présentera une liste de contenus (appelés des nœuds, ou *nodes*, en langage Drupalien) classés du plus récent au plus ancien. Vous pouvez cependant changer cette valeur pour

n'importe quelle référence (URL) à un contenu que vous aurez créé, comme nous le verrons un peu plus loin.

Lorsque vous aurez terminé le paramétrage, vous pourrez cliquer sur le bouton *Enregistrer la configuration*.

Journalisation et alertes

Par défaut, Drupal stocke dans la base de données toutes les erreurs détectées par les modules ainsi qu'un grand nombre d'informations liées à la vie du site (ajout de commentaires, de contenus, authentification d'utilisateurs, etc.). Ces messages sont disponibles dans la section *Rapports* de l'administration, par le lien *Entrées récentes du journal*.

Figure 5-35
Réglage du nombre d'enregistrements
dans le journal

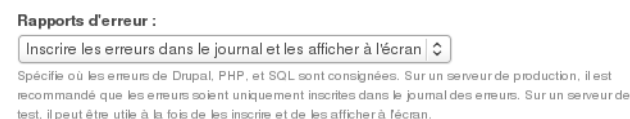


Comme ces données peuvent à terme prendre un volume très important dans la base, il est important d'en limiter le nombre. Pour régler cette valeur, cliquez sur le lien *Database logging*. La liste déroulante qui apparaît indique que le nombre de messages conservés est de *1 000*. La vidange des données supplémentaires est effectué par le cron de Drupal que nous allons étudier un peu plus loin.

Rapports d'erreur

Laissons pour le moment les deux premiers champs que nous utiliserons plus loin dans notre projet pour nous intéresser à la liste déroulante *Rapports d'erreur*. Cette liste va nous permettre de déterminer si Drupal doit afficher les erreurs PHP à l'écran ou seulement les consigner dans le journal des erreurs.

Figure 5-36
Paramétrage des rapports d'erreur
pour une plate-forme de développement



Le premier cas est surtout utile pendant la phase de développement de notre site où nous apprécierons d'avoir un retour visuel des erreurs que nous aurions commises. En revanche, il est nécessaire de positionner cette valeur dans le mode *Journal d'erreur seulement* si vous paramétrez votre site de production.

Une fois le paramétrage terminé, n'oubliez pas d'appuyer sur le bouton *Enregistrer la configuration*.

Transfert de fichiers

Le transfert de fichiers est géré par le module Upload que nous avons activé. Il est nécessaire à l'attachement de fichiers aux contenus.

Nous le verrons plus loin dans ce livre, les documents attachés à un contenu peuvent être affichés sous la forme d'une liste ou être invisibles. Le champ *Liste les fichiers par défaut* nous permet de déterminer lequel des deux modes est activé par défaut. Cela n'empêchera pas l'utilisateur de changer cela au cas par cas.

Ensuite, nous allons pouvoir imposer un certain nombre de limites aux fichiers qui pourront être envoyés sur le serveur. S'il s'agit d'une image, nous pouvons définir une dimension maximale. Si cette dimension est dépassée, l'image sera automatiquement redimensionnée. Par défaut, la dimension est de 0 et signifie qu'aucune limite n'est appliquée.

Résolution maximale des images transférées :

Largeur x Hauteur

La taille maximale autorisée pour les images transmises (par exemple 640x480) ou "0" pour indiquer aucune restriction. Si une *boîte à outil image* est installée, les fichiers qui dépassent cette taille seront redimensionnés.

Figure 5–37

Toute image de plus de 1280 × 1024 pixels sera automatiquement redimensionnée.

Nous pouvons aussi imposer une contrainte de taille pour chaque fichier (par défaut, 1 mégaoctet) ou pour un utilisateur (par défaut, 1 mégaoctet ici aussi). Tout dépassement entraînera le refus du téléchargement. Attention cependant, pour la taille maximale d'un fichier de ne pas dépasser la taille maximale autorisée par votre configuration de PHP (ici 8 mégaoctets). Si vous avez besoin de plus, changez cette valeur dans votre fichier *php.ini* et redémarrez ensuite le service Apache.

(Par défaut) Taille maximale de chaque fichier :

Mo

La taille maximale que peut avoir un fichier transféré par un utilisateur. Si une image est transférée et qu'une résolution maximale est positionnée, la taille sera contrôlée après que le fichier ait été redimensionné.

(Par défaut) Taille totale des fichiers par utilisateur :

Mo

(Par défaut) La taille maximale de tous les fichiers qu'un utilisateur peut placer sur le site.

Vos paramètres de PHP limitent la taille maximale de chaque transfert à 8 Mo.

Figure 5–38

Augmentation à 8 et 10 Mo des tailles limites

Il nous est possible de définir une liste d'extensions autorisées nous permettant d'éviter des formats de fichiers qui ne nous conviendraient pas.

Figure 5–39
Liste des extensions autorisées



BONNE PRATIQUE Les formats ouverts

Lorsque vous attachez un document à un contenu, pensez toujours à l'utilisateur final et si ce dernier a la capacité ou non de le lire. À ce titre, évitez autant que possible les formats propriétaires qui ne seront lisibles que des personnes disposant des licences adéquates. Pour des documents non modifiables, vous pouvez utiliser facilement le format PDF, lisible par tous et sur toutes les plates-formes. Pour des documents bureautiques, le format ouvert OpenDocument est pris en charge par de nombreux logiciels gratuits comme OpenOffice.org (<http://www.openoffice.org>).

Système de fichiers

Drupal a besoin de deux emplacements pour stocker ses fichiers. Le premier emplacement correspond aux fichiers qui sont attachés à des contenus par le module Upload, le second à des fichiers temporaires qui n'ont pas vocation à être conservés.

Figure 5–40
Dossier temporaire utilisé par Drupal



Par défaut, Drupal paramètre le dossier temporaire de manière à utiliser le dossier temporaire de votre système d'exploitation, ici `/var/tmp` (ou `/tmp`) pour un système Unix ; nous pouvons donc le laisser ainsi sans trop de problèmes.

Figure 5–41
Dossier de stockage des fichiers attachés



BON À SAVOIR Déplacer le dossier des fichiers attachés

Les fichiers attachés sont référencés dans la table `Files` de la base de données. Si vous êtes dans la nécessité de déplacer le dossiers des fichiers, vous avez toujours la possibilité de corriger les chemins contenus dans la colonne `filepath` de cette table.

S'agissant des fichiers attachés aux contenus, il est important que vous choisissiez un dossier que vous sauvegarderez régulièrement. La norme Drupal veut que ces fichiers soient stockés par défaut dans le dossier `sites/default/files` sous sa racine, mais rien ne vous empêche de les placer ailleurs. Gardez cependant à l'esprit qu'à chaque fois qu'un fichier est attaché à un contenu, Drupal stocke en base de données le chemin exact de ce dernier. Si vous changez ce dossier en cours de route, l'ensemble des fichiers précédemment attachés deviendront inaccessibles.

Dans cette page, il est aussi possible de régler la *Méthode de téléchargement* des fichiers. Le choix entre l'option *Privée* et l'option *Publique* dépend de deux critères :

- dans le mode public, c'est le service Apache qui va être chargé d'envoyer les fichiers au navigateur. Le dossier doit donc dans ce cas être sous la racine de Drupal, ce qui est le cas par défaut.
- dans le mode privé, c'est Drupal qui envoie les fichiers au navigateur. Le dossier peut donc être n'importe où sur le disque.

De plus, en mode privé, Drupal gère les autorisations de téléchargement, alors qu'en mode public, Apache autorise par défaut tous les fichiers à être téléchargés. C'est d'ailleurs ce dernier point qui vous permet de choisir l'un des deux modes. Si vous souhaitez attacher à vos contenus des fichiers confidentiels, vous devez choisir le mode *Privé*. Au contraire, si vous n'avez que des fichiers qui sont de toute façon accessibles à tous, prenez l'option *Publique*, qui présente l'avantage de consommer moins de ressources.


Dans un cas comme dans l'autre, il faut que le service Apache ait les droits d'écriture sur ce dossier.

En résumé

- L'administration de Drupal est certes complexe, mais elle est rationnelle. En peu de temps, vous acquerrez les bons réflexes qui vous permettront de vous y repérer.
- L'ensemble des fonctions fondamentales de Drupal est issu des modules fournis en standard.
- Il est possible d'étendre les fonctionnalités de Drupal à l'aide de nouveaux modules que vous pouvez télécharger sur drupal.org.
- Drupal est un outil communautaire qui a besoin de votre aide. Lorsque vous trouvez une erreur dans le cœur de Drupal ou dans l'un de ses modules, n'hésitez jamais à signaler l'anomalie.
- Si vous butez sur un problème concernant un module, ou si vous n'arrivez pas à identifier le module correspondant à votre besoin, vous pouvez poser une question sur le forum <http://drupalfr.org/forum>, dans la section *Support général*.

6

chapitre



Colchique

Accueil • Administrer • Gestion des utilisateurs

administrateur

- Mon compte
- Créer un contenu
- Administrer
 - Construction du site
 - Gestion du contenu
 - Configuration du site
 - Gestion des utilisateurs
 - Droits d'accès**
 - Paramètres des utilisateurs
 - Règles d'accès
 - Rôles
 - Utilisateurs
 - Rapports
 - Aide
- Se déconnecter

Droits d'accès

Les droits d'accès vous permettent de contrôler ce que les utilisateurs font sur votre site. Chaque rôle d'utilisateur (défini sur la [page des rôles d'utilisateur](#)) a ses propres droits d'accès. Par exemple, vous pourriez donner aux utilisateurs classés comme "Administrateurs" le droit "administrer les noeuds" mais refuser ce pouvoir aux utilisateurs ordinaires, "authentifiés". Vous pouvez utiliser les droits d'accès pour révéler de nouvelles fonctions à des utilisateurs privilégiés (les abonnés par exemple). Les droits d'accès permettent aussi à des utilisateurs de confiance de partager la charge de l'administration d'un site très actif.

Droit d'accès	utilisateur anonyme	utilisateur identifié	contributeur
module block			
administrer les blocs	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
utiliser PHP pour la visibilité des blocs	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
module comment			
accéder aux commentaires	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
administrer les commentaires	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
soumettre des commentaires	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
soumettre des commentaires sans approbation	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Gérer les utilisateurs

Les utilisateurs sont un élément fondamental de tout CMS. Ce sont eux qui produisent des contenus, seuls ou à plusieurs ; ce sont aussi eux qui commentent les contenus et les font vivre. Il est alors logique sous Drupal d'accorder aux utilisateurs un statut spécifique, associé à un mode de gestion particulièrement poussé couvrant de multiples dimensions.

SOMMAIRE

- ▶ Comprendre le système de droits de Drupal
- ▶ Paramétrer les comptes utilisateur
- ▶ Gérer les nouveaux utilisateurs

Rôles, permissions et utilisateurs

Le système de droits de Drupal est fondé sur une combinaison permissions/rôles et rôles/utilisateurs.

- Une permission est un droit simple portant sur une opération de Drupal, par exemple « Créer un nouvel article ».
- Un rôle est quant à lui un ensemble donné de permissions.
- Un utilisateur est la manière dont Drupal voit une personne qui visite le site. À cet utilisateur est associé un ou plusieurs rôles.

Par défaut, Drupal définit trois rôles fondamentaux :

- le rôle Administrateur, qui est associé à un seul utilisateur qu'on appelle donc l'administrateur ;
- le rôle Utilisateur identifié, qui est associé automatiquement à tout utilisateur qui est connu de Drupal, c'est-à-dire qui dispose sur le site d'un compte utilisateur et qui a en conséquence saisi un identifiant et un mot de passe ;
- le rôle Utilisateur anonyme, qui est associé automatiquement à un visiteur. À noter que si une personne possédant un compte sur le site ne saisit pas son identifiant et son mot de passe, Drupal le considère comme Utilisateur anonyme.

Venant seulement d'installer notre Drupal, un seul utilisateur est pour le moment connu du système : c'est l'administrateur.

L'administrateur est l'utilisateur ayant le plus de pouvoir sur le système Drupal. Un peu à la manière de l'utilisateur root sur un système Unix, rien n'est interdit à l'administrateur. Il n'est en conséquence pas très avisé de travailler en permanence avec ce rôle. Parce qu'il possède les droits les plus étendus, vous risquez de biaiser votre développement en croyant que certaines actions sont possibles, ou certaines pages accessibles, alors qu'elle ne le sont pas pour un simple visiteur.

C'est donc une bonne idée pour la qualité de notre projet que de devenir schizophrène en créant trois identités distinctes que l'on endossera tour à tour, et selon les tâches à effectuer :

- administrateur : en saisissant l'identifiant et le mot de passe administrateur que nous avons choisi lors de l'installation ;
- visiteur : pour cela, il nous suffira de nous déconnecter ;
- Marine : une utilisatrice que nous allons créer qui disposera automatiquement du rôle Utilisateur identifié mais aussi d'un nouveau rôle Contributeur que nous allons aussi créer. Ce rôle sera doté de toutes les permissions lui permettant de créer des contenus.

CULTURE

L'origine du concept rôle/permission

Le système de droit de Drupal est issu d'un modèle datant de 2003 appelé OrBAC (*Organization Based Access Control*, en français Contrôle d'accès basé sur l'organisation) qui introduit un sous-modèle appelé R-BAC (*Role-Based Access Control*, en français Contrôle d'accès à base de rôles) où les utilisateurs exerçant des fonctions similaires sont regroupés sous un même rôle auquel est associé, par une autorité supérieure, une série d'autorisations d'accès (nos permissions Drupal).

Dans la suite de ce livre, nous basculerons donc tour à tour sur l'un de ces rôles selon que nous travaillerons sur la structure du site (utilisateur administrateur), sur le contenu du site (utilisateur contributeur) ou sur l'utilisation du site (utilisateur visiteur).

Créer vos propres rôles

La première phase de la gestion des droits consiste donc à définir les rôles qui nous seront nécessaires pour notre site. Dans cet ouvrage, nous nous limiterons aux trois types d'utilisateurs évoqués plus haut : administrateur, contributeur et visiteur.

Le rôle d'utilisateur identifié correspond à un rôle pris par n'importe quel utilisateur s'étant préalablement identifié sur notre système – c'est-à-dire ayant saisi un identifiant et un mot de passe – à la différence de l'utilisateur anonyme. Un utilisateur ayant le rôle Contributeur ou Administrateur possède donc de manière implicite le rôle Utilisateur identifié.

Pour créer notre rôle Contributeur, nous devons aller dans le menu *Administrateur*, puis cliquer sur *Administrer*, *Gestion des utilisateurs*, puis enfin sur *Rôles*.

Nom	Opérations
utilisateur anonyme	verrouillé modifier les droits d'accès
utilisateur identifié	verrouillé modifier les droits d'accès
<input type="text" value="contributeur"/>	<input type="button" value="Ajouter un rôle"/>

Figure 6–1

Ajout d'un nouveau rôle Contributeur

Nous retrouvons ici installés par défaut deux des rôles que nous venons d'évoquer : utilisateur anonyme et identifié. Le rôle Administrateur est quant à lui absent car, possédant par définition tous les droits, il n'est pas nécessaire de le faire apparaître.

Nous allons donc saisir le nom de notre nouveau rôle, *contributeur*, et appuyer sur le bouton *Ajouter un rôle*.

Gérer les permissions par rôle

Disposant maintenant des rôles Utilisateur anonyme et Contributeur, il nous faut définir les permissions qui leur sont associées. Pour ce faire, vous pouvez soit cliquer simplement sur le lien *modifier les droits d'accès* à côté du rôle correspondant, soit retourner dans le menu *Administrateur*,

puis *Gestion des utilisateurs*, et enfin *Droits d'accès*. La seule différence entre ces deux approches est que la seconde va nous permettre de modifier les droits pour l'ensemble des rôles sur une seule et même page.

Droit d'accès	utilisateur anonyme	utilisateur identifié	contributeur
module block			
administrer les blocs	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
utiliser PHP pour la visibilité des blocs	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
module comment			
accéder aux commentaires	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
administrer les commentaires	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
soumettre des commentaires	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
soumettre des commentaires sans approbation	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Figure 6–2 Gestion des permissions

BON À SAVOIR Comment créer très simplement un site privé

Si vous cherchez à créer un site accessible uniquement aux utilisateurs identifiés, il faut absolument qu'aucune case ne soit cochée dans la colonne de l'utilisateur anonyme. De cette manière, la seule chose que verra l'utilisateur anonyme en arrivant sur votre site sera un message d'erreur « Vous n'êtes pas autorisé à accéder à cette page » et une boîte de saisie pour un nom d'utilisateur et un mot de passe. Si le visiteur dispose d'un compte, il devra se connecter pour voir le contenu, sinon, ce contenu restera privé. Cela représente une sécurité raisonnable.

Chaque ligne de cette nouvelle page correspond à une permission, un droit d'accès. Comme vous le voyez, chaque module propose ses propres permissions, ce qui implique une granularité des autorisations rarement égalée dans le monde des CMS.

Le rôle Contributeur aura pour caractéristique de pouvoir, comme le rôle Administrateur, créer un article. Les articles et les contenus sont pris en charge par le module Node. En se déplaçant jusqu'à cette section, nous découvrons qu'il est possible de régler à peu près tous les aspects de la création de contenu. Nous concernant, nous allons cocher *accéder au contenu* pour tous les rôles car tout le monde aura le droit de voir les articles, y compris les anonymes. En revanche, nous cochons *create story content* (créer un article) pour le seul rôle Contributeur.

Voyons maintenant plus précisément, module par module, quelles permissions nous devons affecter à tel ou tel rôle.

Permission du module Comment

Ce module est dédié à la gestion des commentaires. Nous avons besoin d'affecter les droits suivants.

- *Accéder aux commentaires* : pour permettre à tous les utilisateurs identifiés ou non de voir les commentaires.
- *Soumettre des commentaires* et *Soumettre des commentaires sans approbation* : pour permettre à tous les utilisateurs identifiés ou non d'écrire des commentaires. Le second droit, quant à lui, permet à l'utilisateur de publier directement ces commentaires, sans être modéré.

Droit d'accès	utilisateur anonyme	utilisateur identifié	contributeur
module comment			
accéder aux commentaires	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
administrer les commentaires	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
soumettre des commentaires	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
soumettre des commentaires sans approbation	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Figure 6–3
Les permissions à activer pour le module Comment

Notez que nous ne donnons ces droits qu’au rôle Utilisateur identifié et pas au rôle Contributeur. En effet, un utilisateur ayant le rôle Contributeur aura aussi automatiquement le rôle Utilisateur identifié. Comme nous cherchons à ce que tous les utilisateurs puissent déposer des commentaires, cocher cette permission pour Contributeur n’apporterait rien de plus.

Permissions du module Node

Le module Node est, comme nous venons de le voir, chargé de la gestion des contenus. Dans cette section, nous allons donner les droits suivants.

- *Accéder au contenu* : comme pour les commentaires, tous les utilisateurs, identifiés ou non, peuvent lire les contenus.

Droit d'accès	utilisateur anonyme	utilisateur identifié	contributeur
accéder au contenu	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Figure 6–4
La permission Accéder au contenu

- *Create story content* : nous le verrons plus loin, un contenu de type « story » est un article. Seuls les contributeurs disposent du droit de créer des articles. La présence de termes en anglais tient à ce que la traduction française de Drupal repose sur le travail de bénévoles et est encore imparfaite. Référez-vous au chapitre « Gérer les langues » pour plus d’informations.

Droit d'accès	utilisateur anonyme	utilisateur identifié	contributeur
create story content	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
delete any page content	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
delete any story content	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
delete own page content	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Figure 6–5
Droit de création et de suppression d’article

Figure 6–6
Droit d’édition d’un article
et de visualisation des révisions

Droit d'accès	utilisateur anonyme	utilisateur identifié	contributeur
edit own story content	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
revenir sur les révisions	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
voir les révisions	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

- *Delete own story content* : ce droit permet aux contributeurs de supprimer leurs propres articles.
- *Edit own story content* : comme le droit précédent, celui-ci permet aux contributeurs de modifier leurs propres articles.
- *Voir les révisions* : pour permettre au contributeur de visualiser l’historique des contenus.

Permissions du module Contact

Accéder au formulaire de contact global : nous voulons que tout le monde puisse nous contacter, nous cochons donc aussi bien les utilisateurs anonymes que ceux identifiés.

Figure 6–7
Les permissions à activer
pour le module Contact

Droit d'accès	utilisateur anonyme	utilisateur identifié	contributeur
module contact			
accéder au formulaire de contact global	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Permissions du module Path

Le module Path permet de définir des alias entre le chemin natif d’un contenu (ex. <http://www.colchique-conseil.fr/node/12>) et une version plus lisible de l’URL (ex. <http://www.colchique-conseil.fr/mon-contenu>). Nous ne voulons donc donner ce droit qu’aux seuls contributeurs. Vous trouvez juste en-dessous la permission *gérer les alias d’URL*. Il s’agit là d’une action d’administration consistant à permettre à un rôle de modifier les alias créés, dans leur ensemble. Nous ne donnons donc pas ce droit au contributeur.

Figure 6–8
Les permissions à activer pour le module Path

Droit d'accès	utilisateur anonyme	utilisateur identifié	contributeur
module path			
gérer les alias d'url	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
créer des alias d'url	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Permissions du module Search

Le module Search permet de rechercher dans les contenus en mode simple (une peu comme la recherche basique de Google) ou en mode avancé (avec des critères supplémentaires).

Généralement, nous donnons le droit de recherche simple à tout le monde, et la recherche avancée aux seuls utilisateurs authentifiés. Cela s'explique principalement par le fait que la recherche avancée demande plus de ressources et qu'on ne veut pas augmenter exagérément la charge machine en la rendant trop facilement accessible.

Droit d'accès	utilisateur anonyme	utilisateur identifié	contributeur
module search			
administrer la recherche	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
rechercher dans le contenu	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
utiliser la recherche avancée	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Figure 6–9
Les permissions à activer pour le module Search

Permissions du module Upload

Le module Upload va nous permettre d'attacher des fichiers binaires (PDF, images, etc.) à nos contenus. Il convient donc de donner le droit de transfert aux seuls contributeurs, mais en revanche, de permettre la visualisation des fichiers attachés à tous. Notez cependant qu'en donnant le droit *voir les fichiers transférés* à l'utilisateur anonyme, ces fichiers seront aussi lus par tous les automates de recherche comme Google. À vous de décider si cela vous pose un problème.

Droit d'accès	utilisateur anonyme	utilisateur identifié	contributeur
module upload			
transférer des fichiers	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
voir les fichiers transférés	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Figure 6–10
Les permissions à activer pour le module Upload

BON À SAVOIR

Limitations spécifiques à un rôle

Lorsque vous donnez le droit *transférer des fichiers* à un rôle, une zone dépliable dédiée à ce rôle apparaîtra dans le panneau de configuration du module Upload (*Configuration du site*, puis *Transfert de fichiers*). Dans cette zone, il sera possible de définir des limitations en taille et en résolution spécifiques à ce rôle. À chaque fois que vous ajoutez un rôle, vérifiez que ces valeurs sont bien celles auxquelles vous vous attendez.

Permissions du module User

Le module User va gérer tous les aspects de la vie d'un utilisateur et plus particulièrement son compte (ou profil).

- *Accéder aux profils des membres* : en activant cette permission pour les utilisateurs anonymes et identifiés, nous les autorisons, en cliquant sur le nom d'un utilisateur, à accéder à leur profil et aux informations (non confidentielles évidemment) qu'il contient.

Figure 6–11
Les permissions à activer pour le module User

BON À SAVOIR Disposer d'un rôle Administrateur

Comme cela a été dit, il n'y a qu'un seul administrateur sur un système Drupal, ce qui peut poser problème lorsque nous voulons déléguer l'administration à un utilisateur tiers. Il n'est en effet pas concevable de lui donner l'identifiant et le mot de passe administrateur.

La solution consiste à créer un nouveau rôle Administrateur et de lui affecter toutes les permissions. En procédant de la sorte, il suffira par la suite d'affecter ce rôle à l'administrateur délégué pour qu'il puisse administrer le système.

L'énorme avantage de cette méthode est que ce rôle peut être ajouté ou retiré très facilement. À noter qu'il est aussi possible de créer des versions allégées en ne cochant qu'une partie des droits.

Figure 6–12
La boîte d'authentification

Droit d'accès	utilisateur anonyme	utilisateur identifié	contributeur
module user			
accéder aux profils des membres	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
administrer les droits d'accès	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
administrer les utilisateurs	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
changer son nom d'utilisateur	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

- *Changer son nom d'utilisateur* : ce droit est à votre discrétion, il permet à un utilisateur de changer son nom d'utilisateur dans son menu *mon profil*.

Une fois toutes les cases correctement cochées, il suffit d'appuyer sur le bouton *Enregistrer les droits* d'accès pour que les rôles deviennent effectifs.

Paramétrage des comptes utilisateur

Qu'est-ce qu'un compte utilisateur ?

Lorsqu'un visiteur arrive sur notre site, rien ne le distingue d'un autre visiteur. Il est alors désigné comme « anonyme ». Pour passer du statut d'anonyme à celui d'utilisateur identifié, il devra saisir un nom d'utilisateur et un mot de passe. C'est dans ce but qu'une boîte d'identification est affichée sur chacune des pages du site.

Connexion utilisateur

Nom d'utilisateur : *

Mot de passe : *

Se connecter

Créer un nouveau compte

Demander un nouveau mot de passe

Mais pour s'identifier, encore faut-il que l'utilisateur dispose d'un compte sur notre site. Nous allons donc maintenant voir comment Drupal permet d'ajouter de tels comptes et comment leur associer les rôles que nous venons de créer.

Choix du mode de création d'un compte utilisateur

Drupal propose de nombreuses options liées à la création d'un compte utilisateur qu'il convient de maîtriser pour adapter son fonctionnement à vos besoins. L'ensemble de ces options est concentré sur la page accessible par *Gestion des utilisateurs* puis *Paramètres des utilisateurs*.

Drupal offre trois manières distinctes de créer un compte utilisateur, allant de la plus restrictive à la plus laxiste. Pour choisir l'un de ces trois modes, remontons au début de la page *Paramètres des utilisateurs*.

Mode Administrateur seul

☐ Seuls les administrateurs du site peuvent créer de nouveaux comptes.

Figure 6-13

Création d'un compte par le seul administrateur

Il s'agit là du premier et du plus simple de ces trois modes qui demande à l'administrateur de prendre en charge l'entière procédure de création. En réalité, même si vous choisissez l'une des deux autres options, l'administrateur gardera toujours le privilège de création d'un compte. La seule différence est qu'avec ce mode il sera le seul et unique utilisateur habilité à le faire.

Comme nous le verrons un peu plus loin, l'administrateur, après avoir créé ce compte, pourra demander à ce qu'un courriel soit envoyé au nouvel utilisateur pour le prévenir que son accès est prêt. Vous pouvez modifier ce courriel à partir du modèle se trouvant dans *Bienvenue, nouvel utilisateur créé par l'administrateur*.

Mode Sans approbation

☐ Les visiteurs peuvent créer leur propre compte sans approbation de l'administrateur.

Figure 6-14

Création d'un compte Sans approbation

Dans ce second mode, n'importe quel visiteur aura la possibilité de créer son propre compte. Les étapes du processus de création sont les suivantes.

- 1 L'utilisateur anonyme doit cliquer sur le lien *Créer un nouveau compte* présent dans la boîte *Connexion utilisateur*. Notez que si ce mode n'est pas sélectionné, ce lien n'apparaîtra pas.
- 2 Un formulaire s'affiche où l'utilisateur anonyme doit choisir son identifiant, son adresse courriel, puis valider.
- 3 Si vous avez coché la case *Exiger la vérification de l'adresse e-mail quand un visiteur crée un compte*, un message apparaît indiquant qu'un courriel de confirmation lui a été envoyé. Le modèle de ce courriel peut lui aussi être modifié dans la sous-section *Bienvenue, pas d'approbation*

Figure 6–15
Création d'un compte Avec approbation

requis. Le cas échéant, si la case n'est pas cochée, le processus passe directement à l'étape 6.

- 4 Lorsque l'utilisateur reçoit le courriel de confirmation, il doit cliquer sur le lien qui lui est donné ; il aboutit alors sur une page lui demandant de choisir cette fois son mot de passe.
- 5 Lorsqu'il a terminé, l'utilisateur clique sur le bouton *Enregistrer*.
- 6 La création du compte est terminée ; le compte est maintenant actif, et l'utilisateur reçoit un courriel de confirmation dont vous pouvez modifier le modèle dans la sous-section *E-mail d'activation du compte*.

Mode Avec approbation

● Les visiteurs peuvent créer leur propre compte, mais l'approbation d'un administrateur est requise.

Il s'agit ici d'un mode de création hybride autorisant l'utilisateur à créer son compte mais laissant à l'administrateur le soin de le rendre actif.

Le processus de création se passe donc exactement comme pour le mode précédent, à une différence près : lorsque l'utilisateur a saisi son identifiant et son adresse, il reçoit cette fois le courriel dont le modèle se trouve dans la sous-section *Bienvenue, en attente d'approbation*.

À ce stade, le compte existe mais n'est pas actif car bloqué. Seul l'administrateur a le pouvoir de changer cette situation. Il est informé de l'existence du nouveau compte par un courriel contenant un lien vers le profil de l'utilisateur, et pourra ainsi le rendre actif. C'est aussi à cette étape que l'administrateur pourra affecter des rôles spécifiques au nouvel utilisateur.

Lorsque l'administrateur a débloqué le compte, l'utilisateur reçoit alors le courriel lui indiquant que son accès a été activé et à partir de ce point, les choses se déroulent de la même manière que pour le mode Sans approbation.

Les trois modes étant décortiqués, nous allons choisir pour notre exemple une création Avec approbation en cochant *Les visiteurs peuvent créer leur propre compte, mais l'approbation d'un administrateur est requise*. Ceci nous permettra d'explorer tous les aspects de la création de compte.

Dans la foulée, nous allons aussi cocher *Exiger la vérification de l'adresse e-mail quand un visiteur crée un compte*. Cette option indique à Drupal qu'il doit vérifier les adresses saisies par les utilisateurs en leur envoyant un courrier de confirmation.

Figure 6–16
Activer la vérification des adresses de courriel

☒ Exiger la vérification de l'adresse e-mail quand un visiteur crée un compte
Si cette case est cochée, les nouveaux utilisateurs devront valider leur adresse e-mail avant de pouvoir choisir leur propre mot de passe. Si cette case est décochée, les utilisateurs seront directement informés de leur mot de passe.

Dans la zone de saisie suivante, il vous est possible d'ajouter des remarques qui seront inscrites dans l'en-tête du formulaire de création de compte. Il peut s'agir d'un avertissement légal, ou de toute autre information que vous avez besoin de communiquer avant que l'utilisateur ne crée son compte. Notez qu'il est possible d'insérer des balises HTML dans cette zone. Pour l'exemple, saisissez le code HTML suivant.

Directives pour la création d'un compte :

Attention, ce compte ne sera activé qu'après approbation de l'administrateur du site

Ce texte est affiché dans la partie supérieure du formulaire de création de compte. Il sert à donner des directives ou des informations à vos futurs utilisateurs.

Figure 6-17

La directive de création d'un compte permet de prévenir les utilisateurs du mode de création de compte.

Avant de poursuivre, vous aurez remarqué que dans la liste des modèles dont nous disposons, il reste trois cas que nous n'avons pas encore examiné :

- *Mail d'oubli de mot de passe* : qui, vous l'aurez compris, est le message envoyé à un utilisateur ayant cliqué dans la boîte de connexion sur le lien *Mot de passe oublié* ;
- *E-mail de notification d'un blocage de compte* : qui correspond au courriel envoyé à l'utilisateur dont le compte a été désactivé (état « bloqué ») par l'administrateur ;
- *E-mail de notification de suppression de compte* : qui correspond au courriel envoyé à l'utilisateur dont le compte a été détruit par l'administrateur.

Paramétrage des courriels liés aux étapes de création d'un compte

Comme nous l'avons vu, de nombreux courriels peuvent être envoyés lors du processus de création d'un compte. Il est évidemment possible de modifier chacun des textes contenus dans ces courriels par le biais de modèles disponibles plus bas sur la même page.

Pour chacun de ces modèles, vous pouvez changer l'objet du courriel et son corps. Mais plus important, il vous sera possible d'insérer dans ces textes des mots-clés qui seront au moment de l'envoi remplacés par leur valeur associée, à l'instar des champs d'un traitement de texte.

Figure 6–18
Définition d'un modèle de courriel

The screenshot shows the 'Bienvenue, en attente d'approbation' email template configuration page. At the top, it says 'Personnalisez le mail de bienvenue envoyé aux nouveaux membres après leur enregistrement, lorsque leur approbation est requise. Les variables disponibles sont : !username, !site, !password, !uri, !uri_brief, !mailto, !date, !login_uri, !edit_uri, !login_url.' Below this, there are two main sections: 'Objet :' and 'Corps :'. The 'Objet :' field contains the text 'Détails du compte de !username sur !site (en attente de validation par l'adminis)'. The 'Corps :' field contains the text '!username,' followed by a line break and 'Merci de votre inscription sur !site. Votre demande de compte est actuellement en'.

Ces champs se distinguent par le point d'exclamation qui les précède, en voici la liste :

- **!site** sera remplacé par le titre de votre site ;
- **!uri** sera remplacé par l'adresse de votre site (ex. http://www.mon_site.fr) ;
- **!uri_brief** sera remplacé par l'adresse courte de votre site (ex. www.mon_site.fr) ;
- **!login_uri** sera remplacé par un lien pointant sur une page de saisie du nom d'utilisateur et du mot de passe ;
- **!username** sera remplacé par le nom de l'utilisateur créé ;
- **!password** sera remplacé par son mot de passe ;
- **!mailto** sera remplacé par l'adresse courriel de l'utilisateur ;
- **!login_url**, à ne pas confondre avec **!login_uri**, sera remplacé par un lien qui permettra à l'utilisateur de se connecter au site sans avoir à saisir son mot de passe. Ce lien est à usage unique et représente, par exemple, un moyen bien pratique de le laisser choisir lui-même son mot de passe lors d'une première connexion ;
- **!edit_url** sera remplacé par un lien pointant sur la page de profil de l'utilisateur, lui permettant de changer les paramètres de son compte après s'être authentifié ;
- **!date** sera remplacé par la date de création du compte.

Nous savons maintenant comment fonctionnent les modèles, voyons à présent ce qu'il en est de la création d'un compte.

Signatures et portraits

Les deux dernières sections de la page *Paramètres des utilisateurs* permettent respectivement d'activer et de désactiver la gestion des signatures et des portraits.

Figure 6–19
Activation des signatures

Les signatures sont de courts messages qui seront ajoutés automatiquement à la fin de chaque commentaire posté par l'utilisateur. En activant cette option, ce dernier pourra changer sa signature dans sa page [Mon compte](#). Notez que ces signatures sont dynamiques, c'est-à-dire que si l'utilisateur en change, elle sera aussi modifiée pour l'ensemble de ses commentaires déjà postés.

Figure 6–20
Activation des portraits

Les portraits (ou avatars) sont quant à eux de petites images liées à l'utilisateur qui sont affichées soit dans ses commentaires, soit dans les articles et autres contenus dont il est l'auteur. Ici aussi, le portrait, s'il est activé, est géré dans la page [Mon compte](#).

Dès lors que vous activez les portraits, une nouvelle partie de formulaire apparaît. Elle vous permet de spécifier le dossier dans lequel vont être stockées les images. Généralement, il s'agit du dossier `picture` contenu dans le dossier `files` que nous avons défini au chapitre précédent, à la section « Le paramétrage général de Drupal ».

Figure 6–21
Dossier de stockage des portraits

Vous pouvez aussi spécifier les *Dimensions maximales du portrait*, fixées à 85 × 85 pixels par défaut. Toute image plus grande sera refusée. Pour notre site, nous allons choisir une taille moins importante de 48 × 48 pixels.

Figure 6–22
Dimensions maximales des portraits

Figure 6–23
Poids des portraits

Puis vient la *Taille de fichier maximale des fichiers de portrait*, exprimée en kilo-octets. Par défaut, toute image de plus de 30 Ko sera rejetée.

Taille de fichier maximale des fichiers de portrait :

Taille de fichier maximale des portraits, en Ko.

Comme pour la création de compte, il est possible d'ajouter des directives concernant les portraits. Tout ce que vous saisirez ici sera automatiquement inclus dans le formulaire d'ajout de portrait dans la section *Mon compte* de l'utilisateur. Il peut s'agir là aussi d'un avertissement pour prévenir, par exemple, que le compte sera bloqué si les images ajoutées contiennent à la charte de déontologie du site.

Une fois les paramétrages terminés, appuyez sur le bouton *Enregistrer la configuration*. Drupal vous indique alors que les nouveaux paramétrages sont pris en compte et que le dossier de stockage des portraits a été créé.

Figure 6–24
Message de Drupal attestant
de la sauvegarde de la confirmation

Paramètres des utilisateurs

- Les options de configuration ont été enregistrées.
- Le répertoire « *sites/default/files/pictures* » a été créé.

Gestion des utilisateurs

Création manuelle d'un utilisateur

Quel que soit le mode de création choisi, l'administrateur dispose systématiquement du pouvoir de créer ou de modifier manuellement un compte, et bien sûr de le supprimer. Nous allons mettre cet aspect en pratique en créant le compte de Marine qui aura pour rôle Contributeur.

La première chose à faire est d'aller sur la liste des utilisateurs connus de Drupal. Dans le menu *Gestion des utilisateurs*, cliquez sur le lien *Utilisateurs* ; apparaît alors une liste d'utilisateurs se réduisant pour l'heure au seul administrateur.

Figure 6–25
Pour l'instant, l'administrateur
est seul sur son Drupal.

<input type="checkbox"/>	administrateur	actif	5 semaines 3 jours	il y a 2 min 6 sec	modifier
--------------------------	----------------	-------	--------------------	--------------------	--------------------------

En haut de la page se trouve un onglet *Ajouter un utilisateur* sur lequel nous allons cliquer pour arriver sur le formulaire de création de compte.

Informations du compte

Nom d'utilisateur : *

Marine

Les espaces sont autorisés, mais pas les signes de ponctuation à l'exception des points, traits d'union et car.

Adresse électronique : *

marine@colchique-conseil.fr

Une adresse e-mail valide. Tous les messages de ce système seront envoyés à cette adresse. Cette adresse que si vous désirez obtenir un nouveau mot de passe ou recevoir personnellement certaines nouvelles ou ave

Mot de passe : *

..... Sécurité du mot de passe : **Haute**

Confirmer le mot de passe : *

..... Concordance des mots de passe : **Oui**

Entrez un mot de passe pour le nouveau compte dans les deux champs.

Figure 6–26
Création du compte de Marine

La première chose à faire est de saisir le *Nom d'utilisateur* qui sera l'identifiant utilisé dans la boîte de connexion. Dans notre cas, il s'agira donc de « Marine ».

Suit alors l'adresse courriel de l'utilisateur qui doit absolument être unique, avec le mot de passe en double exemplaire. Rappelez-vous bien que vous devez pour vos tests saisir ici une adresse courriel valide que vous lisez régulièrement. Dans le cas contraire, vous ne recevriez jamais les courriels envoyés par Drupal.

Statut :

☐ Bloqué

☒ Actif

Figure 6–27
Statut de l'utilisateur

À la suite se trouve le statut du compte, qui peut être *Actif* ou *Bloqué*. Ici, nous choisissons logiquement *Actif*.

Rôles :

☒ utilisateur identifié

☒ contributeur

Figure 6–28
Attribution du rôle Contributeur à l'utilisateur Marine

Viennent alors les fameux rôles que nous allons affecter à l'utilisateur Marine. Comme vous le constatez, par défaut Marine est considérée comme *utilisateur identifié*. À cela, nous allons ajouter le rôle *contributeur*.

☒ Notifier l'utilisateur de son nouveau compte

Figure 6–29
Envoi à l'utilisateur de la notification de création de son compte

BON À SAVOIR

Où changer les portraits et signatures ?

Vous constaterez que le choix du portrait utilisateur ou de la signature n'est pas disponible dans ce formulaire. Ceci s'explique par le fait que la création d'un utilisateur ne comporte que les éléments qui lui sont essentiels.
L'utilisateur pourra par la suite modifier tous ces détails via le menu *Mon compte*.

Vous pouvez ensuite cocher la case *Notifier l'utilisateur de son nouveau compte* qui aura pour effet, comme nous l'avons vu, d'envoyer un courriel à Marine pour la prévenir que son compte utilisateur a été activé.
Lorsque nous avons fini, il suffit d'appuyer sur le bouton *Créer un nouveau compte* pour obtenir un message de Drupal confirmant la bonne fin du processus de création.

Un mot de passe ainsi que des instructions détaillées ont été envoyés au nouvel utilisateur *Marine*.

Figure 6–30 Message de confirmation de la création du compte

Si vous obtenez en plus du message précédent l'erreur suivante, reportez vous au chapitre « Mettre en place son environnement de développement », à la section « Résolution des problèmes techniques ».

Unable to send e-mail. Please contact the site administrator if the problem persists.

En cliquant sur le lien *Liste* dans les onglets en haut de page, vous êtes alors renvoyé sur la liste des utilisateurs parmi lesquels apparaît cette fois Marine.

<input type="checkbox"/>	Marine	actif	<input type="radio"/> contributeur	19 sec	il y a 19 se
<input type="checkbox"/>	administrateur	actif		2 heures 52 min	il y a 1 sec

Vous constatez à cette occasion que le rôle *Contributeur* de l'utilisateur est clairement indiqué dans la liste, le rôle *Utilisateur* identifié étant quant à lui sous-entendu.

Création d'un compte par l'utilisateur lui-même

Puisque nous maîtrisons la version manuelle de la création de compte, voyons comment créer un compte de manière automatique. Pour faire ce test, vous devez préalablement vous déconnecter en cliquant sur le lien *Se déconnecter* du menu *Administrateur*.

Vous êtes maintenant sur la page visible par les utilisateurs anonymes, avec sur la droite, une zone permettant de se connecter avec ses identifiants. Pour simuler la création d'un compte, nous allons donc cliquer sur le lien *Créer un nouveau compte*.

Figure 6–31

Drupal n'arrive pas à envoyer de courriel.

Figure 6–32

L'utilisateur Marine est apparu dans la liste.

Attention, ce compte **ne sera activé** qu'après approbation par l'administrateur de ce site.

Nom d'utilisateur : *

Les espaces sont autorisés, mais pas les signes de ponctuation à l'exception des points, traits d'union et caractères de soulignement.

Adresse électronique : *

Une adresse e-mail valide. Tous les messages de ce système seront envoyés à cette adresse. Cette adresse ne sera pas rendue publique, passe ou recevoir personnellement certaines nouvelles ou avertissements.

Figure 6–33
Formulaire de création automatique
d'un nouveau compte

Nous avons ici le formulaire décrit plus haut, et allons saisir le nom d'utilisateur – ici « Philippe » – avec son adresse courriel, avant d'appuyer sur le bouton *Créer un nouveau compte*. Remarquez le texte d'avertissement affiché en haut du formulaire reprenant ce que nous avons saisi dans la section *Paramètres des utilisateurs*.

Drupal renvoie alors un message indiquant que la demande a bien été prise en compte et qu'elle est en attente de validation, ce qui est confirmé par les deux courriels que nous venons de recevoir – l'un à l'administrateur, le second à Philippe.

L'administrateur a pour tâche d'activer ce nouveau compte qui pour l'instant a pour statut *Bloqué*. Il clique pour cela sur le lien indiqué dans le courriel qui l'amène au même formulaire que pour la création manuelle d'un compte, mais prérempli avec les valeurs saisies par Philippe. Il lui suffit alors simplement de cocher la case *Activé* pour rendre le compte opérationnel.

Dans la foulée, l'administrateur va aussi assigner le rôle Contributeur à Philippe et appuyer sur le bouton *Enregistrer*.

Une fois le compte activé, Philippe reçoit un nouveau courriel lui permettant de se connecter, et de finaliser ainsi le processus de création.

Comme vous le voyez, ce mode mixte de création de compte présente de nombreux avantages. Il donne à l'administrateur la possibilité de filtrer les « faux » utilisateurs qui peuvent être des spammeurs. Mais il permet surtout de laisser à l'utilisateur le choix d'un mot de passe qu'il sera seul à connaître, et d'éviter ainsi de faire circuler ce dernier dans un courrier lisible par n'importe qui.

Administrer les comptes utilisateur

Lorsque plusieurs centaines de comptes ont été générés par les utilisateurs eux-mêmes, avec ou sans approbation, il devient impossible de

gérer les activations ou les blocages au cas par cas. Fort heureusement, la liste des utilisateurs que nous avons entraperçue précédemment se révèle être un outil puissant de gestion des comptes qu’il convient de savoir manipuler. Pour tester cette fonctionnalité, nous allons préalablement nous déconnecter pour créer un nouveau compte pour l'utilisateur Karine. Ceci fait, reconnectez-vous en tant qu'administrateur.

Pour retrouver la liste des utilisateurs, vous devez cliquer, dans la section *Gestion des utilisateurs*, sur *Utilisateurs*. La page qui s’affiche alors se compose de trois parties : la zone de filtrage, les actions applicables et la liste elle-même.

Afficher uniquement les utilisateurs où

☐ rôle est

☐ permission

☐ statut

Figure 6–34
La zone de filtrage de la liste des utilisateurs

La zone de filtrage permet de réduire le nombre d’éléments dans la liste en ajoutant des critères de filtrage. La marche à suivre consiste à cocher un seul des trois types de critères : rôle, permission ou statut.

Le critère *rôle* nous permet de filtrer tous les utilisateurs ayant un rôle donné. Le critère *permission* nous permet de choisir l’une des permissions (droit d’accès) que nous avons précédemment étudiées et de n’afficher que les utilisateurs ayant le droit d’accéder à cette fonctionnalité. Enfin, le critère *statut* nous permet de n’afficher que les comptes utilisateur actifs ou bloqués.

Une fois le type de critère choisi, vous devez sélectionner une valeur pour ce critère puis cliquer sur le bouton *Filtre* pour l’appliquer. Ne seront alors affichés dans la liste que les éléments répondant au critère défini. Vous pouvez ensuite ajouter un autre critère et cliquer cette fois sur *Affiner*, ou alors revenir à la liste de départ en appuyant sur le bouton *Réinitialiser*.

La liste contient quant à elle tous les utilisateurs connus du système Drupal. Nous y retrouvons Karine, que nous venons de créer et qui est encore à l’état bloqué.

<input type="checkbox"/>	Nom d'utilisateur	Statut	Rôles	Membre depuis ▼	Dernier accès
<input type="checkbox"/>	Karine	bloqué		3 min 9 sec	jamais
<input type="checkbox"/>	Philippe	actif	<input type="radio"/> contributeur	19 min 49 sec	jamais
<input type="checkbox"/>	Marine	actif	<input type="radio"/> contributeur	53 min 9 sec	il y a 52 min 59 sec
<input type="checkbox"/>	administrateur	actif		2 heures 50 min	il y a 2 min 2 sec

Figure 6–35
La liste des utilisateurs

Cette liste des utilisateurs est constituée de colonnes sur lesquelles il est possible d'effectuer un tri croissant ou décroissant, en cliquant le nombre de fois nécessaire sur le libellé correspondant. Chaque ligne dispose d'une case à cocher faisant office de sélection. La case à cocher sur la ligne d'en-tête du tableau permet de sélectionner ou désélectionner l'ensemble des éléments de la page. Enfin, un lien *éditer* est présent en fin de ligne donnant accès au formulaire de modification du compte.

L'intérêt de la sélection réside dans la dernière zone de la page, les *Options de mise à jour*. Vous y trouverez une liste déroulante d'actions à appliquer sur la sélection. Il est ainsi possible de bloquer ou d'activer un ensemble de comptes d'un seul coup, de supprimer des comptes, mais aussi d'ajouter un rôle particulier à un ensemble de comptes. Ici, nous allons sélectionner *Karine* et choisir l'option de mise à jour *Débloquer les utilisateurs sélectionnés*. Puis, nous allons renouveler l'opération avec cette fois *Ajouter un rôle aux utilisateurs sélectionnés/contributeur*.

Options de mise à jour

contributeur ⇅ Mise à jour

<input type="checkbox"/>	Nom d'utilisateur	Statut	Rôles	Membre depuis ▼	Dernier accès	Opérations
<input checked="" type="checkbox"/>	Karine	actif		4 min 14 sec	il y a 1 sec	éditer
<input type="checkbox"/>	Philippe	actif	<input type="radio"/> contributeur	20 min 54 sec	jamais	éditer
<input type="checkbox"/>	Marine	actif	<input type="radio"/> contributeur	54 min 14 sec	il y a 54 min 4 sec	éditer
<input type="checkbox"/>	administrateur	actif		2 heures 52 min	il y a 1 sec	éditer

Figure 6–36
Attribution à Karine du rôle
Contributeur

Pour appliquer une action à une sélection, il suffit donc de la choisir puis d'appuyer sur le bouton *Mise à jour*. Une page de confirmation résumant la liste des comptes à modifier s'affiche alors et après validation, les modifications sont effectuées.

Changer rapidement d'utilisateur

Dans la suite de ce livre, nous allons très régulièrement endosser le rôle de l'administrateur, du contributeur, ou d'autres rôles, de manière à tester Drupal sous tous les angles. À chaque fois, il faudra donc se déconnecter, puis se reconnecter en saisissant l'identifiant et le mot de passe de l'utilisateur dont nous prenons la personnalité. Cette opération risque de devenir très rapidement fastidieuse.

URL **Module Masquerade**

► <http://drupal.org/project/masquerade>

Figure 6–37
Réglages du module Masquerade

Fort heureusement, il existe un module extrêmement pratique permettant de basculer automatiquement d'un utilisateur à un autre, Masquerade. Téléchargez, installez et activez ce module, puis rendez-vous dans la section *Administration* > *Configuration du site* puis *Masquerade*. Vous pouvez ici paramétrer le module. Le principe est d'autoriser l'administrateur à endosser rapidement l'identité de l'utilisateur contributeur. Pour ce faire, configurez le module comme sur la copie d'écran suivante.

Cela permet à l'administrateur (et seulement lui) de pouvoir rapidement basculer sur l'identité de Marine. Attention de ne pas sélectionner de rôles supplémentaires dans la section *Roles that are considered « administrator »*, sinon vous risquez d'ouvrir une grande faille de sécurité sur votre site.

Enregistrez la configuration, et rendez-vous dans *Administration* > *Construction du site* puis *Blocs*. Nous verrons en détail à quoi correspond cette page plus loin dans ce livre ; pour l'heure, recherchez simplement la ligne *Masquerade* et dans la liste déroulante, sélectionnez la valeur *Barre de gauche* puis enregistrez.

Figure 6–38
Réglages du module Masquerade

Si tout s'est bien passé, vous devez voir apparaître un nouveau menu *Masquerade* à gauche de la page.

Figure 6–39
Le bloc de changement rapide d'utilisateur

Ce menu n'est visible que pour le rôle Administrateur. Lorsque vous cliquez sur le lien *Marine*, vous devenez automatiquement cet utilisateur. Pour preuve, vous voyez apparaître un message d'erreur « Accès refusé » car effectivement, Marine n'a pas le droit de modifier les blocs. Sur le côté gauche, vous remarquez que le menu précédent a lui aussi changé, vous indiquant (en anglais malheureusement) que l'utilisateur courant est bien Marine et vous donnant un lien *Switch back* qui vous permettra de redevenir administrateur.

Ce module nous fera gagner énormément de temps dans la suite de notre découverte. Ceci étant dit, comme pour le module Administration menu que nous avons installé dans le chapitre précédent, le reste de ce livre ne fera pas de référence directe à ce module. C'est donc à vous de l'installer et de l'utiliser si vous le souhaitez.

Règles d'accès

Le dernier aspect de la gestion des utilisateurs consiste à définir un ensemble de règles d'accès au site dont le but est la lutte contre les comportements indésirables. Nous pourrions non seulement bloquer ou autoriser la création de nouveaux comptes, mais aussi l'utilisation de comptes existants en fonction du nom de l'utilisateur, de son adresse électronique ou encore de son nom de domaine.

Vous pouvez ainsi décider qu'aucun utilisateur ne puisse choisir « pirate » comme identifiant, que seuls les utilisateurs provenant d'une adresse IP commençant par 10.0.0 ont le droit de créer un compte ou encore que les utilisateurs provenant du domaine *pirates.com* sont interdits.

Pour paramétrer ces règles d'accès, vous devez vous rendre dans le menu *Administrateur > Gestion des utilisateurs* puis *Règles d'accès*. Drupal affiche alors une page contenant une liste de règles – vide pour l'instant.



Figure 6–40

La liste des règles d'accès au site

Pour ajouter une nouvelle règle, vous devez cliquer sur l'onglet *Ajouter une règle*. Un formulaire s'affiche alors pour permettre le paramétrage de notre restriction.

Cela commence par le choix de l'action de restriction à effectuer : *Auto-riser* ou *Refuser*. Pour notre première règle, nous commencerons par interdire le choix de l'identifiant « pirate » et sélectionnons donc *Refuser*.

Notre restriction portant sur le nom d'utilisateur, nous choisissons dans la section *Type de règle* l'option *Nom d'utilisateur*. Il ne nous reste alors plus qu'à saisir *pirate* dans le champ *Masque*.

Notez que dans ce champ, il est possible d'étendre la restriction en utilisant le caractère % qui se substituera à n'importe quelle chaîne de caractères. Ainsi, pour interdire non pas uniquement l'identifiant « pirate » mais tous les identifiants contenant le mot « pirate », nous saisissons *%pirate%*. De la même manière, si nous voulons interdire un mot à une lettre près, nous utiliserons le caractère *_*. Ainsi, *pirate_* bloquera *pirate* mais aussi *pirate1*, *pirateA*, etc.

Lorsque notre règle est définie, il ne reste plus qu'à appuyer sur le bouton *Ajouter une règle* pour l'activer. Drupal nous repositionne alors sur la liste contenant notre nouvelle règle. Il est alors possible de la modifier, de la supprimer ou d'en ajouter une nouvelle.

Pour être certain que nos règles fonctionnent comme nous le souhaitons, il est possible de les tester en cliquant sur l'onglet *Vérifier les règles*. Apparaissent alors trois zones de saisie correspondant aux trois types de règles. Si nous saisissons *grand_pirate* dans le champ *Nom d'utilisateur* et que nous appuyons sur le bouton *Vérifier le nom d'utilisateur*, le message suivant devrait alors apparaître :

Figure 6-41
Message de confirmation
du bon fonctionnement de notre règle

Le nom d'utilisateur *grand_pirate* n'est pas autorisé.

Nous constatons ainsi que notre règle est maintenant effective. Ce système peut être très utile si vous constatez des attaques massives provenant d'une adresse IP donnée, car il vous sera alors aisé de bloquer les utilisateurs indésirables.

La page Mon compte

Nous avons pour l'instant beaucoup étudié les utilisateurs sous l'angle de l'administrateur. Nous allons maintenant changer de point de vue pour

étudier ce que tous ces réglages impliquent pour l'utilisateur, ce qui se résume essentiellement à son espace purement personnel au sein de Drupal : la page *Mon compte*.

Accéder à Mon compte

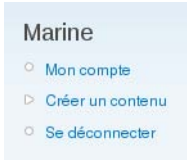


Figure 6-42
Menu d'accès à la page Mon compte

La page *Mon compte* regroupe toutes les informations et les réglages propres à l'utilisateur identifié. Par défaut, la partie *information* que l'on appelle aussi le « profil » de l'utilisateur est relativement pauvre mais de nombreux modules peuvent venir greffer ici de nouvelles sections selon vos besoins.



Figure 6-43
Le profil de Marine

Modification du profil

Ce profil est aussi la page que pourra voir l'utilisateur anonyme qui cliquera sur un nom d'auteur de contenu ou de commentaires. L'onglet *Modifier* est en revanche absolument spécifique à l'utilisateur connecté et ne peut être vu que par l'administrateur ou l'utilisateur qui dispose des droits suffisants.

Nous retrouvons dans cet onglet des informations maintenant connues comme le nom de l'utilisateur, son adresse courriel et son mot de passe. Tout peut être changé dynamiquement par Marine.

Dans la section *Portrait*, nous retrouvons la possibilité donnée à l'utilisateur de télécharger une image qui lui est propre. Une fois ce portrait choisi et le bouton *enregistrer* cliqué, la section *Portrait* change pour permettre la suppression ou le changement de l'image.

Figure 6-44
Le portrait de Marine

Portrait

☐ Supprimer le portrait
Cocher cette case pour supprimer votre portrait actuel.

Transférer un portrait :

Parcourir...

Votre portrait, réel ou virtuel. Les dimensions maximales sont de 48x48 et la taille maximale est de 30 Ko.



La section *Paramètre de signature* permet à Marine de définir le court texte qui sera inséré à la fin de chacun de ses commentaires. Comme nous le disons plus haut, ce message est changé dynamiquement si Marine le modifie dans ses réglages.

La section *Paramètre du contact* est présente parce que nous avons activé le module Contact. Elle permet à Marine de choisir, en cochant la case *Formulaire de contact personnel*, si elle désire ou non pouvoir être contactée par ce biais. Si la case est cochée, tout utilisateur allant sur son profil pourra aussi la contacter.

Enfin, la section *Langue et paramètres régionaux* permet à Marine de définir son propre *fuseau horaire* (si elle travaille par exemple à l'étranger). Lorsque nous aborderons la gestion des langues, c'est aussi à cet endroit que l'utilisateur pourra définir la langue qu'il désire voir utiliser pour son interface.

Étendre le profil de l'utilisateur

Si vous activez le module Profil (fourni en standard avec Drupal), vous avez la possibilité d'ajouter dynamiquement des champs au profil de l'utilisateur. L'ajout de tels champs est alors accessible par le menu *Gestion des utilisateurs*, puis *Profil*.

Titre	Nom	Type	Opérations
Aucun champ disponible.			

Ajouter un nouveau champ

☐ zone de texte simple

☐ zone de texte multiligne

☐ case à cocher

☐ liste de sélection

☐ liste libre

☐ URL

☐ date

Figure 6-45
Ajout de nouveaux champs
au profil des utilisateurs

Nous pouvons, pour l'exemple, cliquer sur l'ajout d'un champ de type *URL*, qui nous donne accès à un formulaire de paramétrage du champ. Saisissez dans *Catégorie* : *Coordonnées*, dans *Titre du champ* : *Page personnelle*, dans *Nom du champ* (qui par une erreur de traduction est dénommé *Nom du formulaire*) : *profil_homepage*, et entrez une éventuelle petite *Explication*.

Paramètres du champ

Catégorie : *

Coordonnées

La catégorie à laquelle le nouveau champ appartiendra. Les catégories permettent de regrouper les champs de manière logique. Exemple de catégorie : "Informations personnelles".

Titre : *

Page personnelle

Le titre du nouveau champ. Il sera montré à l'utilisateur. Exemple de titre : "Couleur favorite".

Nom du formulaire : *

profil_homepage

Le nom du champ. Le nom du formulaire n'est pas montré à l'utilisateur, mais utilisé en interne dans le code HTML et les URLs. Sauf si vous savez ce que vous faites, il est fortement recommandé de préfixer le nom du formulaire avec `profil_`, de manière à éviter des conflits de nommage avec d'autres champs. Les espaces ou tout autre caractère, à l'exception du tiret (-) et du caractère de soulignement (_), sont interdits. Exemple de nom : "profil_couleur_favorite" ou plus simplement "profil_couleur".

Explication :

La page personnelle de l'utilisateur

Figure 6–46
Définition d'un champ Page personnelle

Ensuite, vous pouvez déterminer la visibilité du champ, nous choisirons *Champ public, contenu affiché sur la page de profil, mais non utilisé sur la page de la liste des membres* qui permet de rendre visible le champ à tous, mais de ne pas pour autant l'afficher dans la liste. Lorsque c'est terminé, cliquez sur *Enregistrer le champ*. Vous êtes alors renvoyé sur la page *Profil* où notre nouveau champ est apparu.

Titre	Nom	Type	Opérations
Coordonnées			
Page personnelle	profil_homepage	url	modifier supprimer

Figure 6–47
Le nouveau champ Page personnelle

Si vous vous reconnectez en tant que Marine pour vous rendre sur l'onglet *Modifier* de *Mon compte*, vous constaterez l'apparition d'un nouvel onglet portant le nom de la catégorie que nous avons donné au champ, l'onglet *Coordonnées*.

Marine

Voir

Modifier

Contact

Compte utilisateur

Coordonnées

Figure 6–48
Apparition de l'onglet Coordonnées

Figure 6–49
Saisie des données pour
le nouveau champ du profil

Une fois l'onglet *Coordonnées* enregistré, la valeur du champ est automatiquement ajoutée à la page du profil de Marine.

Figure 6–50
Le profil étendu de Marine

MODULES MAGIQUES Profil et contenu

Le profil d'un utilisateur n'est pas un contenu au sens général du terme. Il jouit de ce fait d'un traitement particulier au sein de Drupal. Dans certains cas, il peut cependant être utile que le profil soit un type de contenu à part entière. C'est le rôle du module *Content profil* :

► http://drupal.org/project/content_profil

L'avantage de cette approche est qu'un profil de type contenu peut être recherché par le même moteur de recherche que les articles, il peut aussi être transformé en PDF, rendu imprimable, envoyé par courriel, etc.

En résumé

- Un rôle est une liste nommée de permissions.
- Chaque module définit son propre jeu de permissions.
- Drupal permet la création d'autant de rôles que nécessaire.
- À un utilisateur donné peuvent être assignés autant de rôles que nécessaire.
- Un utilisateur qui a saisi un identifiant et un mot de passe dispose automatiquement du rôle Utilisateur identifié.
- Le visiteur ne possédant pas d'identifiant et de mot de passe associé a pour unique rôle Utilisateur anonyme.

7

chapitre

Nom	Type	Description	Opérations
Article	story	Un <i>article</i> , similaire dans la forme à une <i>page</i> , est idéal pour créer et afficher du contenu qui informe et pousse les visiteurs à réagir. Revues de presse, actualité, billet de blog peuvent être créés avec un <i>article</i> . Par défaut, un <i>article</i> est automatiquement publié sur la page d'accueil de votre site et permet aux visiteurs d'ajouter des commentaires.	modifier supprimer
Page	page	Une <i>page</i> , similaire à un <i>article</i> dans sa forme, est une méthode simple pour afficher des informations qui changent rarement, comme la page "A propos" d'un site. Par défaut, un contenu de type <i>page</i> ne permet pas aux visiteurs de laisser des commentaires, et celle-ci ne sera pas publiée sur la page d'accueil.	modifier supprimer

Créer du contenu

Nous entrons ici dans le cœur du métier de Drupal : la gestion des contenus. Lors de ce premier contact, nous manipulerons le plus simple des types de contenu, qui est l'article, avant de voir comment créer et mettre en ligne des articles. Puisque mise en ligne implique de facto interaction avec vos visiteurs, nous étudierons pour finir la manière dont vous gérerez leurs commentaires, techniquement mais aussi humainement.

SOMMAIRE

- Paramétrer et créer des types de contenu
- Administrer les contenus et les commentaires
- Comprendre les formats d'entrée

Paramétrer les types de contenu

Les types de contenu

Pour un Drupal fraîchement installé, les types de contenu disponibles sont au nombre de deux : Article et Page. Schématiquement, Article est destiné à la création d'articles, et Page à la création d'éléments statiques du site tels que les informations légales, la présentation de la société ou l'aide.

Pour chaque type de contenu, Drupal offre la possibilité de définir un comportement par défaut, en déterminant la manière de gérer les commentaires, le mode de publication, ou encore la prise en charge des révisions.

Ainsi, le type Article, qui est le type de contenu vivant par excellence, est paramétré par défaut comme pouvant recevoir des commentaires ou encore être publié en page d'accueil. À l'inverse, un contenu de type Page ne permet pas par défaut de recevoir de commentaires et n'a pas vocation à être visible sur la une de votre site.

Il faut cependant bien avoir conscience qu'il s'agit d'un comportement de base, chaque module pouvant potentiellement ajouter de nouveaux paramétrages aux différents types de contenu.

Modification du type de contenu Article

Pour modifier le comportement d'un type de contenu, nous devons tout d'abord redevenir Administrateur, puis dans le menu *administrateur*, nous diriger vers la section *Gestion du contenu*, puis *Types de contenu*.

Nom	Type	Description	Opérations
Article	story	Un <i>article</i> , similaire dans la forme à une <i>page</i> , est idéal pour créer et afficher du contenu qui informe et pousse les visiteurs à réagir. Revues de presse, actualité, billet de blog peuvent être créés avec un <i>article</i> . Par défaut, un <i>article</i> est automatiquement publié sur la page d'accueil de votre site et permet aux visiteurs d'ajouter des commentaires.	modifier supprimer
Page	page	Une <i>page</i> , similaire à un <i>article</i> dans sa forme, est une méthode simple pour afficher des informations qui changent rarement, comme la page "À propos" d'un site. Par défaut, un contenu de type <i>page</i> ne permet pas aux visiteurs de laisser des commentaires, et celle-ci ne sera pas publiée sur la page d'accueil.	modifier supprimer

Figure 7-1
Les types de contenu standards de Drupal

Sur cette nouvelle page, nous voyons apparaître les types de contenu Article et Page évoqués plus haut et nous verrons plus loin qu'il nous sera possible d'en rajouter. Pour l'heure, concentrons-nous sur la ligne correspondant au type de contenu Article et cliquons sur le lien *modifier*.

Identification du contenu

En première partie de ce formulaire, nous trouvons les informations décrivant textuellement le type de contenu.

Identification

Nom : *

Le nom affiché de ce type de contenu. Ce texte sera affiché dans la liste sur la page *créer un contenu*. Ce nom doit commencer par une lettre majuscule et ne doit contenir que des lettres, chiffres, et espaces. Ce nom doit être unique.

Type : *

Le nom interne de ce type de contenu. Ce texte sera utilisé pour construire l'URL de la page *créer un contenu* pour ce type de contenu. Ce nom ne doit contenir que des lettres minuscules, des chiffres, et des caractères de soulignement. Les caractères de soulignement seront convertis en trait d'union lors de la construction des URL de la page *Créer un contenu*. Ce nom doit être unique.

Description :

Un `article`, similaire dans la forme à une `page`, est idéal pour créer et afficher du contenu qui informe et pousse les visiteurs à réagir. Revues de presse, actualité, billet de blog peuvent être créés avec un `article`. Par défaut, un `article` est automatiquement publié sur la page d'accueil de votre site et permet aux visiteurs d'ajouter des commentaires.

Une brève description de ce type de contenu. Ce texte sera affiché dans la liste qui se trouve sur la page *créer un contenu*.

Figure 7-2
Propriétés basiques
d'un type de contenu Article

Le *Nom* du type de contenu, ici *Article*, correspond à ce que le contributeur verra lorsqu'il cliquera sur *Créer un contenu*.

Le *Type* est quant à lui l'identifiant interne utilisé par Drupal pour repérer les contenus de type Article. Il n'y a aucune raison pour nous de modifier cet identifiant.

La *Description* correspond quant à elle au petit texte affiché en dessous du contenu dans la page *Créer un contenu*. Il n'est pas utile de modifier cette partie – sauf si vous désirez la rendre plus explicite pour vos contributeurs.

Les champs du formulaire de contribution

La seconde partie du formulaire – *Paramètres du formulaire de contribution* – nous permet de modifier les intitulés des champs *Titre* et *Corps*. Là aussi, la seule finalité d'un tel changement serait de rendre tout ceci plus compréhensible pour un contributeur.

Le champ suivant, *Nombre minimal de mots*, permet de déterminer à partir de combien de mots Drupal peut autoriser le contributeur à publier son nouveau contenu. Par défaut, la valeur 0 indique qu'il n'y a aucune limite appliquée.

Figure 7-3
Propriétés de publication d'un contenu

**BON À SAVOIR La permission
Administrer les nœuds**

L'administrateur qui crée ou modifie un contenu aura accès à un formulaire beaucoup plus étendu que celui du contributeur, permettant de définir au cas par cas la valeur des éléments que nous choisissons ici. Il en sera de même pour tout utilisateur dont le rôle dispose de la permission *Administrer les nœuds*.

Figure 7-4
Activation des fichiers attachés

Comme pour la création d'un compte ou pour les avatars, vous pouvez ajouter des directives à afficher en tête du formulaire de création d'un nouveau contenu, par exemple pour rappeler au contributeur ses obligations.

Procédure de publication

Options par défaut :

- ☐ Publié
- ☐ Promu en page d'accueil
- ☐ Épinglé en haut des listes
- ☒ Créer une révision

Dans la section *Procédure de publication*, nous paramètrons la manière dont le contenu est traité au moment de l'enregistrement.

Nous retrouvons ici en toute logique le comportement que nous avons constaté pour notre premier contenu. Il sera bien *Publié*, c'est-à-dire rendu visible à tous, ainsi que *Promu en page d'accueil*. *Épinglé en haut des listes* vous permet de conserver le contenu en haut de la liste, et ce même si de nouveaux contenus plus récents sont ajoutés.

Enfin, le dernier choix permet de *Créer une révision* à chaque sauvegarde d'un même contenu. Les révisions sont des photographies à un moment *t* donné du contenu, permettant à l'administrateur de revenir en arrière pour retrouver l'état du contenu avant modification. Nous étudierons cette fonctionnalité en détail un peu plus loin.

Pour notre exemple, décochons *Publié* et *Promu en page d'accueil*, et sélectionnons *Créer une révision*. Par conséquent, une révision sera alors créée à chaque modification du contenu, et seul l'administrateur pourra publier et rendre visible les articles.

Fichiers attachés :

- ☐ Désactivé
- ☒ Activé

Le dernier choix figurant dans cette section correspond à la gestion des fichiers attachés aux contenus. Comme pour un courriel, Drupal permet en effet au contributeur d'ajouter un ou plusieurs fichiers à un contenu. Ce réglage a été ajouté par le module Upload. Dans le cas d'un article, c'est une bonne idée d'activer l'attachement de fichiers pour permettre au contributeur d'enrichir son contenu de divers documents de référence. Pour l'exemple, nous activerons donc cette option.

Paramètres des commentaires

Paramètre par défaut des commentaires :

- ☐ Désactivé
- ☐ Lecture uniquement
- ☒ Lecture/Écriture

Figure 7-5
Définition du mode de gestion des commentaires

Dernière section du paramétrage d'un type de contenu, mais non la moindre : la *Gestion des commentaires*. Le premier paramètre qu'il est possible de définir ici est l'ouverture ou non des commentaires. Drupal nous propose ici trois possibilités :

- **Désactivé** : indique que les commentaires ne seront tout simplement pas autorisés pour ce type de contenu ;
- **Lecture/Écriture** : permet aux visiteurs de lire et écrire des commentaires pour ce type de contenu ;
- **Lecture uniquement** : n'autorise que la lecture des commentaires existants, mais pas la saisie de nouveaux.

BON À SAVOIR À quoi servent les commentaires en Lecture uniquement ?

Autant le choix de l'une des deux autres options est compréhensible, autant l'option *Lecture uniquement* a quelque chose de déroutant.

En effet, quel intérêt y a-t-il à pouvoir lire des commentaires si personne ne peut en écrire ? La réponse est... aucun. Ce choix est sans intérêt dans le paramétrage d'un type de contenu et n'est jamais utilisé à cet endroit.

En revanche, il peut devenir particulièrement utile lorsque vous décidez de fermer les commentaires sur un billet qui est soit trop ancien, soit trop « agité ». Une opération qui sera réalisée par l'Administrateur.

Mode de discussion

Mode d'affichage par défaut :

- ☐ A plat - replié
- ☐ A plat - déplié
- ☐ Par discussion - replié
- ☒ Par discussion - déplié

Figure 7-6
Définition du mode d'affichage des commentaires

Le visiteur dispose de deux liens pour ajouter un commentaire. Soit il commente l'article en lui-même, auquel cas il utilisera le lien *Ajouter un commentaire* qui se trouve juste en dessous de l'article lui-même ; soit il veut répondre à un autre commentaire, auquel cas il cliquera sur le lien *répondre* en dessous de ce commentaire. L'option *Mode d'affichage par défaut* permet de gérer l'affichage des commentaires résultant de ces deux choix.

Lorsque le mode *Par discussion* est sélectionné, l'affichage des commentaires correspondra à celui habituellement trouvé dans un forum. Les commentaires créés par le lien *Ajouter un commentaire* seront affichés au même niveau que l'article et ceux créés par le lien *Répondre* se positionneront sous le commentaire initial, décalé d'un cran vers la droite. Les réponses successives formeront ainsi une arborescence de commentaires.

Si vous choisissez le mode *À plat*, les commentaires seront alors affichés sous la forme d'une liste simple, quel que soit le lien de création utilisé.

Figure 7-7
Ordre d’affichage des commentaires

Dans les deux cas, si vous choisissez l’option *replié*, seul le titre du commentaire sera affiché et il faudra cliquer dessus pour accéder à son contenu. En mode *déplié*, les contenus seront visibles d’emblée.

Dans le cadre de notre exemple, le choix ici sera *Par discussion – déplié*.

Ordre des commentaires

Ordre d’affichage par défaut :

- ☐ Par date - du plus récent au plus ancien
- ☒ Par date - du plus ancien au plus récent

Lorsqu’un visiteur réagit à un article en postant un commentaire, le court texte qui en résulte est alors affiché sous l’article lui-même, soit *Par date – du plus ancien au plus récent*, soit *Par Date – du plus récent au plus ancien*. C’est cet ordre que l’option *Ordre d’affichage par défaut* permet de définir. Sélectionnez *Par Date - plus récent au plus ancien* qui est le mode le plus usuellement utilisé.

Pagination des commentaires

Figure 7-8
Limitation du nombre
de commentaires par page

Commentaires par page par défaut :

De très nombreux commentaires pouvant être ajoutés pour un même article, il peut devenir nécessaire de les afficher sur plusieurs pages. L’option *Commentaires par page par défaut* permet de définir à partir de combien de commentaires Drupal passera à une nouvelle page (50 par défaut).

Gestion des commentaires

Gestion des commentaires :

- ☐ Afficher au-dessus des commentaires
- ☐ Afficher au-dessous des commentaires
- ☐ Afficher au-dessus et au-dessous des commentaires
- ☒ Ne pas afficher

Figure 7-9
Position du panneau
de gestion des commentaires

La gestion des commentaires est un petit tableau de bord permettant aux utilisateurs de régler eux-mêmes le mode d’affichage, l’ordre et la pagination des commentaires. Il leur est aussi possible de mémoriser ces choix.

Par discussion - replié ▾

Par date - du plus ancien au plus récent ▾

50 commentaires par page ▾

Enregistrer les paramètres

Sélectionnez la méthode d'affichage des commentaires que vous préférez, puis cliquez sur "Sauvegarder les paramètres" pour activer vos changements.

Figure 7-10 Panneau personnel de gestion des commentaires

Vous avez ici la possibilité d'activer l'affichage de ce tableau de bord, et de déterminer sa position dans la page.

Commentaires des visiteurs anonymes

Commentaires anonymes :

☐ Les utilisateurs anonymes ne peuvent pas laisser leurs coordonnées

☐ Les utilisateurs anonymes peuvent laisser leurs coordonnées

☒ Les utilisateurs anonymes doivent laisser leurs coordonnées

Figure 7-11
Gestion des commentaires pour les visiteurs anonymes

Lorsqu'un visiteur anonyme (c'est-à-dire non identifié sur le site) poste un commentaire, il est possible, avec l'option *Commentaires anonymes*, de définir s'il doit fournir une adresse courriel valide, de lui interdire de la laisser ou simplement de lui laisser le choix de le faire ou non. Généralement, il est d'usage d'obliger le visiteur anonyme à laisser une adresse (troisième choix). Libre à vous d'activer cette option qui alourdit visuellement un peu la page.

Champ sujet d'un commentaire

Champ sujet du commentaire :

☒ Désactivé

☐ Activé

Figure 7-12
Désactivation du champ sujet du commentaire

Le commentaire peut être précédé par un sujet. Vous pouvez choisir par l'option *Champ sujet du commentaire* de rendre ce dernier optionnel. Sélectionnez donc ici le mode *Désactivé* car nous n'en avons pas l'usage.

Gestion de l'aperçu

Aperçu :

☐ Facultatif

☒ Obligatoire

Figure 7-13
Activation de l'aperçu sur un nouveau commentaire

La personne qui commente peut obtenir un aperçu de son commentaire avant qu'il soit mis en ligne en appuyant sur le bouton *Aperçu*. Dans la mesure où les visiteurs anonymes ne peuvent pas modifier leurs com-

BON À SAVOIR Aperçu et utilisateurs identifiés

Malheureusement, il n'est pas possible de conditionner l'aperçu au fait d'être anonyme. Une fois que ce mode est choisi pour un type de contenu donné, il est actif pour tout le monde, y compris pour les utilisateurs identifiés qui pourtant peuvent modifier leurs commentaires.

Figure 7-14
Position du formulaire
de soumission des commentaires

mentaires une fois ceux-ci validés – contrairement aux authentifiés –, c’est une bonne idée de rendre cet aperçu *Obligatoire*.

Emplacement du formulaire de commentaires

Emplacement du formulaire de soumission de commentaire :

- ☒ Afficher sur une autre page
- ☐ Afficher en-dessous de la contribution ou des commentaires

L’option *Emplacement du formulaire de soumission de commentaire* précise l’emplacement de la zone de saisie du commentaire : celle-ci peut être située sur une autre page ou à la fin de l’article. Le choix entre ces deux modes est avant tout une question de goût, pour l’heure sélectionnez *Afficher sur une autre page*.

Notre type de contenu Article est maintenant prêt à l’emploi, et nous pouvons créer notre premier article.

Création d’un contenu de type Article

Sortons à présent de notre rôle Administrateur pour devenir un Contributeur. Pour cela, nous commençons par nous déconnecter et nous reconnecter pour devenir Marine, l’utilisateur possédant le rôle Contributeur que nous avons créé au chapitre précédent.

Le rôle Contributeur

Une fois connecté en tant que Contributeur, la première chose que nous constatons est que le menu anciennement *Administrateur* a été remplacé par un menu équivalent mais beaucoup moins riche, appelé cette fois *Marine*. La logique de ce menu reste identique et regroupe toutes les actions que ce rôle a le droit d’effectuer sur le système. Il reflète donc très exactement les permissions que nous avons paramétrées dans le chapitre précédent.

Marine

- [Mon compte](#)
- [Créer un contenu](#)
- [Se déconnecter](#)

Figure 7-15
Accès à la création d’un nouveau contenu

Nous trouvons dans ce menu l’entrée *Mon compte* permettant à l’utilisateur de changer les paramètres de son profil, et le lien *Se déconnecter*

pour retrouver un état « anonyme » et éventuellement se reconnecter en tant qu'Administrateur.

Ceci étant posé, entre ces deux éléments de menu, nous avons *Créer un contenu* qui est, comme vous pouvez l'imaginer, le point d'entrée obligatoire pour créer un contenu de n'importe quel type. En cliquant sur ce lien, vous voyez apparaître la liste des types de contenu que Marine a le droit de créer, liste pour l'instant réduite au seul type Article.



Figure 7-16
Création d'un contenu de type Article

Insistons sur le fait qu'un utilisateur identifié qui n'aurait pas le rôle Contributeur ne verrait jamais le menu *Créer un contenu*. Ce dernier correspond en effet très exactement au droit *Créer un article* que nous avons spécifiquement donné au rôle Contributeur, et donc à l'utilisateur Marine puisque ce rôle lui a été affecté.

Formulaire de création d'un article

Pour passer à la création en elle-même, cliquez simplement sur le lien *Article*. La page suivante est un formulaire qui nous permettra de saisir notre contenu de type Article.

Créer 'Article'

Titre : *

Corps :

Créer un résumé à partir du curseur

- Les adresses de pages web et de messagerie électronique sont transformées en liens automatiquement.
- Tags HTML autorisés : <a> <cite> <code> <u> <dl> <dd>
- Les lignes et les paragraphes vont à la ligne automatiquement.

[Plus d'informations sur les options de formatage](#)

Enregistrer Aperçu

Figure 7-17
Formulaire de saisie d'un contenu de type Article

Comme vous le voyez, le formulaire de saisie de contenu est très simple, composé seulement d'un titre et d'un corps. Il s'agit là de la structure basique d'à peu près tous les types de contenu.

Nous allons commencer par saisir le *Titre* de notre article, par exemple « Mon premier article ». Puis dans le champ *Corps*, nous allons saisir un court texte pour l'exemple :



The screenshot shows a form with two main sections. The first section is labeled 'Titre : *' and contains a text input field with the value 'Mon premier article'. To the right of this field is a button labeled 'Créer un résumé à partir du curseur'. The second section is labeled 'Corps :' and contains a larger text area with the text 'Ceci est l'introduction de mon article' on the first line and 'Et ceci est son développement...' on the second line.

Figure 7-18
Titre et corps de l'article

Notez au passage la partie grise en dessous de la zone d'édition du corps qui vous permet d'en agrandir la surface. C'est une constante ergonomique que nous retrouverons sur l'ensemble des zones de texte des formulaires Drupal.

Le résumé d'un article

Ce que Drupal appelle le « résumé » s'apparente davantage à l'introduction d'un article, et a une certaine importance. En effet, lorsque viendra le moment d'afficher une liste de contenus – par exemple la liste des billets récents – en première page de notre site, c'est cette introduction qui sera affichée et non l'article complet. Vous prendrez ainsi progressivement l'habitude de raisonner sur deux versions d'un même article : la version dite « abrégée » qui ne contiendra que le titre et l'introduction, et la version « complète ».

Pour créer ce résumé, placez simplement le curseur entre les deux lignes de notre corps et cliquez sur le bouton *Créer un résumé à partir du curseur* (*Split summary at cursor*). La zone de saisie va alors se séparer en deux parties avec en haut le résumé et en bas le reste de l'article.



The screenshot shows the same form as Figure 7-18, but with additional options at the top. There is a checkbox labeled 'Afficher le résumé en mode pleine page' which is checked, and a button labeled 'Fusionner le résumé et le corps du message'. Below these, the 'Corps :' section is shown with the text 'Ceci est l'introduction de mon article' in the top part and 'Et ceci est son développement...' in the bottom part, separated by a horizontal line.

BON À SAVOIR Notation HTML pour une séparation manuelle de l'introduction

Le même effet peut être obtenu en insérant la balise spéciale `<!--break-->` entre l'introduction et le reste de l'article.

Figure 7-19
Séparation de l'introduction et du développement de l'article

Il est possible à tout moment de revenir en arrière en cliquant sur le bouton *Fusionner le résumé et le corps du message* (*Join summary*). La case à cocher *Afficher le résumé en mode pleine page* permet quant à elle de déterminer si ce résumé n'est visible que lors de l'affichage du contenu dans une liste (décochée) ou s'il est aussi présent lorsque le contenu est

affiché au complet (cochée). Dites-vous que lorsque cette case est cochée, la première zone est effectivement une introduction à notre texte, et lorsqu'elle est décochée il ne s'agit que d'un résumé.

Décrire votre création

Nous verrons dans le prochain chapitre à quoi sert concrètement la section *Informations sur les révisions*. Pour l'heure, dites-vous qu'il est de bon ton, lorsqu'on travaille à plusieurs, de décrire chacune de ses créations de contenu, mais aussi les modifications qui suivront. Nous saisisons ici un petit message qui indiquera aux autres nos motivations pour créer cet article. Cette description, nous le verrons un peu plus loin, sera visible dans l'onglet *Révisions* du contenu.

Figure 7-20
Une courte description de la contribution

Attacher un fichier

Drupal, à travers le module Upload que nous avons activé dans le paramétrage du type de contenu Article au cours du chapitre 6, nous permet d'attacher des fichiers à nos articles. Nous disposons donc à la fin du formulaire d'une section dédiée à cet usage.

Figure 7-21
Attachement d'un fichier à l'article

Pour ajouter un fichier, appuyez sur le bouton *Parcourir...* Lorsque vous validez votre sélection, le chemin complet de votre document doit apparaître dans le champ *Attacher un nouveau fichier*. Pour l'insérer dans le contenu, appuyez sur le bouton *Attacher*. Une liste apparaît alors contenant les informations sur le fichier que vous venez d'attacher.

ERGONOMIE
Redimensionnement des zones de texte
Notez que la découpe automatique du résumé, ainsi que la possibilité de redimensionner les zones, ne sont accessibles que si vous avez activé JavaScript dans votre navigateur.

Figure 7–22
La liste des fichiers attachés

Supprimer	Liste	Description	Taille
	<input type="checkbox"/>	<input checked="" type="checkbox"/> Manuel utilisateur.pdf <small>http://www.colchique-conseil.fr/sites/www.colchique-conseil.fr/files/Manuel%20utilisateur.pdf</small>	274.03 Ko

ERGONOMIE L'ordonnancement des listes

À chaque fois que Drupal propose des listes pour lesquelles il est utile de pouvoir changer l'ordre des éléments, un mécanisme JavaScript permet d'effectuer cette opération de manière interactive. Ce comportement est signalé par des symboles en forme de croix qui se trouvent en début de chaque ligne et n'est disponible que si JavaScript est activé sur votre navigateur.

Par une manipulation classique de glisser-déposer sur ces croix, il est ainsi possible de changer l'ordre des éléments de la liste. Faites cependant attention, ces modifications ne sont pas prises en compte tant que vous n'avez pas cliqué sur le bouton *Enregistrer* de la page en cours.

Figure 7–23
Organisation des fichiers attachés

BON À SAVOIR Utiliser les fichiers attachés dans les contenus

Si vous décochez la case *Liste* d'un fichier attaché, celui-ci sera invisible pour le lecteur. L'intérêt de cette option est de vous permettre d'intégrer par exemple une image à votre contenu. En effet, en attachant un fichier `mon_image.png`, vous constatez que son adresse est `http://mon_site/files/mon_image.png`. Il est donc possible de mettre dans le corps de l'article une balise HTML comme ceci :

```

```

Dans cette liste, il est possible de changer la *Description* qui par défaut est le nom du fichier. Si vous laissez la case à cocher de la colonne *Liste* activée, cette description sera utilisée dans la liste des attachements qui sera présentée au lecteur de l'article. La case décochée, le document lui sera invisible.

MODULES MAGIQUES Des dossiers paramétrables pour les fichiers attachés

Par défaut, Drupal stocke les fichiers dans le dossier `sites/default/files`. Ce n'est pas un problème si vous avez peu de fichiers mais cela devient vite gênant lorsque ces derniers se multiplient.

Le module Upload Path vous permet de spécifier vous-même le chemin de sauvegarde en utilisant des variables. Il est ainsi possible de spécifier le chemin `[nid]` qui permettra le stockage des fichiers dans un sous-dossier `NID` du dossier `files`. Le `NID` étant l'identifiant unique du contenu, nous aurons donc un dossier par contenu contenant l'ensemble des attachements de ce contenu.

Faites cependant attention au paramètre *Clean file upload filenames*, qui implique le renommage du fichier avant écriture sur le disque. Décochez cette case pour garder le même nom de fichier que celui d'origine.

► <http://drupal.org/project/uploadpath>

Supprimer	Liste	Description	Taille
	<input type="checkbox"/>	<input checked="" type="checkbox"/> La copie d'écran <small>http://www.colchique-conseil.fr/sites/default/files/copie%20ecran_0.png</small>	105.4 Ko
	<input type="checkbox"/>	<input checked="" type="checkbox"/> Le manuel utilisateur <small>http://www.colchique-conseil.fr/sites/default/files/Manuel%20utilisateur.pdf</small>	274.03 Ko

* Les modifications réalisées sur cette table ne seront enregistrées que lorsque le formulaire sera soumis.

Enfin, si vous désirez supprimer le fichier attaché, cochez simplement la case correspondante dans la colonne *Supprimer*. Le fichier sera effectivement effacé dès que nous enregistrerons l'article.

Aperçu avant sauvegarde

Notre article rédigé, nous avons la possibilité de le prévisualiser avant publication en cliquant sur le bouton **Aperçu**. Devrait alors apparaître le même formulaire, précédé du rendu final de l'article en version abrégée et en version complète.

Vous pouvez encore modifier le texte, le formulaire de saisie étant placé juste en dessous de l'aperçu.

Aperçu de la version abrégée

Mon premier article

8 juillet 2009 - 11:10am — Marine

Ceci est l'introduction de mon article

Aperçu de la version complète

Mon premier article

8 juillet 2009 - 11:10am — Marine

Ceci est l'introduction de mon article

Et ceci est son développement...

Fichier attaché	Taille
La copie d'écran	105,4 Ko
Le manuel utilisateur	274,03 Ko

Figure 7–24
Aperçu de l'article

Comme vous le voyez, notre fichier attaché est bien présent dans la liste en dessous de l'article dans sa forme complète et l'utilisateur peut le télécharger en cliquant sur le lien correspondant.

Enregistrement de l'article

Lorsque vous jugerez votre article satisfaisant, vous pourrez appuyer sur le bouton **Enregistrer**. Devrait alors apparaître la version longue de l'article précédé d'un message de Drupal vous indiquant que le contenu a bien été créé dans la base de données.

En haut de l'article se trouve son titre, suivi de deux onglets **Voir** et **Modifier**. Un clic sur le bouton **Modifier** aura pour effet de vous renvoyer dans le formulaire précédent, vous permettant ainsi d'en modifier le contenu.

MODULES MAGIQUES

Éviter de se marcher sur les pieds

Si un autre contributeur modifie avant vous un contenu que vous décidez à votre tour de modifier avant que ce dernier n'ait eu le temps d'enregistrer, vos modifications seront rejetées par Drupal qui vous indiquera que le contenu a changé entre le moment de l'édition et celui de l'enregistrement. Vos modifications seront donc perdues. Pour éviter cela, vous pouvez utiliser le module Checkout qui va verrouiller les modifications sur un contenu de manière à ce que les modifications par d'autres utilisateurs soient rejetées dès le début. Lorsque vous sauvegarderez le contenu, il sera automatiquement déverrouillé. Ce module est indispensable pour le travail en équipe.

► <http://drupal.org/project/checkout>



Figure 7–25
L'article en version complète

BON À SAVOIR

Qui a le droit de modifier un article

Les boutons *Voir* et *Modifier* ne sont visibles que pour les utilisateurs ayant le droit de modification sur l'article, c'est-à-dire son auteur et l'administrateur du site. De manière générale, ces utilisateurs peuvent être l'auteur de l'article dont le rôle dispose du droit *Modifier ses propres articles*, et tous les utilisateurs bénéficiant du droit *Modifier les articles*.

En bas de l'article s'affiche une zone de liens, qui pour l'instant n'en compte qu'un seul : *Ajouter un commentaire*. Ce lien sera utilisé par vos visiteurs pour commenter votre article.

Enfin, le corps de l'article est surmonté d'un en-tête indiquant sa date de création, suivi du nom de l'utilisateur qui l'aura créé.

Notre billet est enregistré, mais comme nous avons paramétré notre type de contenu Article sans l'option de mise en ligne automatique, il n'est pour l'instant visible que par l'Administrateur (qui a tous les droits) et par Marine (qui en est l'auteur). C'est ce que signifie ce fond rouge. Vous pouvez vous déconnecter pour vérifier en tant que visiteur anonyme la non-visibilité de ce billet.

Nœuds, chemin interne et alias

Si vous regardez l'URL affichée dans la barre d'adresse de votre navigateur, vous constatez qu'elle est de la forme `http://www.colchique-conseil.fr/node/1`. Si on écarte l'adresse du site lui-même, reste ce que Drupal appelle le chemin interne de la page : `node/1`.



Figure 7–26
Le chemin interne de l'article

Toute page de Drupal a un chemin interne. Par exemple, le chemin interne lorsque vous ajoutez un nouveau contenu est `node/add`, celui pour afficher l'administration est `admin`, celui pour la configuration du site `admin/build`, et ainsi de suite. Pour revenir à notre contenu, `node/1` signifie le « nœud ayant pour identifiant 1 ».

Un nœud pour Drupal (en anglais *node*) est un contenu au sens large du terme. Ici `node/1` est notre article.

Si vous cliquez sur le bouton *Modifier* de la page, vous constaterez que le chemin change en `node/1/edit`. Ce chemin signifie que nous sommes en train d'éditer (en anglais *edit*) le nœud ayant pour identifiant 1.

Certains peuvent considérer qu'un article ayant pour chemin `node/1` n'est pas très explicite pour l'utilisateur et préféreraient que l'URL soit `http://www.colchique-conseil.fr/mon_premier_article`. C'est sans aucun doute une question de goût, mais Drupal respecte ceux de chacun : il vous offre la possibilité de définir (ou de redéfinir) des alias sur les chemins, en vous permettant de créer les URL de votre choix. Ce fonctionnement est géré par le module Path que nous avons activé en début d'ouvrage.

Créer un alias sur le chemin de notre article se fait très simplement en descendant dans la section *Paramètres du chemin d'URL*. Vous pouvez ici taper le chemin de votre choix. Ne perdez pas de vue que ceci est un chemin : il doit en conséquence ne contenir que des lettres et des chiffres sans accents, et certains caractères sont interdits (`?`, `&`, `#` par exemple).

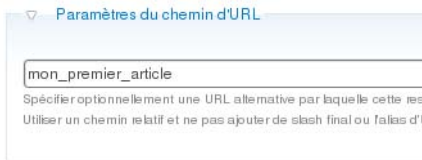


Figure 7-27
Définition d'un alias sur l'article

En enregistrant le contenu (sans oublier de saisir un petit message explicatif de votre action dans le journal), vous constaterez que le chemin de l'article et son URL ont changé, affichant ainsi notre alias.



Figure 7-28
Le chemin lisible de l'article

Toutefois, si vous tapez à la main dans la barre d'adresse de votre navigateur l'URL `http://www.colchique-conseil.fr/node/1`, vous constaterez que votre contenu reste disponible depuis l'ancien chemin. Ce chemin, toujours valide quel que soit l'alias que vous déciderez de lui affecter, est appelé *permalien* (*permalink* en anglais), ou encore « chemin interne » de Drupal.

Pour mieux comprendre comment tout ceci fonctionne, redevenons Administrateur et dirigeons-nous vers *Construction du site* > *Alias d'URL*.

Nous retrouvons ici l'alias que nous venons de créer. Vous pouvez le *modifier* ou le *supprimer*. En cliquant sur *Ajouter un alias*, vous pouvez aussi en créer de nouveaux. Le tout est de connaître le chemin interne de la page que vous cherchez à lier et ensuite de saisir l'alias de votre choix.

ATTENTION Les alias sont des choses changeantes

Si vous communiquez l'URL d'un article sur le Web, pensez toujours à donner son permalien et jamais un alias. Car un alias peut être changé très facilement et votre lien serait dès lors inutilisable. Ne communiquez un chemin avec alias que lorsque vous êtes certain qu'il ne changera jamais.

Figure 7–29
La liste des alias

Alias de filtre

Alias ▲	Système	Langue	Opérations	
mon_premier_article	node/1	Français	modifier	supprimer

MODULES MAGIQUES

Création automatisée d’alias

Le module contribution Pathauto permet de générer automatiquement les alias en fonction de divers éléments comme le titre de l’article, la date, le nom de l’auteur, etc.

Attention à l’utilisation de la date dans l’alias d’un article, car cet alias changera à chaque fois que vous modifierez l’article, ce qui peut rendre le référencement problématique. Choisissez la nomenclature la plus stable possible, comme le titre de l’article seulement.

► <http://drupal.org/project/pathauto>

Notez que vous pouvez avoir plusieurs alias pour un même chemin interne.

Pour finir sur les alias, il est important de bien comprendre que lorsque vous utilisez cette fonctionnalité, vous obligez Drupal à rechercher dans une table en base de données les alias pour un chemin interne donné, ce qui aura un impact sur les performances. Pour un site à fort trafic, veillez à bien mesurer cet impact en termes de vitesse avant de généraliser l’usage des alias.

Administration des contenus

Publication de l’article

Il est temps à présent de rendre notre contenu visible à tous, et cela consiste à modifier le statut du contenu pour le passer de *non publié* à *publié* et *promu en page d’accueil*.

Pour effectuer cette tâche, reconnectons-nous avec la casquette Administrateur, avant de nous diriger une fois de plus vers *Administrer > Gestion du contenu > Contenu*. Nous arrivons ainsi sur une page qui rappelle dans sa structure la *Gestion des utilisateurs*.

Comme pour cette dernière, cette page permet d’effectuer des actions de masse sur un ensemble de contenus. Nous pouvons ainsi filtrer les contenus suivant un statut donné (*publiés*, *non publiés*, *promus en page d’accueil*, etc.), un type de contenu donné (Article, Page, etc.).

Nous y retrouvons une zone de filtrage, une zone d’actions et une liste de contenus pour l’instant réduite à notre seule production.

Figure 7–30
La liste des articles du site

<input type="checkbox"/>	Titre	Type	Auteur	Statut	Langue	Opérations
<input type="checkbox"/>	Mon premier article	Article	Marine	non publié	Indépendant de la langue	éditer

Sur cette ligne apparaissent des notions connues telles que *Titre*, *Type* (de contenu) et *Auteur*. Nous constatons également que l'article est *non publié*, ce qui correspond bien au paramétrage défini pour ce type de contenu.

Dans la section de mise à jour, nous retrouvons la possibilité de supprimer des contenus mais aussi d'en changer le statut. Nous pourrions donc très facilement publier notre article en le cochant et en sélectionnant l'action *Publier*, puis en appuyant sur *Mise à jour*.

Mais il est plus intéressant pour notre apprentissage de cliquer sur le lien *éditer* afin de retrouver le formulaire d'édition de contenu que nous connaissons, affiché cette fois-ci au complet et sans limitations car vous êtes dans le rôle Administrateur.

Pour l'essentiel, il s'agit bien du même formulaire, comportant juste des sections supplémentaires. Certaines de ces sections sont familières, car il s'agit des mêmes options dont nous avons défini les valeurs par défaut lors du paramétrage du type de contenu : *Paramètres des commentaires* et *Options de publication*. Si nous modifions dans le cas présent ces valeurs par défaut, elles s'appliqueront pour ce seul contenu uniquement.

Pour publier notre article, il nous suffit d'étendre la section *Options de publication* et de cocher *Publié* et *Promu en page d'accueil* avant d'enregistrer le contenu. Notre article est donc maintenant en ligne et présent en page d'accueil.

Retour à la page d'accueil

Par défaut, Drupal affiche sur la page d'accueil du site la liste des contenus classés du plus récent au plus ancien. Notre premier article étant à présent publié, nous pouvons vérifier qu'il est bien visible à cet endroit stratégique du site. Pour effectuer cette vérification, le mieux est de se déconnecter complètement, de manière à voir le site tel que le visiteur le verra, et de cliquer en haut de la page sur le lien *Accueil*.



Figure 7–31

Retour à la page d'accueil par le fil d'Ariane
(en anglais, breadcrumbs)

Nous voici donc renvoyés à la page d'accueil, qui n'est plus la page de bienvenue que nous avons découverte lors de l'installation de Drupal, mais bel et bien la liste de nos contenus, n'affichant pour l'instant qu'un seul article. Déconnectons-nous pour vérifier ce que voit le visiteur anonyme.



Figure 7–32 La version page de garde de notre site avec notre premier article

Nous retrouvons ici le résumé de notre article, sa date de création, les informations sur son auteur et le lien pour ajouter un commentaire. Au regard de la vue précédente, les nouveautés sont le lien *En savoir plus* qui pointe sur la version complète de l'article, et *2 fichiers attachés* donnant accès aux fichiers que nous avons transférés dans le contenu. Cliquons maintenant sur *En savoir plus* pour découvrir l'article dans sa version complète.

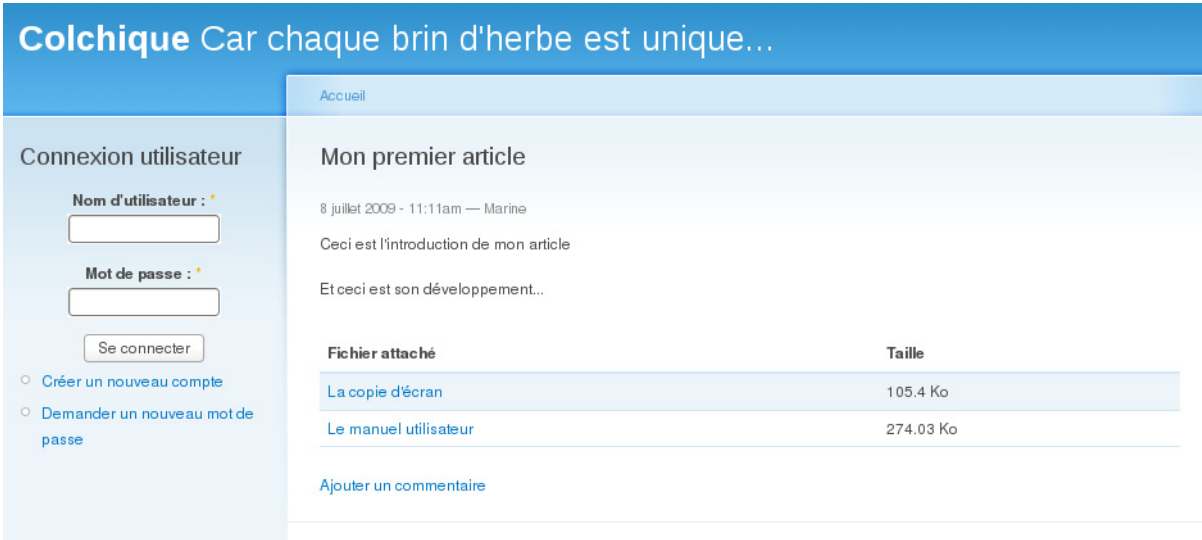


Figure 7–33 L'article en version complète

Supprimer un article

Si un contenu ne vous convient plus, il vous est évidemment possible de le supprimer. Pour cela, en étant Administrateur ou l'auteur du contenu, il suffit de cliquer sur *Modifier*, puis en bas du formulaire sur *Supprimer*. Après confirmation, le contenu sera définitivement détruit avec tous les fichiers et les commentaires qui auraient pu y être attachés.

Notez qu'il est aussi possible de retirer le statut *publié* d'un article si vous désirez juste le rendre invisible pour les visiteurs.

MODULES MAGIQUES Déplacer un article

Il arrive parfois qu'on ait à regrouper deux articles en un seul. Généralement, il suffit de recopier le contenu du second article à la suite du premier, de mettre en forme, de l'enregistrer, puis de supprimer l'article en trop.

Le problème qui se pose alors est que vos visiteurs ont peut-être mis dans leur liste de marque-pages l'adresse de l'article désormais disparu. Ou alors, cet article a été mis en référence sur un autre site et son URL amène donc à une erreur « 404 Document non trouvé ».

La solution consiste à utiliser le module Path Redirect qui va vous permettre d'ajouter l'adresse de l'article supprimé et de lui associer celle de l'article fusionné. L'utilisateur qui arrivera sur l'ancienne adresse sera alors automatiquement redirigé sur la nouvelle.

► http://drupal.org/project/path_redirect

Une autre option est d'utiliser le module Search 404 qui va tout simplement transformer un chemin introuvable en un résultat de recherche, permettant ainsi de retrouver le contenu perdu. Attention, cependant, car cela ne fonctionne bien que si vous avez utilisé des alias sur vos chemins, et dans tous les cas, cela peut générer une plus forte charge sur vos serveurs si vous avez beaucoup de pages « non trouvées ».

► <http://drupal.org/project/search404>

Les commentaires

Ajout d'un commentaire

Nous avons passé du temps à paramétrer les commentaires pour le type de contenu Article, il est maintenant temps d'en voir le résultat. Pour cela, le mieux est de se placer dans la peau d'un visiteur anonyme. Déconnectez-vous de votre compte, et dirigez-vous vers la page d'accueil.

[Ajouter un commentaire](#) [En savoir plus](#) 1 fichier attaché

Ajouter un nouveau commentaire à cette page.

Figure 7-34

Lien d'ajout d'un nouveau commentaire

Le visiteur peut commenter l'article par le lien *Ajouter un commentaire* présent à la fin du texte, que ce soit dans sa forme abrégée ou sa forme

complète obtenue après avoir cliqué sur le lien *En savoir plus*. Ceci l’emmènera sur un écran vierge de saisie, puisque nous avons paramétré le système pour afficher le formulaire sur une nouvelle page.

Répondre

Votre nom : *

Roberto

Adresse électronique : *

roberto@miopalmo.com

Le contenu de ce champ sera maintenu privé et ne sera pas affiché publiquement.

Page d'accueil :

http://roberto.com

Commentaire : *

Article passionnant mais manquant néanmoins un peu de substance.

Figure 7–35

Le formulaire de commentaire pour les utilisateurs anonymes

Dans ce formulaire, le visiteur anonyme pourra saisir son nom (ou pseudo), son adresse courriel (que nous avons rendue obligatoire dans les paramètres), sa page web (optionnelle) et bien évidemment son commentaire.

Puisque nous avons rendu obligatoire l’aperçu du commentaire avant publication, notre utilisateur n’a d’autre choix que de finir par le bouton *Aperçu*.

Aperçu

Article passionnant mais

8 juillet 2009 - 12:02pm — Roberto (non vérifié)

Article passionnant mais manquant néanmoins un peu de substance.

Figure 7–36

L’aperçu du commentaire

Le visiteur a donc encore la possibilité de modifier son commentaire avant de finalement cliquer sur *Enregistrer*. Le commentaire apparaît alors en bas de l’article.

Commentaires

Article passionnant mais

8 juillet 2009 - 12:03pm — Roberto (non vérifié)

Article passionnant mais manquant néanmoins un peu de substance.

[répondre](#)

Figure 7–37

Le commentaire une fois publié

Si Marine se reconnecte, les liens sous la forme abrégée de l’article en page d’accueil ont alors changé, lui indiquant qu’un commentaire est présent.

Figure 7–38

Indication des commentaires présents et encore non lus

[1 commentaire](#) [1 nouveau commentaire](#) [En savoir plus](#)

Réponse aux commentaires

Marine a alors la possibilité de visualiser le commentaire et bien évidemment d'y répondre en cliquant sur le lien *répondre*.



Figure 7-39
Le lien de réponse à un commentaire

Le formulaire de Marine, qui est une utilisatrice authentifiée, contient moins de champs que celui du visiteur anonyme. En effet, Marine a la possibilité de modifier dans le menu *Mon compte* sa page web, et son adresse courriel est déjà connue du système.



Figure 7-40
Le formulaire de commentaire pour un utilisateur identifié

En revanche, comme nous l'avons vu, Marine est obligée d'utiliser le bouton *Aperçu* avant de pouvoir *Enregistrer* sa réponse.



Figure 7-41
La réponse une fois publiée

MODULES MAGIQUES

Faire de la bouture de commentaires

Beaucoup d'utilisateurs, lorsqu'ils désirent répondre à un commentaire, se trompent et cliquent sur le lien *ajouter un commentaire* au lieu de *répondre*. Du coup, les commentaires ne sont plus affichés de manière décalés et les conversations deviennent plus difficiles à suivre.

La solution est un module appelé Comment mover, qui permet de replacer les commentaires dans le bon ordre. Faites cependant attention, ce module est pour l'instant à l'état « instable » et il faudra attendre la version « stable » pour l'utiliser en production.

► http://drupal.org/project/comment_mover

Parce que nous avons choisi le mode *Par Discussion – déplié*, l’affichage du nouveau commentaire de Marine apparaît sous celui du visiteur, décalé d’un cran vers la droite.

Administration des commentaires

Tout ceci est pour le moment parfait, mais voyons ce qui se passe lorsqu’un mauvais esprit se mêle à la partie.

Imaginons donc un nouveau visiteur qui lui aussi poste un commentaire, mais un peu moins constructif.

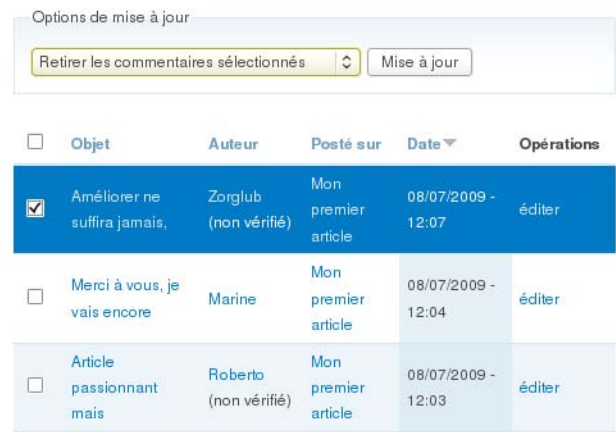
Figure 7–42
Un commentaire de troll



Ce genre de commentaires n’apporte pas grand-chose au sujet et n’a qu’un objectif : mettre le bazar dans le fil de discussion. C’est ce qu’on appelle un *troll* (voir l’aparté page suivante). C’est à l’administrateur ici d’intervenir.

Pour gérer les commentaires douteux, l’administrateur dispose d’un outil efficace disponible dans *Gestion des contenus > Commentaires*.

Figure 7–43
Administration des commentaires



Cette page vous permet de lister l’ensemble des commentaires et d’effectuer des actions sur une sélection. Nous allons ici pouvoir *Supprimer le commentaire* ou *Retirer le commentaire* indécidat que nous aurons préalablement sélectionné.

CULTURE Trolls des villes et trolls des montagnes

Ce type de contributeur indélicat est communément appelé un troll. Cette famille de nuisibles existe depuis la naissance même de l'Internet, de quand date cette appellation. Son seul objectif est de passer le temps en montant les contributeurs les uns contre les autres. Vous avez plusieurs manières de gérer ce genre de spécimen avant de passer au stade ultime de la destruction massive :

- Tout d'abord, une critique – même exprimée avec un manque de civisme caractérisé – n'est pas forcément un troll. Analysez bien à froid le commentaire avant de le supprimer, car supprimer une critique constructive est risqué pour l'image de votre site.
- Utilisez la règle « si tu me cherches, tu ne me trouves pas », et surtout ne rentrez jamais dans le jeu d'un troll. La règle ancestrale sur ce point est le fameux « *Don't feed the troll* » (ne nourrissez pas le troll). Ignorez-le tout simplement et le troll déçu s'ennuiera avant d'aller *troller* ailleurs.
- Si vous devez répondre, ne gardez que ce qu'il y a de positif dans le commentaire du troll et ignorez le reste.
- Si la conversation s'emballe et que les autres contributeurs sont pris à parti, n'hésitez pas à bloquer le troll en utilisant les règles d'accès vu au chapitre précédent. Vous pouvez aussi fermer la conversation en passant les commentaires en lecture seule. Mais ceci risque de ne pas être apprécié par les *non-trolls* ; à utiliser avec prudence donc, et toujours en prévenant préalablement de votre démarche via un commentaire explicatif.

Pour plus d'information sur la *trollogie* (la science d'étude des trolls), veuillez vous référer à l'adresse suivante :

► <http://uzine.net/article1032.html>

Lorsque vous retirez un commentaire, il sera non pas détruit, mais placé en file d'approbation. C'est un bon moyen de temporiser une attaque de troll sans être aussi définitif qu'une suppression. Nous allons donc appliquer cette action et voir ce qui en découle.

Tout d'abord, cliquez sur l'onglet *File d'attente d'approbation*. Vous retrouvez ici une liste identique à la précédente, mais ne contenant que les commentaires non publiés, celui de notre troll pour l'instant. Nous disposons ici aussi d'actions sur la sélection permettant soit de republier, soit de supprimer définitivement.

Avant de supprimer ce commentaire, refaisons un tour sur l'article pour voir comment apparaît pour l'administrateur un commentaire non publié. Vous remarquez aussitôt que le nombre de commentaires pour notre article est repassé à deux. Enfin, vous constaterez que le commentaire « dépublié » est toujours visible, mais avec cette fois un fond rouge.

Vous pouvez vous déconnecter pour constater que le commentaire non publié n'est effectivement visible que pour l'administrateur. Ce constat fait, vous avez maintenant tout ce qu'il faut pour détruire définitivement ce commentaire.

Figure 7–44
Le fond rouge indique
que l'article n'est pas publié
et qu'il est en file d'approbation.



Les formats d'entrée

Les formats d'entrée sont une chose assez mal nommée. En réalité, la bonne expression devrait être « format du texte » car il s'agit en réalité de définir la syntaxe utilisée pour saisir l'article ou, de manière plus générale, des transformations à effectuer sur ce texte pour en obtenir une version XHTML.

Fonctionnement d'un format d'entrée

Nous allons à présent modifier l'article que nous venons de créer en cliquant sur son lien *modifier* pour le rendre plus présentable.

Figure 7–45
Modification de notre premier article

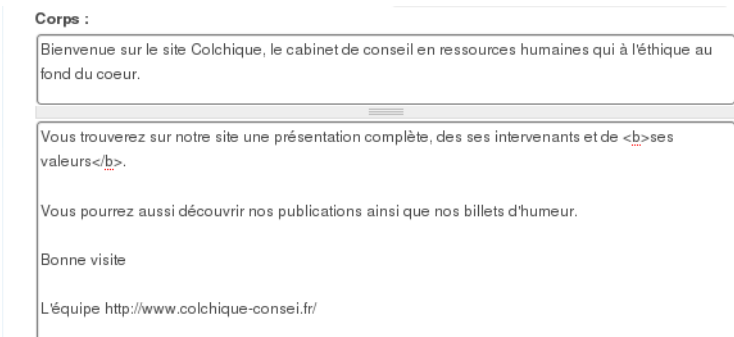


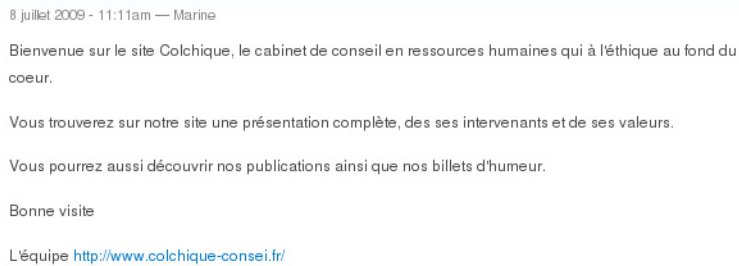
Figure 7–46
Aide sur le format d'entrée actif

Avant de valider, prêtez attention au texte qui se situe juste en dessous de la zone d'édition :

- Les adresses de pages web et de messagerie électronique sont transformées en liens automatiquement.
- Tags HTML autorisés : `<a>` `` `` `<cite>` `<code>` `<u>` `` `` `<dl>` `<dt>` `<dd>`
- Les lignes et les paragraphes vont à la ligne automatiquement.

Nous y apprenons que les passages à la ligne seront transformés automatiquement en paragraphes, que les adresses courriel et les liens web seront rendus cliquables et que seules les balises listées sont autorisées.

Une fois le contenu saisi, enregistrez-le. Le résultat devrait ressembler à cela :



8 juillet 2009 - 11:11am — Marine

Bienvenue sur le site Colchique, le cabinet de conseil en ressources humaines qui à l'éthique au fond du coeur.

Vous trouverez sur notre site une présentation complète, des ses intervenants et de ses valeurs.

Vous pourrez aussi découvrir nos publications ainsi que nos billets d'humeur.

Bonne visite

L'équipe <http://www.colchique-conseil.fr/>

Figure 7-47
L'article une fois publié

Comme vous le voyez, les sauts de lignes sont correctement respectés, ce qui indique que des balises XHTML `<p></p>` ont été ajoutés. De même, l'adresse du site est devenue un lien. En revanche, la balise de mise en gras `` a été éliminée, ce qui est logique puisqu'elle ne fait pas partie des balises autorisées.

Cet ensemble de règles de transformation constitue ce qui est appelé un « format d'entrée », à comprendre comme « règles de formatage appliquées aux contenus saisis en entrée », la « sortie » étant du code XHTML. Nous pouvons vérifier cela en utilisant le clic droit de notre navigateur et en sélectionnant dans le menu contextuel l'option *Code source de la page*.

```
<p>
  Vous trouverez sur notre site une présentation complète de la société, de ses intervenants et de
ses valeurs.
</p>
<p>
  Vous pourrez aussi y découvrir nos publications ainsi que nos billets d'humeur.
</p>
<p>Bonne visite</p>
<p>
L'équipe de
<a href="http://www.colchique-conseil.fr" title="http://www.colchique-conseil.fr">
  http://www.colchique-conseil.fr
</a>
</p>
```

Comme vous le constatez, notre simple texte a donc bien été transformé en code XHTML valide par le format d'entrée.

De prime abord, ce format apparaît efficace et bien adapté à des contributeurs qui ne connaissent pas le XHTML. Mais il peut sembler restrictif pour qui désire un plus grand degré de liberté. Comme souvent avec Drupal, il ne s'agit là que de la vision par défaut, le système de formats d'entrée pouvant aller bien au-delà de ces simples transformations.

Configuration d'un format d'entrée

Pour comprendre comment tout ceci fonctionne, redevenons Administrateur et rendons-nous dans la *Configuration du site* puis dans *Formats d'entrée*.

Par défaut	Nom	Rôles	Opérations
<input checked="" type="radio"/>	Filtered HTML	Tous les rôles peuvent utiliser ce format	configurer
<input type="radio"/>	Full HTML	Aucun rôle ne peut utiliser ce format	configurer supprimer

Définir le format par défaut

Figure 7-48
La liste des formats d'entrée

Nous trouvons ici les deux formats d'entrée connus de Drupal, avec *Filtered HTML* comme format par défaut. Dans la colonne *Rôles*, nous voyons qu'un format d'entrée peut être limité à certains rôles seulement. Le format d'entrée Filtered HTML étant défini par défaut pour tous les rôles, c'est donc celui qui est utilisé depuis le début. Cliquons sur son lien *configurer* pour voir ce qui se cache derrière.

Nous retrouvons sur cette page le nom du format, suivi de la liste des rôles qui ont le droit de l'utiliser. S'agissant d'un format fondamental, il est impossible de modifier ces choix car il doit rester accessible à tous.

Nom :

Entrez un nom unique pour ce format d'entrée.

Rôles

Tous les rôles pour le format par défaut doivent être activés,

☒ utilisateur anonyme

☒ utilisateur identifié

☒ contributeur

Figure 7-49
Les rôles autorisés à utiliser un format

Dans la section suivante, nous avons l'ensemble des règles de filtrage qui composent notre format.

Filtres

Choisissez les filtres qui seront utilisés dans ce format d'entrée.

- ☒ **Convertisseur de saut de ligne**
Convertit les sauts de lignes en HTML (c'est à dire en balises
 et <p>).
- ☒ **Correcteur HTML**
Corrects faulty and chopped off HTML in postings.
- ☒ **filtre HTML**
Allows you to restrict whether users can post HTML and which tags to filter out. It will also remove harmful content such as JavaScript events, JavaScript URLs and CSS styles from those tags that are not removed.
- ☒ **filtre URL**
Transforme les adresses web et e-mail en liens cliquables.

Figure 7-50

La liste des filtres entrant dans la composition du format d'entrée

Nous retrouvons ici les règles qui nous ont été exposées lors de l'édition du contenu :

- *Convertisseur de saut de ligne* est le filtre qui transforme les retours chariot en balises `
` et `<p></p>` ;
- *Correcteur HTML* corrige (légèrement) le code HTML mal écrit ;
- *Filtre HTML* empêche l'utilisation de certaines balises non compatibles XHTML (comme `` ou `<i>`) ou dangereuses pour le site, comme `<script>` qui permettrait au contributeur de rajouter du JavaScript dans les contenus ;
- *Filtre URL* est quant à lui responsable de la transformation des adresses web et courriel en leur version XHTML cliquable.

Nous avons donc là un filtrage parfait pour des commentaires, car très restrictif et ne laissant au visiteur mal intentionné ou étourdi que peu de possibilités de commettre des erreurs graves pour le site. Cependant, si l'application de certains filtres ne vous convient pas, vous avez bien sûr la possibilité de les décocher ; après enregistrement de la configuration, ils ne seront plus utilisés.

Certains filtres sont paramétrables. Pour avoir accès à leur configuration, vous devez cliquer sur l'onglet *Configurer*. Apparaît alors un formulaire permettant de déterminer la longueur maximale des liens pour le filtre URL, ou encore la liste des balises autorisées pour le filtre HTML. Vous pouvez par exemple décider d'ajouter la balise `` (attention, le code généré ne sera dès lors plus certifié XHTML). Dès que vous aurez enregistré la configuration, les nouveaux réglages seront actifs, y compris pour les contenus déjà créés.

BON À SAVOIR L'impact sur les performances des formats d'entrée

Le format d'entrée d'un contenu est appliqué lors du premier affichage de la page. Pour ne pas fabriquer à chaque fois le code XHTML et perdre ainsi en performance, Drupal va stocker le résultat dans une table de la base de données. Ce genre de table est appelé un cache. Ainsi, lors du second affichage et des suivants, c'est la valeur placée en cache qui sera utilisée, réduisant d'autant le temps de création de la page.

Lorsque vous enregistrez des modifications sur un format d'entrée (activation ou désactivation d'un filtre, ordonnancement ou paramétrage), Drupal va simplement vider la table des caches associée aux filtres. Lors du prochain affichage, ce sera donc le nouveau format qui sera utilisé.

BON À SAVOIR Pas de droits, pas d'édition

Attention, si vous créez un contenu avec un format d'entrée donné, seuls les utilisateurs dont le rôle le spécifie auront le droit de le modifier, et ce, même si vous avez explicitement ajouté les permissions d'édition du type de contenu. Ainsi, dans le cas où vous auriez donné le droit d'édition à un rôle, mais où un utilisateur ayant ce rôle ne verrait pas l'onglet *Modifier*, vérifiez les droits sur le format d'entrée qu'utilise le contenu.

Enfin, il est important dans certains cas de pouvoir choisir l'ordre dans lequel les filtres sont exécutés. C'est le rôle de l'onglet *Réordonner* que nous allons étudier un peu plus loin.

Comme pour les fichiers attachés, il est possible via cette page de positionner les filtres les uns par rapport aux autres en faisant du simple glisser-déposer. Une fois la configuration enregistrée, le nouvel ordonnancement sera actif, y compris pour les contenus déjà créés.

Droits sur un format d'entrée

En retournant sur la liste des filtres (vous pouvez passer directement par le lien *Format d'entrée* qui se trouve en haut de la page), nous découvrons qu'en plus du format d'entrée Filtered HTML, il existe le format d'entrée Full HTML.

Comme son nom le laisse présager, ce format est beaucoup moins limité que le précédent et n'est en conséquence disponible par défaut que pour le rôle Administrateur. En cliquant sur le lien *configurer*, nous constatons tout d'abord que *Filtre HTML* est désactivé, mais surtout qu'aucun rôle n'est coché.

Il peut pourtant être utile de laisser à nos contributeurs, qui sont des personnes de confiance, la possibilité d'utiliser un tel format qui leur laissera toute liberté pour écrire des billets directement en XHTML. Pour ce faire, il suffit simplement de cocher le rôle Contributeur et d'enregistrer ensuite la modification du format.



Figure 7-51
Ajout du rôle Contributeur à ce format

Il ne reste plus alors qu'à se reconnecter en tant que Marine et à modifier le dernier contenu que nous avons créé, pour constater qu'une nouvelle section dédiée aux formats d'entrée est à présent disponible en dessous de la zone d'édition du corps :



Figure 7-52
Section permettant de choisir un format

L'apparition des deux formats est en soit logique, car le rôle Contributeur avait déjà accès au format d'entrée Filtered HTML de manière implicite. Drupal n'avait auparavant pas jugé utile de faire apparaître une liste de formats, car il n'y avait qu'une seule possibilité.

Avec ce deuxième format disponible, le contributeur peut maintenant choisir celui qui lui convient le mieux pour un contenu donné. Si nous cochons cette fois *Full HTML* et que nous enregistrons notre contenu, nous constatons que le rendu final n'est plus exactement le même :

8 juillet 2009 - 11:11am — Marine

Bienvenue sur le site Colchique, le cabinet de conseil en ressources humaines qui à l'éthique au fond du coeur.

Vous trouverez sur notre site une présentation complète, des ses intervenants et de **ses valeurs**.

Vous pourrez aussi découvrir nos publications ainsi que nos billets d'humeur.

Bonne visite

L'équipe <http://www.colchique-conseil.fr/>

Figure 7-53

Le même article avec le format d'entrée Full HTML

L'apparition de l'expression « ses valeurs » en gras nous indique que le format d'entrée ne filtre à présent plus cette balise.

Le format des commentaires

Les commentaires que saisissent les visiteurs utilisent eux aussi un format d'entrée. Si la personne qui commente est un anonyme, c'est évidemment le format d'entrée Filtered HTML qui sera utilisé. Pour cette raison, prenez toujours soin de laisser activés les filtres essentiels pour ce format (principalement *Filtre HTML*).

Si la personne qui commente est un utilisateur identifié, il verra apparaître en dessous de sa zone de saisie de commentaires la même section *Format d'entrée* que pour la création d'un contenu. Un contributeur pourra donc saisir un commentaire sous forme XHTML en sélectionnant le format d'entrée *Full HTML*.

Le format d'entrée PHP

En activant le module Filtre PHP, un troisième format d'entrée appelé « PHP code » apparaît alors dans la liste des formats. Comme Full HTML, ce format n'est par défaut accessible qu'à l'administrateur. Vous pouvez cependant autoriser l'utilisation de ce format pour un rôle particulier, mais vérifiez très soigneusement ce que vous faites car ce filtre est loin d'être anodin, comme nous allons le voir.

URL	Module	External links filter
►	http://drupal.org/project/elf	

Figure 7-54
Activation du module External links filter

Ce format permet de saisir directement du code PHP dans les contenus en utilisant les balises `<?php ?>`. Cette technique peut paraître bien pratique pour qui maîtrise PHP, mais peut aussi bien se révéler très traître. En effet, dans la hâte du développement d'un projet, on a tôt fait de coder énormément de choses dans ce type de pages notamment des fonctionnalités qui auraient davantage leur place dans un module Drupal...

L'inconvénient flagrant est que ce code va devenir très vite difficile à maintenir, à déboguer ou simplement à détecter, pour qui n'aura pas réalisé ces pages.

L'autre inconvénient de taille est que l'activation pour un contenu donné de ce format d'entrée désactive implicitement le système de cache. De ce fait, l'utilisation abusive de ce format affectera les performances de votre site.

Si vous êtes amené à l'utiliser, n'allez jamais plus loin que l'affichage de variables ou l'appel d'une fonction, et n'incluez jamais de boucles ou, pire encore, de requêtes SQL.

Ajouter de nouveaux filtres

À présent, vous avez certainement noté que les formats sont des compositions de filtres et que chacun de ces filtres est lui-même fourni par un module. Ainsi, tous les filtres présents par défaut sont contenus dans le module Filter (section *Core – obligatoire*). En installant les bons modules de contribution, vous pourrez donc avoir accès à de nouveaux filtres.

Prenons un exemple concret avec le module External links filter. Ce module ajoute un nouveau filtre qui vérifie si un lien est interne ou externe, et dans le second cas, affiche une petite icône indiquant à l'utilisateur qu'en cliquant sur ce lien, il sortira de notre site.

Une fois le module installé et activé, il est automatiquement présent dans chacun des formats d'entrée existants. Activons-le pour le format Full HTML. Comme vous le voyez, le nouveau filtre est bien présent. Cochez-le et enregistrez le format.



Figure 7-55
Ajout du nouveau filtre
au format Full HTML

Avant de tester, il nous faut régler un petit problème. En effet, ce nouveau filtre a besoin d'un lien à modifier pour fonctionner. Or, il est pour l'instant exécuté avant *Filtre URL*. Nous devons donc modifier l'ordre d'exécution des filtres pour placer ce nouveau filtre après *Filtre URL*. Une fois ce paramétrage enregistré, le format est prêt à l'usage. Nous allons vérifier que tout fonctionne correctement en visualisant l'article de bienvenue.



Figure 7-57
Le filtre External links filter en action

Ajouter de nouveaux formats

Jusqu'à présent, nous avons modifié et paramétré des formats d'entrée existants. Il est cependant possible de créer ses propres formats en composant de manière inédite les filtres installés.

Par exemple, il peut être utile de proposer aux contributeurs une syntaxe plus simple que du code HTML pour saisir les contenus. Il existe pour cela plusieurs syntaxes telles que Wiki, phpBB ou encore Markdown. Toutes permettent par des annotations simples (par exemple en utilisant des * pour définir des listes) de créer un contenu formaté.

Le module Markdown contient un filtre capable de comprendre la syntaxe Markdown que vous pouvez découvrir sur Wikipedia (<http://fr.wikipedia.org/wiki/Markdown>).

Une fois le module installé et activé, retournez dans la liste des formats d'entrée et cliquez sur l'onglet *Ajouter un nouveau format*. Vous retrouvez alors la page maintenant bien connue d'édition des formats.

Nommez le nouveau format *Markdown* et cochez le rôle *Contributeur*. Ensuite, dans la section *Filtres*, décochez tous les filtres et cochez *Markdown*.

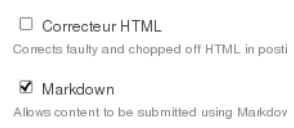


Figure 7-58
Ajout du filtre Markdown au nouveau format



Figure 7-56
Réordonnancement de l'ordre des filtres

BON À SAVOIR

Pourquoi notre URL est-elle externe ?

L'icône est bien présente, le filtre a donc correctement fait son travail. Le fait que le lien sur notre page d'accueil soit ici présenté comme un lien « externe » n'est pas une erreur du module External links filter. En effet, ce dernier considère que tous les liens commençant `http://` sont des liens externes. Pour régler ce problème, nous devrions saisir le lien dans le contenu de l'article de la manière suivante :

```
<a href='/'>  
www.cochique-conseil.fr</a>
```

URL Module Markdown

► <http://drupal.org/project/markdown>

Cliquez sur [Enregistrer](#) et le nouveau format est prêt à l'emploi. Vous pouvez le tester en tant que Marine, par exemple en créant un nouveau contenu tel que celui-ci :

The screenshot shows the Drupal 6 content creation form. The 'Titre' (Title) field contains 'Essai d'article en utilisant MarkDown'. The 'Corps' (Body) field contains the text 'Ceci est une **liste**' followed by a bulleted list with two items: 'Item 1' and 'Item 2'. Below the form, the 'Format d'entrée' (Input format) section shows 'Filtered HTML' as the default, but 'Markdown' is selected with a radio button. The 'Markdown' option includes a description: 'You can use [Markdown syntax](#) to format and style the text. Also see [Markdown Extra](#) for tables, footnotes, and more.'

Figure 7–59
Utilisation du nouveau format Markdown

Dans la section *Format d'entrée*, nous avons maintenant trois choix, dont *Markdown* que nous sélectionnons. Il ne reste plus à Marine qu'à enregistrer pour juger du résultat.

The screenshot shows the rendered content. At the top, it says 'mar, 06/09/2009 - 20:01 — Marine'. Below that, the text 'Ceci est une **liste**' is displayed. Underneath, there is a bulleted list with two items: 'Item 1' and 'Item 2'. The list is properly formatted with bullet points.

Figure 7–60
Le contenu formaté par le filtre Markdown

Le filtre Markdown a bien joué son rôle et a transformé notre syntaxe triviale en un contenu proprement formaté.

Mettre en place un éditeur visuel

La zone d'édition de Drupal peut sembler bien pauvre pour un contributeur qui n'a aucune envie d'apprendre une syntaxe XHTML ou même Markdown. Ce genre de contributeur – généralement non technique – préférerait une édition plus intuitive comme le permet un traitement de texte.

MODULES MAGIQUES

Un format d'entrée par type de contenu

Vous pouvez dans certains cas avoir besoin de forcer un format d'entrée par défaut pour un type de contenu. Cette fonctionnalité n'est malheureusement pas incluse en standard dans Drupal. Il est cependant possible d'y remédier très facilement en utilisant le module Filter by node type.

► <http://drupal.org/project/filterbynodetype>

Il n'y a rien en standard dans Drupal pour permettre la mise en place d'un tel éditeur, mais comme toujours, les modules contributions sont là pour le compléter, et dans ce cas, il s'agit du module Wysiwyg API (WYSIWYG pour *What You See Is What You Get*, qui est l'expression consacrée pour les éditeurs visuels).

Ce module n'est pas un éditeur visuel en soi, mais bien mieux que cela, c'est un système permettant d'utiliser l'éditeur visuel de votre choix. Il existe en effet de nombreux projets libres d'éditeurs fondés sur JavaScript, chacun ayant ses inconvénients et ses avantages. Pour notre exemple, nous allons utiliser TinyMCE, l'un des plus connus.

Pour commencer, vous devez télécharger, installer et activer le module Wysiwyg API. Ceci étant fait, nous commençons par faire un tour dans son paramétrage en allant dans *Configuration du site* > *Wysiwyg*.

Laissons pour l'instant le début du formulaire et intéressons-nous plutôt à la section *Installation Instructions* (eh oui, encore un module non traduit en français).

URL Module Wysiwyg API

► <http://drupal.org/project/wysiwyg>

<p>openWYSIWYG (Download)</p> <p>Not installed.</p> <p>Extract the archive and copy its contents into a new folder in the following location: sites/all/libraries/openwysiwyg</p> <p>So the actual library can be found at: sites/all/libraries/openwysiwyg/scripts/wysiwyg.js</p>
<p>TinyMCE (Download)</p> <p>Not installed.</p> <p>Extract the archive and copy its contents into a new folder in the following location: sites/all/libraries/tinymce</p> <p>So the actual library can be found at: sites/all/libraries/tinymce/jscripts/tiny_mce/tiny_mce.js</p>
<p>markItUp (Download)</p> <p>Not installed.</p> <p>Extract the archive and copy its contents into a new folder in the following location: sites/all/libraries/markitup</p> <p>So the actual library can be found at: sites/all/libraries/markitup/markitup/jquery.markitup.js</p>
<p>NicEdit (Download)</p> <p>Not installed.</p> <p>Extract the archive and copy its contents into a new folder in the following location: sites/all/libraries/nicedit</p> <p>So the actual library can be found at: sites/all/libraries/nicedit/nicEdit.js</p>

Figure 7–61
Choix de la bibliothèque JavaScript
d'édition visuelle

Figure 7-62
Téléchargement de l'archive de TinyMCE

Vous avez ici un aperçu de tous les éditeurs visuels que ce module prend en charge. Autant dire la très grande majorité. Il vous est à chaque fois donné le nom du projet, un lien [download](#) pour télécharger l'éditeur, et l'endroit où vous devez l'extraire. En effet, ce module ne contient aucun éditeur par défaut.

Pour notre exemple, nous allons installer l'éditeur TinyMCE. Pour cela, cliquez sur le lien [download](#) et téléchargez l'archive [jQuery package](#) (une version optimisée pour jQuery, qui est installé en standard avec Drupal).



Revenez ensuite sur la page d'accueil de TinyMCE, puis cliquez sur le lien [Language pack services](#). Nous allons télécharger la traduction française pour l'interface de l'éditeur, en cochant pour cela la ligne [French](#) et en cliquant en bas de page sur le bouton [download](#).

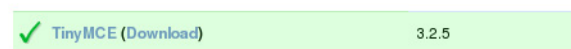
Figure 7-63
Téléchargement du pack de langue française



Créez, comme indiqué dans l'aide du module Wysiwyg, un dossier `sites/all/libraries` et décompressez la première archive dans ce dossier de manière à former un sous-dossier `tinymce`. Lorsque vous avez terminé, le chemin `sites/all/libraries/tinymce/jscripts/tiny_mce/tiny_mce.js` doit donner accès à un fichier valide. Ensuite, décompressez le contenu de l'archive de langue dans le dossier `sites/all/libraries/tinymce/jscript/tiny_mce`.

Ceci fait, retournez sur la page de configuration du module Wysiwyg API, et pressez [F5](#) (ce qui réactualise la page). TinyMCE est à présent installé et prêt à être utilisé.

Figure 7-64
La bibliothèque TinyMCE est maintenant opérationnelle



Avant de poursuivre, il va nous falloir bien comprendre ce que fait exactement le couple Wysiwyg API-TinyMCE.

TinyMCE est un script JavaScript, il s'exécute donc sur le navigateur. Son rôle est de permettre une édition visuelle (mise en gras, titres, etc.) et de fournir au moment de l'enregistrement un contenu formaté en XHTML. Drupal reçoit ce code XHTML et le stocke dans la base de données.

Lorsque le contenu devra être affiché, Drupal ressortira de la base de données le contenu XHTML précédemment envoyé par TinyMCE, et le fera passer dans le format d'entrée sélectionné pour ce contenu. Ce format d'entrée fournira ainsi un contenu XHTML éventuellement modifié, qui sera ensuite affiché dans le navigateur web.

De ce type de fonctionnement, nous pouvons déduire deux faits importants.

- TinyMCE est une sorte de super zone d'édition qui nous évite de taper du code XHTML. Pour Drupal, il n'y a donc aucune différence entre un code XHTML tapé à la main dans la zone d'édition standard et du code XHTML fabriqué par TinyMCE.
- Comme TinyMCE ne fabrique que du code XHTML, le format d'entrée utilisé doit être capable de comprendre le XHTML. Ce pourra donc être le format Full HTML, ou encore Filtered HTML, mais en aucun cas Markdown.

C'est pour cela que le module Wysiwyg API n'active pas directement TinyMCE, mais vous propose via son paramétrage d'associer TinyMCE au format d'entrée de votre choix. À vous d'en choisir un qui soit compatible avec le code généré par cet éditeur, en l'occurrence Filtered HTML ou Full HTML.

Pour créer l'association, retournons voir le haut de la page de paramétrage de Wysiwyg API que nous avons laissé tout à l'heure de côté. C'est ici qu'il est possible de sélectionner un éditeur pour un format donné. Sélectionnez donc TinyMCE pour le format Full HTML.

ATTENTION Toute le monde n'aime pas les éditeurs visuels

Le risque que vous encourrez à associer l'éditeur visuel au format d'entrée Full HTML est celui d'énervier vos contributeurs aimant saisir leur code à la main, et ce même s'il est possible de le désactiver à la volée. Dans ce cas, utilisez l'option *Autoriser les utilisateurs à choisir un éditeur par défaut* dans le *Paramétrage par base* du module Wysiwyg. Ainsi les utilisateurs pourront, depuis leur compte, décider si oui ou non, il désirent utiliser l'éditeur visuel.

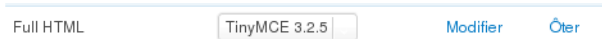


Figure 7-65

Sélection d'un format d'entrée à associer à l'éditeur visuel

Vous avez vu apparaître deux liens, l'un permettant de modifier le paramétrage de l'éditeur, l'autre de casser l'association entre le format d'entrée et l'éditeur. Cliquez sur le lien *Modifier*. Dans le formulaire qui apparaît, vous pourrez régler tous les détails de l'éditeur (boutons affichés, apparence, etc).

Dans la première section, *Basic setup*, commençons par choisir la langue française (*fr*).



Figure 7-66

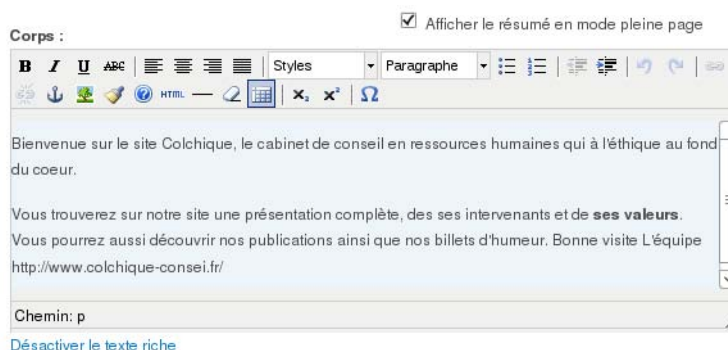
Choix de la langue française pour TinyMCE

Ensuite dans la section *Cleanup and output*, activons les options :

- *Verify HTML*, qui bloquera les tentatives d'écriture de code dangereux ;
- *Convert tags to styles*, pour produire une mini-feuille de style et par conséquent un meilleur code XHTML ;
- *Remove linebreaks*, pour enlever les retours chariot inutiles ;
- *Apply source formatting*, qui permettra de produire un code XHTML propre et lisible ;
- *Force cleanup on standards paste*, pour épurer le code XHTML des « saletés » introduites lors d'un copier-coller à partir de Microsoft Word.

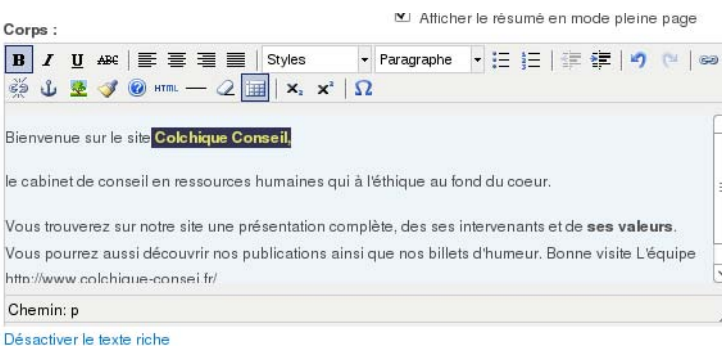
Le paramétrage est à présent terminé et vous pouvez l'enregistrer. Pour tester le nouvel éditeur, redevenons Marine. Éditions l'article de bienvenue et, dans les formats d'entrée, sélectionnons si ce n'est pas déjà fait *Full HTML*. TinyMCE devrait apparaître automatiquement à la place de la zone d'édition standard de Drupal.

Figure 7-67
L'éditeur visuel en action lors
de l'édition d'un contenu



Comme vous le voyez, l'éditeur a bien pris en compte notre ancien formatage (« ses valeurs » en gras). Notez aussi le lien *Désactiver le texte riche* qui vous permet de basculer de l'éditeur visuel à l'éditeur standard, et vice versa. Vous pouvez maintenant tester l'éditeur en appliquant quelques nouvelles mises en forme.

Figure 7-68
Application d'une mise en forme
par l'éditeur visuel



Lorsque vous avez terminé, enregistrez comme d’habitude pour juger du résultat.

8 juillet 2009 - 11:11am — Marine

Bienvenue sur le site **Colchique Conseil**,

le cabinet de conseil en ressources humaines qui à l’éthique au fond du cœur.

Vous trouverez sur notre site une présentation complète, des ses intervenants et de **ses valeurs**. Vous pourrez aussi découvrir nos publications ainsi que nos billets d’humeur. Bonne visite L’équipe
<http://www.colchique-conseil.fr/>

Figure 7–69

Le résultat formaté par l’éditeur TinyMCE

Le texte a bien pris en compte les nouveaux formats et le format d’entrée continue de fonctionner en transformant les URL en liens.

De la même manière, si vous voulez faire plaisir à vos visiteurs (ceci dit, tout le monde n’apprécie pas...), vous pouvez activer l’éditeur visuel pour les commentaires. Pour cela, nous allons associer cette fois TinyMCE au format Filtered HTML qui, rappelons-le, est le format par défaut pour tous les utilisateurs anonymes ou non, et sert donc pour les commentaires. Appliquez à cette association le même paramétrage que précédemment.

Déconnectez-vous ensuite pour juger du résultat en tant que visiteur anonyme.

Répondre

Votre nom : *

Adresse électronique : *

Le contenu de ce champ sera maintenu privé et ne sera pas affiché publiquement. Si vous avez un compte [gravatar](#), l'utilisez pour afficher votre avatar.

Page d'accueil :

Commentaire : *

Sympa ce nouvel éditeur de commentaires !!!

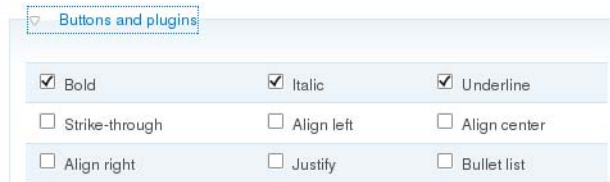
Chemin: p

[Disable rich-text](#)

Figure 7–70

TinyMCE utilisé dans les commentaires

Figure 7-71
Choix des boutons à afficher
pour l'édition des commentaires



Vous pouvez encore améliorer le paramétrage de cette association (section *Buttons and plugins*) en activant uniquement les boutons utiles pour un commentaire. Ceci est d'autant plus judicieux que la majorité des formats proposés par défaut risquent d'être filtrés par HTML filter.

Vous obtiendrez alors une barre de menus davantage en accord avec les circonstances :

Figure 7-72
TinyMCE en mode allégé
pour les commentaires



Comme vous le voyez, le couple Wysiwyg API-TinyMCE fonctionne à merveille. Vous pouvez cependant consacrer un peu de temps à installer les autres éditeurs pour déterminer au final celui qui vous convient le mieux.

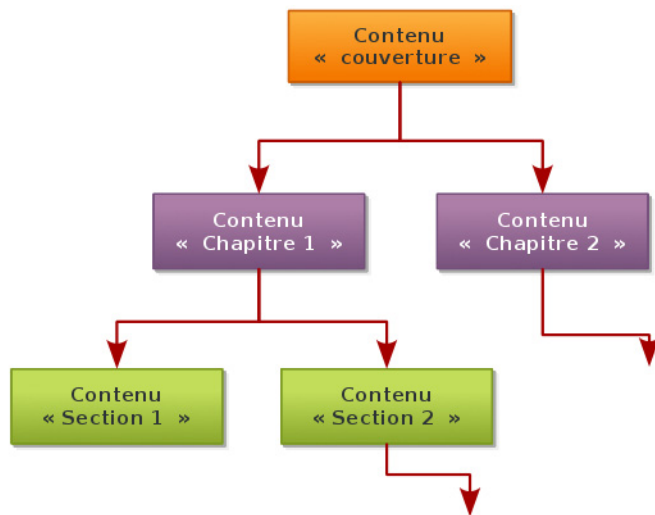
BON À SAVOIR Un éditeur visuel, non visuel...

Sachez à ce sujet que vous pouvez sans aucun problème associer un éditeur à un premier format et un autre éditeur à un second format. Vous pouvez ainsi garder par exemple la puissance de TinyMCE pour les contenus, et choisir la légèreté de MarkitUP pour les commentaires. MarkitUP est en effet un éditeur visuel très particulier, car il permet l'édition des balises et non pas des styles. Il fonctionne à ce titre comme un éditeur HTML. Il peut aussi être configuré pour reconnaître d'autres syntaxes comme phpBB (avec le filtre d'entrée adéquat, évidemment).

En résumé

- N'oubliez jamais que le type de contenu est un élément paramétrable. Lorsque vous vous demanderez où se règle tel ou tel aspect lié au contenu – par exemple après avoir installé un nouveau module –, ayez le réflexe d'aller voir si ce paramétrage ne se trouve pas dans les types de contenu.
- Les commentaires sont considérés comme des contenus ; par conséquent, leurs paramètres sont réglés par type de contenu et leur administration se trouve dans la section *Gestion des contenus*.
- Les formats d'entrée permettent la transformation de ce que vous avez saisi en un code XHTML. Vous pouvez créer de nouveaux formats ou étendre des formats existants à l'aide de filtres fournis par des modules.
- Si vous avez besoin d'aide, n'hésitez pas à poster des questions précises dans la section *Support général* de drupalfr.org.

chapitre 8



Créer des contenus avancés

Nous savons à présent gérer de bout en bout un simple article. Mais Drupal permet d'aller beaucoup plus loin dans la gestion des contenus, aussi étudierons-nous dans ce chapitre les méthodes pour définir des types de contenu plus complexes permettant d'opérer une séparation propre entre l'information et sa présentation. Nous aborderons également les techniques de création de listes de contenus, afin d'améliorer leur visibilité.

SOMMAIRE

- ▶ Hiérarchiser les contenus
- ▶ Gérer l'historique des contenus
- ▶ Créer des contenus complexes
- ▶ Régler finement les droits d'accès
- ▶ Fabriquer des listes de contenus personnalisées

Création d'une FAQ

Utiliser les livres

Dans notre storyboard, nous avons défini le besoin d'une FAQ (*Frequently Asked Questions* ou Foire aux questions). Pour répondre à ce besoin, nous utiliserons le module Book, qui fait partie de la distribution standard de Drupal, tout en étant désactivé par défaut.

Originellement conçu pour rédiger la documentation de Drupal, le module Book permet de lier un ensemble de contenus les uns aux autres en une arborescence formant un livre. Il permet ainsi d'assigner à n'importe quel contenu, peu importe son type, une référence à un contenu parent, et ce, jusqu'au contenu le plus élevé dans la hiérarchie qui devient en quelque sorte la couverture du livre.

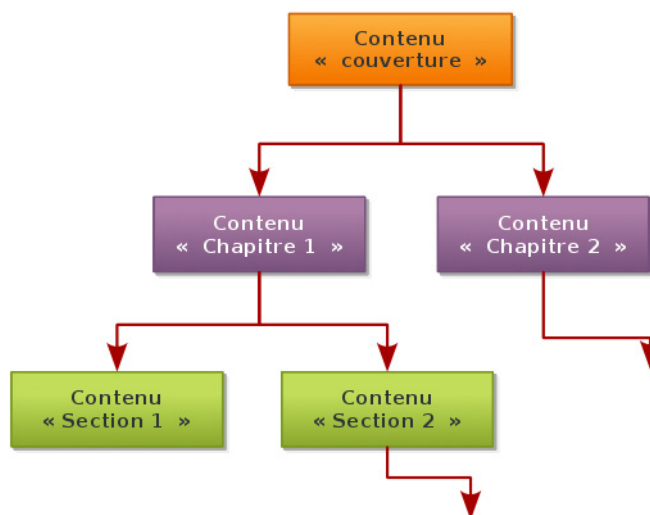


Figure 8–1

Le module Book permet de structurer les contenus en arborescences

Même si Book rend cette fonction accessible à tous les types de contenu existants, il ajoute cependant un nouveau type de contenu appelé Book page (Page de livre) qui est le type que l'on utilisera naturellement pour construire un livre même si une fois de plus, ceci n'a rien d'obligatoire.

Il faut bien comprendre que cette notion de « livre » est avant tout une métaphore. Vous pouvez utiliser Book pour bien d'autres choses qu'un livre. Il peut s'agir d'un dossier traitant d'un sujet donné et regroupant un ensemble d'articles se rapportant à ce sujet, ou encore pour notre exemple, d'une FAQ.

En effet, pour des raisons de lisibilité, l'ensemble des questions de notre FAQ ne vont pas résider sur une seule et même page. Nous allons créer une page [FAQ](#) qui en explique le rôle et qui donne accès à deux autres pages, une pour chaque grand domaine de compétence de la société. Et chaque page « domaine de compétence » va à son tour permettre d'afficher les pages de compétences spécifiques contenant chacune une liste de questions. Nous pouvons donc parfaitement utiliser le module Book pour ce genre d'usage en utilisant comme hiérarchie la structure des domaines de compétence de la société que nous avons définie dans le chapitre 3, « Bien définir son projet ».

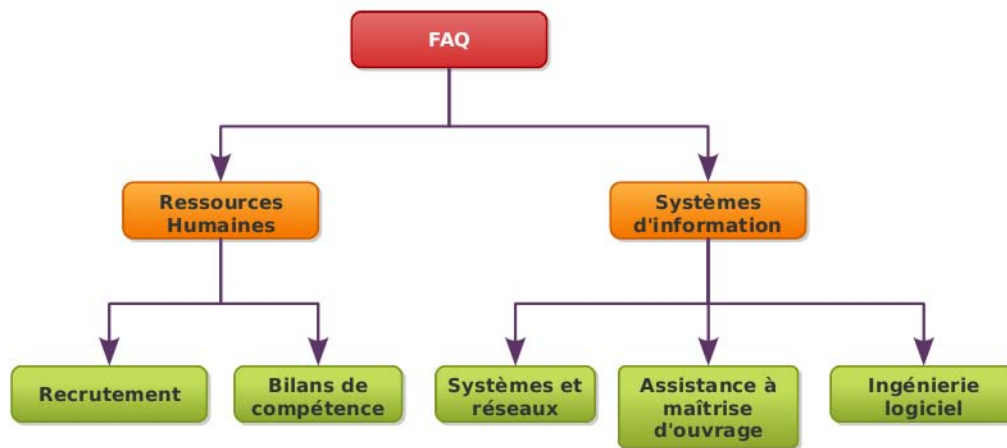


Figure 8-2
Arborescence cible de notre FAQ

Création de la FAQ

Nous allons commencer par créer une première page de livre qui sera en quelque sorte la couverture de notre FAQ. Pour cela, créez en tant qu'administrateur un nouveau contenu de type Page de livre (*Book page*), et commencez, comme d'habitude, par saisir son titre.

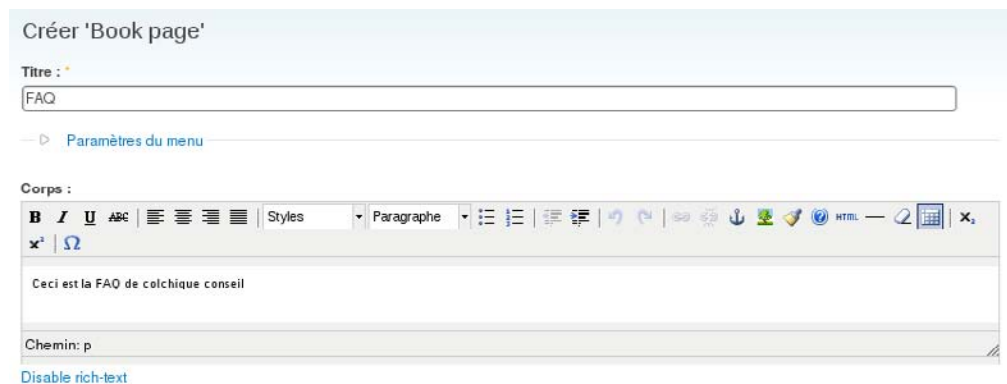


Figure 8-3
Création d'un contenu de type Page de livre

Les sections qui suivent sont les mêmes que pour les autres types de contenu à l'exception de *Structure du livre*. Dans cette section, nous sélectionnerons *Créer un nouveau livre* dans le champ *Livre*. Ceci fait, nous pouvons enregistrer notre contenu en prenant soin de vérifier que son statut *Publié* est activé et l'option *Promu en page d'accueil* désactivée.

Ceci fait, nous obtenons la version complète du contenu qui diffère quelque peu de ce que nous avons pu voir jusqu'ici.

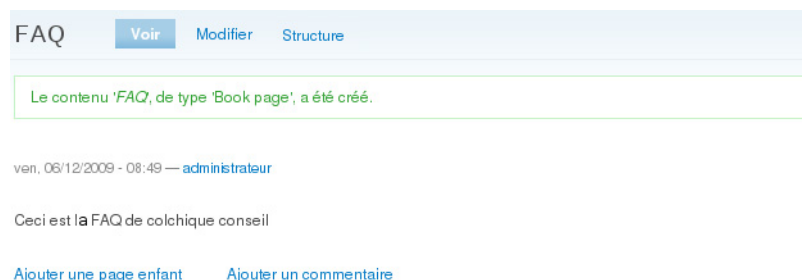


Figure 8-4
Depuis la page d'un livre peuvent être créées les pages enfant.

BON À SAVOIR Le module Book hiérarchise tous les types de contenu

Le module Book va plus loin que la fourniture d'un contenu de type Page de livre. Il permet de hiérarchiser n'importe quel autre type de contenu. Si vous vous rendez dans *Gestion du contenu* > *Books*, puis dans l'onglet *Paramètres*, vous avez la possibilité de cocher les types de contenu que vous désirez hiérarchiser en tant que livre, et en tant que page de livre.

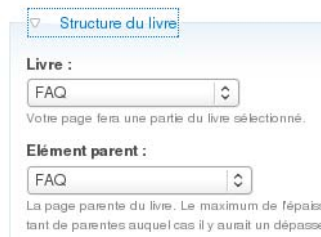


Figure 8-5
Création d'un nouveau « livre »

Comme vous le voyez, un nouveau lien *Ajouter une page enfant* vous permet ici de créer un nouveau contenu de type Page de livre, et qui aura pour parent notre page FAQ. Notez aussi la présence d'un nouvel onglet *Structure* qui n'est qu'un accès rapide permettant de définir à quel livre ce contenu appartient et quel est son contenu parent.

Création des sections de FAQ

Pour l'heure, cliquez sur *Ajouter une page enfant* pour créer le chapitre « Ressources Humaines ». La procédure est la même que pour n'importe quel autre contenu : un titre « Ressources Humaines », et dans le corps, tout ce qui vous semble pertinent pour décrire ce chapitre. Dans la section *Structure du livre*, vous noterez que *FAQ* a été automatiquement sélectionnée en tant que page parent de notre contenu, de par l'utilisation du lien *Ajouter une page enfant*. Vous constaterez également l'arrivée d'un nouveau champ *Livre* qui a la même valeur qu'*Élément parent*. En effet, lorsque nous avons choisi tout à l'heure l'option *Créer un nouveau livre*, nous avons de facto créé un livre du même nom que sa couverture.

Pour que l'accès à notre FAQ soit plus simple, nous allons finir par la définition d'un alias sur le chemin du nœud, nommé simplement « faq ». Les questions seront alors accessibles par l'URL <http://www.colchique-conseil.fr/faq>.

Vous pouvez à présent enregistrer ce premier contenu et poursuivre la création des autres chapitres (*Système d'information*) et sections (*Recrutement*, *Systèmes et réseaux*, etc.). Dans chacun des corps (chapitres ou sections), nous allons pouvoir saisir des questions et des réponses :

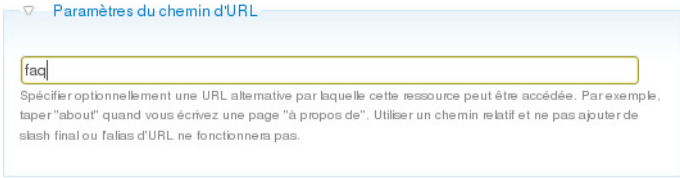


Figure 8–6
Définition d'un alias sur la couverture du FAQ



Figure 8–7
Création d'une page enfant

Organisation des pages de livre

Nous avons limité la hiérarchie des contenus à trois niveaux, mais sachez qu'il n'y a sur ce point aucune limitation particulière.

Lorsque vous avez terminé, allez dans *Gestion du contenu* > *Livres*. Nous retrouvons ici notre livre dans une liste.

Livre	Opérations
FAQ	modifier l'ordre et les titres

Figure 8–8
La liste des livres

Cliquez maintenant sur *modifier l'ordre et les titres*.

Titre	Opérations
⊕ Ressources Humaines	voir modifier supprimer
⊕ Bilans de compétence	voir modifier supprimer
⊕ Recrutement	voir modifier supprimer
⊕ Système d'information	voir modifier supprimer
⊕ Système et réseaux	voir modifier supprimer
⊕ Assistance à maîtrise d'ouvrage	voir modifier supprimer
⊕ Ingénierie logiciel	voir modifier supprimer

Figure 8–9
L'organisation initiale de notre livre

ERGONOMIE **Hiérarchisation**

Nous avons déjà utilisé le système de Drupal permettant de réordonner dynamiquement les lignes d'un tableau. Lorsqu'il s'agit, comme pour le module Book, d'une hiérarchie, il est aussi possible de la modifier en faisant glisser la ligne horizontalement.

Nous constatons que, dans notre hâte, nous avons commis pas mal d'erreurs dans la hiérarchie, le tout émaillé de quelques fautes de frappe. Ce n'est pas grave, car nous pouvons tout modifier. Ce panneau nous permet en effet de réordonner nos pages et changer la hiérarchie (en faisant glisser de gauche à droite et de haut en bas) ainsi que renommer nos titres. Lorsque tout est rentré dans l'ordre, vous pouvez enregistrer.

Titre		Opérations		
+	Ressources Humaines	voir	modifier	supprimer
+	Recrutement	voir	modifier	supprimer
+	Bilans de compétence	voir	modifier	supprimer
+	Système d'information	voir	modifier	supprimer
+	Système et réseaux	voir	modifier	supprimer
+	Assistance à maîtrise d'ouvrage	voir	modifier	supprimer
+	Ingénierie logiciel	voir	modifier	supprimer

Figure 8-10

Le même, un peu mieux agencé

Notre FAQ est maintenant correctement organisée. Pour y accéder, retournons sur la page de gestion des livres en cliquant sur *Books* sous le titre de la page, puis sur *FAQ*.

FAQ

Voir

Modifier

Structure

ven, 06/12/2009 - 08:49 — administrateur

Ceci est le FAQ de colchique conseil

▷ Ressources Humaines

▷ Système d'information

Ressources Humaines >

Figure 8-11

Présentation finale
de la couverture du livre FAQ

Comme vous le voyez, le module Book intègre un système de navigation qui nous permet de circuler d'un chapitre à l'autre et d'une page à l'autre.

Recrutement

Voir

Modifier

Structure

ven, 06/12/2009 - 09:00 — administrateur

Qu'est-ce qu'un recrutement ?

Un recrutement est

◀ Ressources Humaines

haut

Bilans de compétence ▶

Figure 8-12

La navigation est possible
dans plusieurs directions

Nous verrons un peu plus loin dans cet ouvrage comment rendre cette navigation encore plus intuitive.

Gestion des révisions

Consultation des révisions

Comme nous l'avons vu dans sa description générale, Drupal permet d'archiver les versions successives d'un contenu – appelées révisions – et vous permet ainsi de revenir en arrière si une modification ne vous convient plus.

Lorsque nous avons paramétré le type de contenu Article, nous avons indiqué que nous désirions archiver une révision pour chaque modification du contenu. Ayant maintenant fait un certain nombre de modifications sur notre premier article, il est temps de voir ce que donnent ces fameuses révisions.

Pour cela, redevenons Marine, et dirigeons-nous vers la version complète de l'article, puis cliquons sur le lien *Révisions*.

Versions antérieures de « Bienvenue »

[Voir](#) [Modifier](#) [Révisions](#)

Les révisions permettent de suivre les différences entre plusieurs versions d'une contribution.

Révision	Opérations
08/07/2009 - 13:42 par Marine Mise en forme avec l'éditeur visuel	version en cours
08/07/2009 - 12:45 par Marine Changement du format d'entrée	
08/07/2009 - 12:43 par Marine Transformation en article de bienvenue...	
08/07/2009 - 11:55 par administrateur Ajout d'un alias sur l'article	
08/07/2009 - 11:11 par Marine On a tous commencé un jour !	

MODULES MAGIQUES

Un module spécialisé pour les FAQ

L'utilisation de Book pour décrire une FAQ est avant tout un prétexte pour nous permettre de comprendre le fonctionnement de ce module important. Pour une FAQ réelle, Il existe un module spécialement dédié à cet usage qui se nomme tout simplement Faq.

► <http://drupal.org/project/faq>

Figure 8-13
L'historique des révisions du contenu

Figure 8–14
Une révision supplémentaire

Nous découvrons ici que nous avons nourri l'historique des révisions sans même le savoir. Nous voyons aussi tout l'intérêt de systématiquement saisir un message d'explication dans le journal. Nous savons ainsi parfaitement à quelle opération correspond chaque version. Pour l'exemple, nous allons modifier à nouveau ce contenu pour créer une nouvelle révision en mettant comme message dans le journal « Correction de quelques coquilles ».

Révision	Opérations
08/07/2009 - 13:54 par Marine Correction de quelques coquilles	version en cours
08/07/2009 - 13:42 par Marine Mise en forme avec l'éditeur visuel	

Administration des révisions

Redevenons maintenant Administrateur pour voir comment les révisions sur cet article apparaissent pour ce rôle.

Révision	Opérations
08/07/2009 - 13:54 par Marine Correction de quelques coquilles	version en cours
08/07/2009 - 13:42 par Marine Mise en forme avec l'éditeur visuel	revenir supprimer
08/07/2009 - 12:45 par Marine Changement du format d'entrée	revenir supprimer
08/07/2009 - 12:43 par Marine Transformation en article de bienvenue...	revenir supprimer
08/07/2009 - 11:55 par administrateur Ajout d'un alias sur l'article	revenir supprimer
08/07/2009 - 11:11 par Marine On a tous commencé un jour !	revenir supprimer

Figure 8–15
L'historique des révisions pour l'administrateur

Comme vous le voyez, l'écran est le même à ceci près que l'administrateur dispose de contrôles supplémentaires. Il peut *supprimer* une révision, mais aussi, *revenir* sur une révision. Pour comprendre ce que cela implique, cliquons sur le lien *revenir* sur la version qui précède la dernière révision de Marine.

Révision	Opérations
08/07/2009 - 13:59 par administrateur Copier la version du 8 juillet 2009 - 1:42pm.	version en cours
08/07/2009 - 13:54 par Marine Correction de quelques coquilles	revenir supprimer
08/07/2009 - 13:42 par Marine Mise en forme avec l'éditeur visuel	revenir supprimer

Figure 8-16

L'administrateur peut revenir en arrière sur une révision donnée.

Le système de révisions de Drupal est ainsi fait que le retour en arrière consiste en réalité à recopier le contenu de cette révision et à l'utiliser pour en créer une nouvelle. Nous sommes ainsi assurés de ne jamais rien perdre.

Nous pouvons cependant décider de supprimer une révision, par exemple l'avant-dernière en date.

Révision	Opérations
08/07/2009 - 13:59 par administrateur Copier la version du 8 juillet 2009 - 1:42pm.	version en cours
08/07/2009 - 13:42 par Marine Mise en forme avec l'éditeur visuel	revenir supprimer

Figure 8-17

Révision supprimée

Ce système est en quelque sorte la mémoire de vos contenus, doublée d'une mini machine à remonter le temps. Ceci étant dit, ce confort a un prix et chaque révision prend chaque fois un peu plus d'espace de stockage sur la base de données. À vous de déterminer si cette sécurité est utile, sachant que la réponse est très souvent oui.

MODULES MAGIQUES **Afficher visuellement les différences entre révisions**

Le module contribution Diff est le compagnon idéal des révisions. Il rajoute des options à la liste des révisions pour vous permettre d'afficher les différences entre deux révisions.

► <http://drupal.org/project/diff>

Classer les contenus

Qu'est-ce que la taxonomie ?

Comme nous l'avons déjà vu dans les premiers chapitres, pour remonter à l'origine du terme, la taxonomie est la science de la classification. Son objectif est de regrouper ce qu'elle cherche à classer en ensembles appelés « taxons ». Ces taxons sont eux-mêmes regroupés en taxons parent jusqu'à former une taxonomie.

Pour prendre un exemple, une taxonomie des animaux ressemblerait, de manière aussi simplifiée qu'incomplète, à ceci :

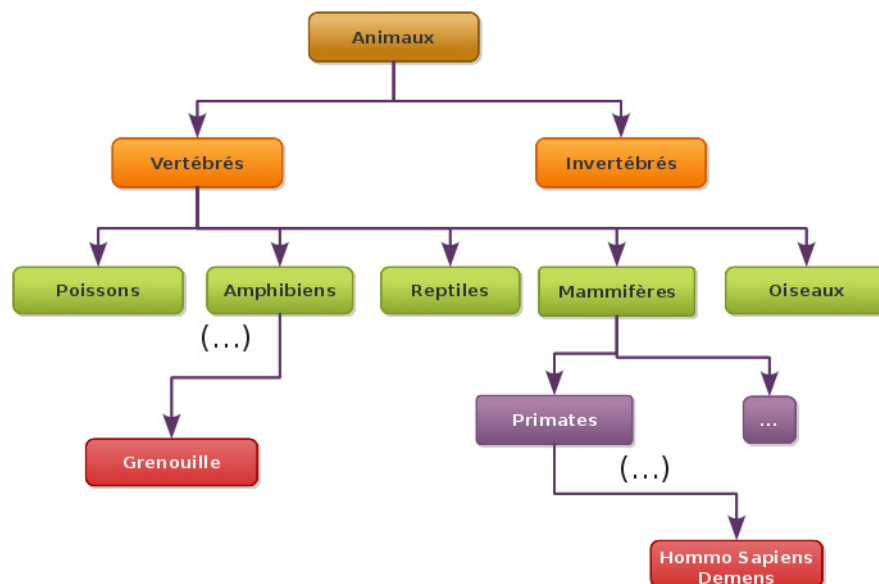


Figure 8-18
Une taxonomie appliquée aux animaux

À quoi sert la taxonomie ?

Une taxonomie va tout simplement nous permettre d'assigner des taxons à chacun de nos contenus de manière à pouvoir les retrouver plus facilement. En langage plus clair, cela va nous servir à déterminer des catégories où ranger nos contenus.

La taxonomie et Drupal

Drupal vous permet de créer des classifications reflétant vos besoins spécifiques pour ensuite y rattacher des contenus. Si nous reprenons l'exemple ci-dessus, un article traitant des grenouilles serait rattaché au taxon Amphibien. Il serait alors possible, et c'est là tout l'intérêt, de rechercher tous les contenus qui parlent d'amphibiens, mais aussi de vertébrés, et à chaque fois d'obtenir une liste de contenus associé à ce sujet.

De la manière la plus générale possible, le système de classification de Drupal se définit selon les règles suivantes.

- Un *terme* (qui serait l'équivalent du *taxon*), est un mot (ex. « reptile ») auquel il est possible d'associer une description et éventuellement une série de synonymes et de termes connexes.

- Un *vocabulaire* (qui serait l'équivalent de la *taxonomie*) est un ensemble de termes. Cet ensemble dispose d'un nom (ex. « Les animaux ») et est complété là aussi par une description.
- Un terme appartient à un et un seul vocabulaire.
- Un terme peut être relié à un ou plusieurs autres termes au sein d'un même vocabulaire.
- Un contenu peut être relié à un ou plusieurs termes provenant d'un ou plusieurs vocabulaires.

L'utilisation de *taxonomie* pour définir ce mode de classement risque de choquer les plus rigoureux. En effet, en appliquant ces règles, nous sommes ici dans une vision plus large qu'une simple taxonomie, car nous pouvons définir des relations terme parent-terme enfant sans ordre particulier et surtout sans hiérarchie particulière. Cependant, dans la grande majorité des cas, ce seront bien des hiérarchies de termes que nous définirons ici.

BON À SAVOIR La taxonomie et les graphes

Le système de classement de Drupal se rapproche plus de la notion mathématique de graphes (http://fr.wikipedia.org/wiki/Théorie_des_graphes). Il est ainsi possible de définir plusieurs parents pour un même terme, et ce, sans orientation particulière.

La notion multiparent en elle-même peut présenter certains avantages comme celui de permettre la fusion de plusieurs vocabulaires. En effet, si nous avons un vocabulaire « les plantes » et un autre « recette de cuisines », nous aurions deux termes « basilic », l'un en tant que plante, l'autre en tant qu'ingrédient. Plutôt que rattacher deux termes « basilic » à un contenu, nous pouvons préférer créer un super-vocabulaire avec deux branches « plantes » et « recettes ». Le terme « basilic » pourra donc être relié à ces deux parents. Ainsi en cherchant « basilic », vous aurez à la fois les articles se rapportant à la plante, et ceux qui l'utilisent en tant qu'ingrédient.

Ceci étant dit, dès lors que vous liez un terme à deux parents, Drupal bascule le vocabulaire dans un mode particulier qui vous interdira, et c'est bien logique, de réorganiser les termes entre eux de manière graphique.

Ajouter un nouveau vocabulaire

Maintenant que nous voyons mieux comment fonctionne le système de classement, nous allons l'exploiter pour décrire la hiérarchie de nos domaines de compétence telle que décrite dans le chapitre « Bien définir votre projet » (comme pour la FAQ).

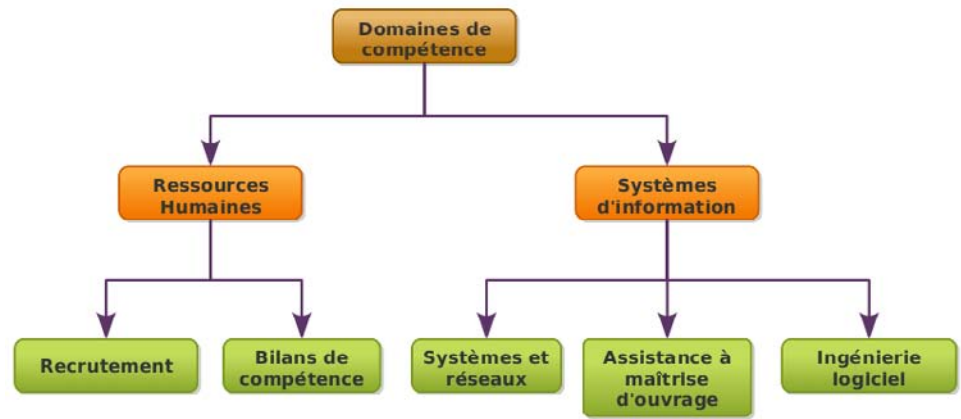


Figure 8–19
La taxonomie des domaines de compétence de Colchique conseil

Tout commence par l'ajout du vocabulaire que nous allons appeler logiquement « Domaines de compétence ». Notez que vous n'êtes en rien limité, tant en nombre de vocabulaires que de niveaux de termes qu'un vocabulaire va pouvoir contenir.

Pour ajouter un vocabulaire, il nous faut redevenir Administrateur et nous diriger vers [Administration](#) > [Gestion du contenu](#) > [Taxonomie](#). Nous avons sur cette page une liste pour l'instant vide que nous allons compléter de ce pas en cliquant sur le lien [Ajouter un vocabulaire](#).

Une fois le nom donné, vous pouvez aussi rédiger une description et un court texte d'aide qui vous seront utiles si vous êtes amené à gérer de nombreux vocabulaires.

Nom du vocabulaire : *

Domaines de Compétence

Le nom de ce vocabulaire, i.e. "Étiquettes".

Description :

Les domaines de compétence de Colchique

Description du vocabulaire : peut être utilisé par d'autres modules.

Figure 8–20
Création d'un nouveau vocabulaire

Types de contenu

Types de contenu :

☒ Article

☐ Page

Sélectionner les types de contenus à classer grâce à ce vocabulaire.

Figure 8–21
Association du vocabulaire à des types de contenu

Drupal permet dans la section suivante de définir les types de contenu pour lesquels ce vocabulaire pourra être utilisé. Nous allons sélectionner *Article*.

Paramètres

☐ **Étiquettes**
Les termes sont créés par les utilisateurs lors de la création de contenus en saisissant une liste de mots séparés par des virgules.

☐ **Choix multiple**
Permet d'affecter aux contenus plus d'un terme de ce vocabulaire (toujours vrai pour les étiquettes).

☒ **Obligatoire**
Au moins un terme de ce vocabulaire doit être sélectionné lors de la publication d'un contenu.

Poids :
0 ▼
Les vocabulaires sont affichés par ordre croissant de poids.

Figure 8–22
Mode de fonctionnement du vocabulaire de contenu

La section suivante décrit le mode de fonctionnement du vocabulaire. Tout d'abord, il permet de fixer son caractère *Obligatoire*. Si cette case est cochée, un contenu utilisant ce vocabulaire ne pourra pas être validé tant qu'au moins un terme n'aura pas été sélectionné.

L'option *Choix multiple* permet au rédacteur du contenu de sélectionner plusieurs termes de ce vocabulaire pour un même nœud.

L'option *Étiquettes* est un mode particulier du fonctionnement que nous venons de décrire. En effet, dans le mode standard, il est nécessaire que les termes soient saisis avant qu'un contributeur puisse les rattacher à son contenu. Lorsque vous sélectionnez l'option *Étiquettes*, le contributeur pourra lui-même donner la liste des termes qui se rapportent à son contenu. Cette option permet de se libérer de la création préalable d'une arborescence de termes, mais présente deux inconvénients :

- les termes que saisira le contributeur ne pourront être rattachés qu'à la racine, pas question de créer des arborescences complexes de termes ;
- une faute de frappe comme « Ressources humaines » au lieu de « Ressources Humaines » ne sera pas détectée par Drupal ; il considérera que les deux entrées sont différentes et créera les deux termes.

Le mode *Étiquettes* demande donc à l'administrateur de vérifier régulièrement la liste des termes créés pour éventuellement les corriger, voire les réorganiser.

Pour notre exemple, nous allons choisir les modes *Obligatoire* et *Choix multiple*. Une dernière option a été passée sous silence : le *Poids* du vocabulaire. En effet, lorsque plusieurs vocabulaires sont activés pour un même type de nœud, cette option permet de déterminer leur ordre d'affichage. Cette option est importante d'un point de vue ergonomique,

mais, comme pour les attachements ou l'ordre des filtres d'entrée, la liste des vocabulaires permet d'opérer ce même réordonnement de manière plus visuelle et donc plus simple.

Nous n'avons maintenant plus qu'à cliquer sur [Enregistrer](#) pour que le nouveau vocabulaire soit effectivement créé.

Ajouter des termes

Une fois le vocabulaire créé, vous êtes renvoyé à la liste des vocabulaires. Il est maintenant temps d'y ajouter des termes. Pour cela, cliquez dans la liste sur le lien [Ajouter des termes](#). Nous arrivons alors sur un formulaire assez simple composé en premier lieu du *Nom du terme* et de sa *Description*. Saisissez *Ressources Humaines*, puis la description de votre choix et enfin, cliquez sur [Enregistrer](#).

Figure 8–23

Ajout du terme « Ressources Humaines »

Le formulaire s'affiche à nouveau avec un message vous indiquant que votre terme a bien été créé. Re commençons, mais cette fois-ci, avec le terme « Systèmes d'information ».

Une fois le terme ajouté, nous sommes à nouveau prêt à saisir un troisième terme. Il s'agira de « Recrutement » qu'il conviendra de relier au terme « Ressources Humaines ». Pour ce faire, saisissez d'abord le nom du terme [Recrutement](#), puis déployez la section [Options avancées](#).

Figure 8–24

Association d'un terme enfant à un terme parent

Dans ce panneau, vous avez la possibilité de sélectionner le terme *Parents* de notre nouveau terme. Sélectionnez *Ressources Humaines*. Vous constatez qu'il est aussi possible de définir des termes connexes (*Notion associée*) représentés par des termes existants, ainsi qu'une liste libre de synonymes (un synonyme par ligne pouvant être une phrase).

Vous avez aussi la possibilité de définir un poids permettant de changer l'ordre d'affichage des termes, mais il sera plus simple de faire cela un peu plus tard. En effet, comme pour la structure d'un livre, Drupal dispose d'un éditeur visuel de termes qui nous permettra non seulement de changer l'ordre, mais aussi de déplacer les termes d'un parent à l'autre. Pour mieux illustrer cette fonctionnalité, saisissez maintenant tous les termes encore manquants, mais cette fois sans définir le parent : saisissez juste le nom et description et enregistrez.

Lorsque vous avez terminé, vous pouvez juger du résultat en cliquant sur l'onglet *Liste*.

Nom	Opérations
+ Assistance à maîtrise d'ouvrage	éditer
+ Bilans de compétence	éditer
+ Ingénierie logiciel	éditer
+ Ressources Humaines	éditer
+ Recrutement	éditer
+ Systèmes d'information	éditer
+ Systèmes et réseaux	éditer

Figure 8–25
Notre vocabulaire encore peu organisé

Cette page est l'éditeur visuel dont nous parlions plus haut. Vous pouvez ici modifier chacun des termes (lien *éditer*) mais surtout organiser la position d'un terme en le faisant glisser de haut en bas, et en le déplaçant pour le rattacher à un parent différent (ne pas oublier d'enregistrer une fois le déplacement effectué). Lorsque cette réorganisation est terminée, nous disposons d'une taxonomie fonctionnelle :

Nom	Opérations
+ Ressources Humaines	éditer
+ Bilans de compétence *	éditer
+ Recrutement	éditer
+ Systèmes d'information	éditer
+ Systèmes et réseaux *	éditer
+ Assistance à maîtrise d'ouvrage *	éditer
+ Ingénierie logiciel *	éditer

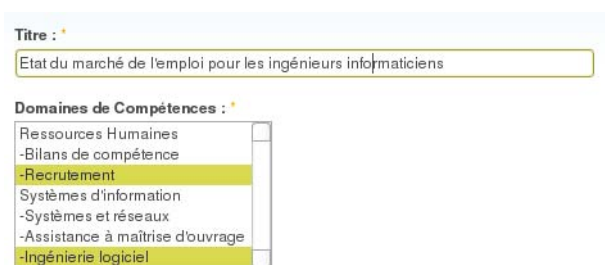
Figure 8–26
Le même vocabulaire avec tous les termes correctement hiérarchisés

Associer des contenus à des termes

Nous avons associé notre vocabulaire au type de contenu Article, il nous est donc possible maintenant d'associer un terme à n'importe lequel des articles que nous allons créer ou que nous avons déjà créés. Pour

l'exemple, le plus simple est de créer un nouvel article qui traitera de recrutement dans le domaine de l'ingénierie logiciel.

Une fois sur la page de création de l'article, vous pouvez saisir le titre, par exemple « État du marché de l'emploi pour les ingénieurs informaticiens ». Juste en dessous du titre, vous découvrez un nouveau champ *Domaines de compétence*. Il s'agit de notre taxonomie qui a été automatiquement associée aux contenus de type Article. Nous avons créé un vocabulaire à choix multiple et obligatoire, nous avons donc là un zone dans laquelle nous devons sélectionner au moins un terme. Pour choisir plusieurs termes, maintenez la touche *Ctrl* enfoncée lorsque vous cliquez sur les termes.

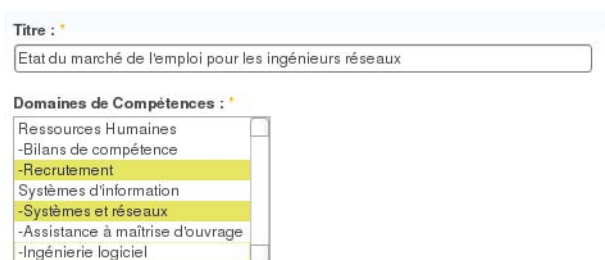


The screenshot shows a form with a title field containing 'Etat du marché de l'emploi pour les ingénieurs informaticiens'. Below it is a dropdown menu labeled 'Domaines de Compétences : *'. The menu is open, showing a list of terms: 'Ressources Humaines', '-Bilans de compétence', '-Recrutement', 'Systèmes d'information', '-Systèmes et réseaux', '-Assistance à maîtrise d'ouvrage', and '-Ingénierie logiciel'. The term '-Recrutement' is currently selected and highlighted in yellow.

Figure 8–27
Association du contenu aux termes
Recrutement et Ingénierie logiciel

Vous pouvez maintenant saisir le corps de votre article puis enregistrer. L'administrateur n'a plus qu'à publier votre article.

Renouvelez l'opération avec un second contenu relié cette fois aux termes *Recrutement* et *Systèmes et réseaux* que nous ferons là aussi publier par l'administrateur.



The screenshot shows a form with a title field containing 'Etat du marché de l'emploi pour les ingénieurs réseaux'. Below it is a dropdown menu labeled 'Domaines de Compétences : *'. The menu is open, showing the same list of terms as in Figure 8-27. In this instance, two terms are selected and highlighted in yellow: '-Recrutement' and '-Systèmes et réseaux'.

Figure 8–28
Association du contenu aux termes
Recrutement et Systèmes et réseaux

Recherches par terme

De nos deux contenus publiés, observons le premier.

Comme vous le voyez, en bas à droite, nous avons maintenant deux liens reflétant les deux termes que nous avons associés à cet article. Cliquez maintenant sur le lien *Recrutement*.



Figure 8-29 Apparition des termes dans la version résumée des contenus



Figure 8-30 Résultat de la recherche des contenus rattachés au terme Recrutement

Drupal affiche alors une page de recherche contenant la liste de tous les articles liés au terme Recrutement. Nous pouvons de même cliquer sur *Ingénierie logiciel* pour ne voir apparaître cette fois qu'un seul contenu qui est le seul lié à ce terme.

Le visiteur peut ainsi, de proche en proche, naviguer facilement à travers les termes de notre taxonomie et afficher la liste des contenus qui traitent de tel ou tel sujet.

Pour aller plus loin dans ce que permet Drupal dans la recherche de termes, redevenez Administrateur et retournez sur la liste des vocabulaires. Ceci fait, cliquez sur le lien *lister les termes* du vocabulaire Domaines de compétence.

Vous êtes alors face à une liste connue représentant notre vocabulaire. Vous pouvez ici constater que chaque terme est un lien. Ainsi, si vous cliquez sur *Recrutement*, vous allez retrouver la même page de résultat que précédemment.

Revenez à la liste, et cette fois cliquez sur *Ressources Humaines*, vous obtenez alors une liste vide.

Figure 8-31
Recherche infructueuse sur les contenus
rattachés au terme Ressources Humaines

Ressources Humaines

Il n'y a actuellement aucune contribution dans cette catégorie.

Ce comportement est pour le moins irritant : si l'on s'est donné la peine de hiérarchiser les termes, c'est justement pour pouvoir élargir les recherches de la sorte.

Mais avant de s'inquiéter, nous allons disséquer l'adresse (URL) qui mène à cette recherche infructueuse, <http://www.colchique-conseil.fr/taxonomy/term/1>. Cette adresse doit être lue comme « voici la liste de l'ensemble des nœuds dont le terme de taxonomie a pour identifiant 1 ». En effet, chaque terme a un identifiant unique, quel que soit le vocabulaire auquel il appartient. Pour vous, l'identifiant peut être différent, il s'agit donc d'adapter les exemples qui suivent en conséquence. Notez seulement que vous pouvez toujours le retrouver en retournant dans la liste des termes, et en cliquant sur le terme dont vous cherchez l'identifiant. Le dernier chiffre affiché dans l'URL est l'identifiant de votre terme.

Vous allez maintenant modifier l'URL dans la barre d'adresse de votre navigateur pour ajouter /2 à la fin et obtenir ainsi, <http://www.colchique-conseil.fr/taxonomy/term/1/2>, puis validez. Cette fois la liste des résultats est plus intéressante :

Ressources Humaines

Etat du marché de l'emploi pour les Administrateurs réseaux

dim, 06/14/2009 - 10:30 — [Marine](#)

Les administrateurs réseaux connaissent-ils la crise ?

[Ajouter un commentaire](#)
[Recrutement](#)
[Systèmes et réseaux](#)

État du marché de l'emploi pour les ingénieurs informaticiens

dim, 06/14/2009 - 10:23 — [Marine](#)

La crise touche t-elle les informaticiens ?

[Ajouter un commentaire](#)
[Recrutement](#)
[Ingénierie logiciel](#)

Figure 8-32 Recherche de profondeur 2 sur les contenus rattachés au terme Ressources Humaines

La première recherche n'a d'abord rien donné, car pour des raisons de performances, Drupal limite par défaut ses recherches à seulement un niveau, c'est-à-dire sur le terme lui-même. En ajoutant /2, vous avez augmenté la profondeur de la recherche à deux niveaux, englobant ainsi

les contenus rattachés à Ressources Humaines, mais aussi ceux rattachés aux termes enfant de Ressources Humaines, dont Recrutement.

De la même manière, dès lors que vous connaissez les identifiants de vos termes, il est possible de composer des URL de recherche complexes comme :

- <http://monsite.fr/taxonomy/term/A,B,C> recherchera les contenus rattachés aux termes ayant pour identifiants respectifs A, B et C. Vous pouvez rajouter autant d'identifiants que nécessaire à votre recherche provenant d'un ou plusieurs vocabulaires. Vous pouvez rajouter à la fin un /N, où N représente la profondeur de recherche désiré ;
- <http://monsite.fr/taxonomy/term/A+B+C> a le même rôle que la syntaxe précédente, mais cette fois ne recherche que la présence d'au moins l'un des termes. Là aussi il est possible d'ajouter une profondeur.

MODULES MAGIQUES Naviguer dans la taxonomie

Cette manière de construire des URL pour faire une recherche peut sembler un peu barbare et en tout cas peu intuitive. Mais l'objectif de cette fonctionnalité est de fournir à la personne qui va réaliser le site, les moyens de créer des liens hypertextes adaptés aux besoins ergonomiques. Cela pourra par exemple être une série de rubriques avec des termes associés aux documents de chacune d'elles.

Il vous est cependant possible d'ajouter une manière simple de naviguer dans les vocabulaires sans passer par les URL, en utilisant le module Taxonomy navigator. Ce module fournit un bloc que nous apprendrons à mettre en place au chapitre suivant.



Figure 8-33 Bloc du module Taxonomy navigator

Dans le même esprit, vous pourrez aussi utiliser la taxonomie pour générer des nuages d'étiquettes. Pour cela, vous pouvez utiliser le module Tagadelic.



Figure 8-34 Nuage d'étiquettes grâce au module Tagadelic

Ces deux modules fournissent ce que l'on appelle des « blocs », une notion que nous aborderons dans le chapitre suivant, « Structurer les pages ».

Ajouter de nouveaux types de contenu

Avant de voir pourquoi il peut être intéressant d'ajouter de nouveaux types de contenu, revenons sur nos définitions.

Un *type de contenu* regroupe un ensemble de champs qui permettent de caractériser les contenus appartenant à ce type.

Dans de nombreux cas, les deux types de contenus fournis en standard par Drupal sont loin d'être suffisants pour tous les usages. La nécessité d'un nouveau type de contenu peut en effet apparaître si :

- vous avez besoin d'un type de contenu ayant un nom particulier (ex. billet, nouvelles, dépêche, etc.) avec en outre des réglages spécifiques (commentaires, publication, etc.) ;
- vous avez besoin d'un type de contenu doté de champs supplémentaires non proposés en standard.

Ajout de types simples

Cette première possibilité offerte par Drupal vous permet de créer très facilement de nouveaux types de contenu qui seront totalement paramétrables, mais l'ajout de nouveaux champs sera impossible. Cette approche correspond donc bien au premier cas de figure où l'on cherche un nom de contenu différent avec un paramétrage spécifique.

C'est par exemple le cas du type de contenu Billet que nous avons évoqué dans le chapitre « Bien définir son projet » : rien ne le distingue réellement d'un Article, mis à part un usage différent.

	Billet
→	Titre
→	Auteur
→	Date de création
→	Domaines de compétence
→	Corps

Figure 8-35
La définition du type Billet

Pour créer ce nouveau type, prenez le rôle Administrateur, puis dirigez-vous vers [Administrer](#) > [Gestion de contenu](#) > [Types de contenu](#), puis cliquez sur l'onglet [Ajouter](#).

Vous retrouvez alors un formulaire connu de configuration de contenu à la différence près que les champs *Nom*, *Type* et *Description* sont ici vierges. Nous allons donc saisir les informations suivantes :

Identification

Nom : *

Billet

Le nom affiché de ce type de contenu. Ce texte sera affiché dans la liste sur la page *créer un contenu*. Ce nom doit commencer par une lettre majuscule et ne doit contenir que des lettres, chiffres, et espaces. Ce nom doit être unique.

Type : *

billet

Le nom interne de ce type de contenu. Ce texte sera utilisé pour construire l'URL de la page *créer un contenu* pour ce type de contenu. Ce nom ne doit contenir que des lettres minuscules, des chiffres, et des caractères de soulignement. Les caractères de soulignement seront convertis en trait d'union lors de la construction des URL de la page *Créer un contenu*. Ce nom doit être unique.

Description :

Utilisez ce type de contenu pour écrire un billet, c'est à dire un contenu d'une page maximum pouvant être lu en moins de 15 minutes. Si votre contenu est plus long que cela, il s'agira d'utiliser le type de contenu "Article"

Une brève description de ce type de contenu. Ce texte sera affiché dans la liste qui se trouve sur la page *créer un contenu*.

Figure 8–36
Informations d'identification
du type de contenu Billet

Dans la section *Procédure de publication*, nous allons cocher les éléments suivants de sorte que, comme pour le type de contenu Article, une révision soit créée automatiquement. En revanche, un Billet ayant moins d'importance qu'un Article, nous allons autoriser la publication immédiate en laissant activés les champs *Publié* et *Promu en page d'accueil*.

Options par défaut :

☒ Publié

☒ Promu en page d'accueil

☐ Epinglé en haut des listes

☒ Créer une révision

Figure 8–37
Définition des options de publication par défaut

Sur le reste des options, nous adopterons pour ce type de contenu un paramétrage identique à celui de l'Article, notamment pour la gestion des commentaires. Une fois ces réglages terminés, il ne reste plus qu'à enregistrer le nouveau contenu.

Pour que ce dernier devienne actif pour le rôle Contributeur, il est nécessaire d'aller dans la gestion des droits d'accès pour activer les permissions concernant ce nouveau type de contenu :

- *Create billet content* (notez que certains termes de l'interface ne sont pas traduits, nous aborderons ce point un peu plus loin) ;
- *Edit own billet content* ;
- *Delete billet content*.

Figure 8–38
Association du vocabulaire Domaines de compétence au type de contenu Billet



Enfin, il nous reste à ajouter le vocabulaire Domaines de compétence à ce type de contenu. Pour ce faire, rendez-vous sur l'administration de la taxonomie, modifiez le vocabulaire Domaines de compétence, et cochez le type de contenu que nous venons de créer.

Notre type de contenu Billet est à présent disponible pour les contributeurs. Vous pouvez le vérifier en vous reconnectant en tant que Marine et en allant dans le menu [Créer un contenu](#).



Figure 8–39
Accès à la création d'un type de contenu Billet

Ajout de types complexes

Le module CCK

L'ajout d'un contenu s'appuyant sur les champs standards de Drupal est une opération simple et directe. Mais lorsque le besoin de champs supplémentaires pour un type de contenu à créer apparaît, les choses deviennent un peu moins simples.

À l'instar de la gestion des filtres, la gestion de nouveaux types de contenu plus spécialisés que Article ou Page sont délégués aux modules. Pour prendre en charge un nouveau type de contenu, vous pouvez dès lors soit trouver un module qui réponde à vos besoins, soit en développer un qui soit parfaitement adapté à votre situation. Cette seconde solution est particulièrement intéressante pour ceux qui cherchent à créer un site professionnel et/ou à haut trafic, car cette approche est la plus efficace tant sur le plan des performances que sur celui de la souplesse d'adaptation à vos besoins réels.

Ceci étant dit, il existe bien évidemment une solution intermédiaire entre ce que propose Drupal et le développement d'un module personnalisé : c'est le module CCK.

CCK (*Custom Content Kit*) est un module très imposant dont vous entendrez parler à maintes reprises. Sa principale finalité est de permettre l'ajout de nouveaux champs à n'importe quel type de contenu. Une fois que vous l'avez téléchargé et installé, dirigez-vous vers l'administration des modules.

Dans la section **CCK**, vous trouverez pas moins de neuf modules. Les modules principaux sont Content et Option Widgets, qu'il faut activer tous les deux.

Nous avons ensuite une série de modules dédiés chacun à un type de champ :

- **Number** pour les champs de type numérique ;
- **Text** pour les champs de type texte ;
- **Node Reference** pour les champs permettant de référencer des contenus ;
- **User Reference** pour les champs référençant des utilisateurs.

Activez l'ensemble de ces modules pour les besoins de notre exemple.

Vous trouverez également les modules :

- **FieldGroup** permettant de regrouper différents champs dans des sections dépliables comme celles que vous avez déjà rencontrées ;
- **Content Permission** qui permet de définir les droits d'accès au niveau d'un champ ;
- **Content Copy** qui propose d'exporter et d'importer les définitions de champs d'un serveur à un autre.

Il n'est pas nécessaire pour l'instant d'activer ces modules. D'une manière générale – et on ne le répétera jamais assez –, n'activez que les modules dont vous avez besoin.

Création du type de contenu Référence

Pour mettre en œuvre notre nouveau module CCK, nous allons créer le type de contenu Référence dont nous avons parlé dans le chapitre « Bien définir son projet ».

Référence
→ Objet de la mission
→ Date de la mission
→ Prestation réalisée
→ Résultats obtenus
→ Domaine de compétence
→ Acteurs

Figure 8–40
Définition du type de contenu Référence

URL Module CCK

► <http://drupal.org/project/cck>

Figure 8-41
Saisie des informations décrivant
le type de contenu Référence

The screenshot shows the 'Nom' (Name) field with the value 'Référence'. Below it is a text area with instructions: 'Le nom affiché de ce type de contenu. Ce texte sera affiché dans la liste sur la page créer un contenu. Ce nom doit commencer par une lettre majuscule et ne doit contenir que des lettres, chiffres, et espaces. Ce nom doit être unique.' The 'Type' field has the value 'reference'. Below it is another text area with instructions: 'Le nom interne de ce type de contenu. Ce texte sera utilisé pour construire l'URL de la page créer un contenu. Ce nom ne doit contenir que des lettres minuscules, des chiffres, et des caractères de soulignement seront convertis en trait d'union lors de la construction des URL de contenu. Ce nom doit être unique.' The 'Description' field has the value 'Utilisez ce type de contenu pour ajouter une nouvelle référence client'.

Le reste du type de contenu se paramètre comme une Page sans commentaires et sans fichiers attachés, avec *Publication immédiate* mais sans l'option *Promu en page d'accueil*.

Traitement des champs Titre et Corps

Avant d'enregistrer, arrêtons-nous un instant sur la section *Paramètres du formulaire de contribution*. Cette section permet de changer les libellés des deux champs standards de Drupal : *Titre* et *Corps*.

Il peut sembler astucieux de remplacer les libellés de ces deux champs par « Objet de la mission » et « Prestations réalisées » : cela nous ferait moins de champs à ajouter avec CCK, et l'on s'imagine déjà que cela optimisera le stockage. Même si cette démarche semble logique, elle n'en est pas moins dangereuse.

Il ne faut en effet jamais oublier que Drupal stocke ses champs dans des tables en base de données. Le titre et le corps sont stockés dans les tables *node* pour la première révision du document, puis *node_revisions* pour les révisions suivantes. Les champs dans la table sont *title* pour le titre et *body* pour le corps.

Pour des raisons de lisibilité – notamment le jour où vous désirerez exploiter vos contenus avec d'autres outils que Drupal –, avoir des noms de champ sans aucun rapport avec ce qu'ils contiennent ne facilitera pas le transfert et la mise en place de vos données dans un nouvel outil.

Lorsque vous créez des champs avec CCK, ce module va automatiquement générer une nouvelle table nommée *content_type_reference* qui contiendra un champ de base de données par champ CCK ajouté. La différence dans ce cas réside dans le fait que nous avons – comme nous le verrons plus loin – le contrôle sur le nommage du champ créé. Vous aurez donc des noms de champ parfaitement cohérents avec les données qu'ils contiennent.

Dans le cas de notre type de contenu Référence, l'idéal serait de pouvoir supprimer ces deux champs standards *Titre* et *Corps* pour créer ensuite l'ensemble des nouveaux champs personnalisés. Le problème est que si faire disparaître le champ *Corps* ne pose aucune difficulté, Drupal n'autorise en revanche pas qu'un contenu n'ait aucun *Titre* (voyons cela comme l'une des rares limitations du produit). Nous ne pourrions donc pas obtenir totalement l'effet désiré : il nous faut transiger en remplaçant le libellé du champ *Titre* par *Objet de la mission* et en faisant disparaître le champ *Corps*.

Pour ce faire, dans la section *Paramètres du formulaire de contribution*, il suffit de changer le libellé de *Titre* et de vider celui de *Corps*.

Figure 8–42
Modification du libellé du titre et suppression du corps

Nous pouvons maintenant enregistrer notre contenu et passer à l'ajout des champs manquants.

Ajout des nouveaux champs

Peut-être avez-vous déjà noté que l'activation du module CCK a induit une légère modification dans la page *Définition des contenus*. Sur chaque ligne de la liste, nous avons à présent une nouvelle opération : *gérer les champs*.

Figure 8–43
La ligne correspondant à notre nouveau contenu

Lorsque nous cliquons sur ce lien, nous arrivons sur une page qui nous permettra d'étendre le type de contenu comme nous l'entendons.

La première partie de la page nous donne la liste des champs actuellement disponibles pour ce contenu. Il s'agit là des champs par défaut de Drupal et, comme prévu, le champ *Corps* a effectivement disparu de la liste et le champ *Titre* a été correctement renommé.

Étiquette	Nom
✚ Objet de la mission	Formulaire du module node.
✚ Paramètres du menu	Formulaire du module menu.
✚ Fichiers attachés	Formulaire du module upload.

Figure 8–44
La liste des champs par défaut du type de contenu Référence

Figure 8–45
Ajout d'un nouveau champ
au type de contenu
Référence

La première zone contient l'étiquette du champ, nous y saisissons donc « Prestation réalisée ».

Dans la zone suivante vient le nom du champ. Il s'agit d'un nommage informatique qui ne doit contenir que des symboles alphanumériques non accentués et éventuellement le symbole souligné. Saisissez `prestation_realisee`. Notez que cet identifiant précédé du préfixe `field_` servira de nom pour ce champ dans notre table `content_type_reference`.

Suit une liste permettant de définir le type de données pris en charge par le champ. Dans notre cas, s'agissant d'un simple texte, nous sélectionnerons donc `Texte`.

La dernière liste permet de choisir le composant (appelé widget) permettant l'édition du champ : une `Zone de texte` conviendra parfaitement. Notre champ étant défini, appuyez sur `Enregistrer` pour passer au paramétrage du champ.

Figure 8–46
Définition du fonctionnement
du champ Prestation réalisée

Ici peu de choses sont à modifier, si ce n'est le nombre de lignes (`Rangées`) du champ, un `Texte d'aide` pour le contributeur à propos de ce champ, et son caractère obligatoire en cochant la case `Obligatoire`.

Figure 8–47
Ce champ est obligatoire.

Ceci fait, appuyez sur `Enregistrer les paramètres` du champ pour créer effectivement le nouveau champ.

Répétez l'opération pour le champ *Résultats obtenus* qui lui aussi est un champ texte avec un contrôle de type *Zone de Texte*.

Ajout d'un champ multiple

Le champ suivant à ajouter est *Acteurs*. Il s'agira cette fois de saisir non plus une valeur simple, mais une liste d'utilisateurs dont la longueur est variable.

Ajoutons pour commencer un champ, *Acteurs*, qui cette fois, a pour type *User reference*, c'est-à-dire qu'il contiendra un nom d'utilisateur. Dans la liste des contrôles, nous utiliserons *Champ texte à auto-complétion*, un contrôle sympathique permettant de retrouver un utilisateur en tapant seulement les premières lettres de son nom.

The screenshot shows a 'Nouveau champ' (New field) dialog box. It contains four input fields with labels below them: 'Acteurs' (labeled 'Étiquette'), 'field_ acteurs' (labeled 'Nom du champ (a-z, 0-9, _)'), 'User reference' (labeled 'Type de données à stocker'), and 'Champ texte à auto-complètement' (labeled 'Élément du formulaire pour l'édition des données').

Figure 8–48
Ajout du nouveau
champ Acteurs

Dans les paramètres du champ, nous allons d'abord remplir les champs habituels (*Obligatoire*, *Description*, etc.) et nous intéresser de plus près au champ *Nombre de valeurs*.

Ce champ détermine combien de valeurs il sera possible d'ajouter dans notre champ *Acteurs*. Jusqu'à maintenant, cette option avait été laissée à *1*, indiquant qu'une seule valeur était possible. Mais comme nous pouvons avoir pour ce champ une série d'utilisateurs, cela ne conviendra pas. Nous allons donc sélectionner *Illimité*, ce qui nous permettra d'ajouter autant d'acteurs que nécessaire.

The screenshot shows a dropdown menu for 'Nombre de valeurs :'. The selected option is 'Illimité'.

Figure 8–49
Champ Nombre de valeurs positionné sur Illimité

Entre *1* et *Illimité*, il nous est possible de choisir un nombre fixe de valeurs. Ceci est utile lorsque nous connaissons à l'avance le nombre maximal de valeurs possibles, ce qui n'est pas le cas ici.

La dernière chose que nous allons modifier ici est l'activation de la référence arrière (ou rétrolien) sur l'utilisateur. Le principe est que tout acteur qui sera ajouté dans ce champ verra sa page de profil enrichie par la liste des contenus auxquels il a participé. Cochez pour cela l'option *Lien retour*.

The screenshot shows a checkbox labeled 'Lien retour' which is checked. Below it, the text 'Si cette option est sélectionnée' is visible.

Figure 8–50
Activation de l'option Lien retour

URL **Module Date**

► <http://drupal.org/project/date>

Notre champ multiple est maintenant prêt à être utilisé, nous pouvons l'enregistrer pour passer au dernier champ, la date de la mission.

Ajout d'un type de champ

Le problème est que CCK ne dispose pas d'un type de champ Date en standard. Mais CCK est un véritable écosystème à lui tout seul. Il existe ainsi de très nombreux modules de contribution dont leur seul objectif est d'étendre ses fonctionnalités en ajoutant un type de champ, un mode d'affichage, etc.

Pour la gestion des dates, le module à installer s'appelle tout simplement Date.

Au moment de l'activer, vous constaterez que Date, comme CCK, est un module assez complexe permettant de gérer à peu près tous les aspects d'un calendrier. Comme CCK, il est composé de nombreux sous-modules. Il convient de tous les activer à l'exception de Date PHP4, qui n'a d'intérêt que si vous utilisez encore une version 4 de PHP.

Une fois le module Date activé, vous pouvez retourner sur la page [Gérer les champs](#) du type de contenu Référence. Vous constatez alors que la liste des types de champs s'est étoffée d'une série de formats de date.

Nous pouvons à présent créer notre champ *Date de la mission* avec comme identifiant `date_mission`. Vient alors le choix du type de champ qui mérite un peu d'attention, car nous avons ici trois formats différents :

- *Date* est à proscrire car il consiste à stocker une chaîne de caractères représentant la date voulue. Ce format sera difficile à exploiter par d'autres applications que Drupal. Mais surtout, il mine les performances de toutes les listes que vous pourrez faire en utilisant cette date comme critère ;
- *DateTime* utilise quant à lui le type spécifique Date de votre base de donnée. Cette approche est compatible avec une utilisation externe des données, mais pas avec le reste de Drupal qui manipule difficilement ce format ;
- *Datstamp* a l'avantage d'être simple à manipuler par d'autres applications et nativement utilisable par Drupal. Toutes les dates de Drupal sont déjà stockées dans ce format (ex. la date de création ou de mise à jour d'un nœud). C'est en fait un timestamp Unix (appelé aussi epoch Unix), c'est-à-dire qu'il indique le nombre de secondes écoulées entre la date indiquée et le 1^{er} janvier 1970. C'est le choix qui vous est conseillé.

Une fois le type de champ *Datstamp* sélectionné, reste à choisir le contrôle qui permettra de l'éditer. Le *Champ Texte avec Pop-up Date* est une solution

agréable et élégante permettant aussi bien une saisie manuelle et directe de la date que l'utilisation d'un petit calendrier pour une sélection visuelle.

Figure 8-51 Création du nouveau champ Date de la mission

Il ne reste plus qu'à valider pour passer au paramétrage du champ.

Dans le paramétrage, nous avons ici davantage d'éléments à changer. Au-delà des classiques ligne d'aide et champ *Obligatoire*, il est pertinent d'écarter la gestion de l'heure et du jour pour ce type de date, car ils présentent peu d'intérêt dans le contexte. Pour cela, sélectionnez seulement *Année* et *Mois* dans la zone *Granularité*.

Figure 8-52
Seuls les années et les mois comptent...

Comme nous ne prenons pas en compte l'heure, nous désactivons aussi la *Gestion des fuseaux horaires* :

Figure 8-53
Avec seulement des mois et des années,
le fuseau horaire est inutile.

Voilà, notre champ est maintenant prêt, nous pouvons donc l'enregistrer.

Ordonnancement des champs

À chaque fois qu'on termine l'enregistrement d'un nouveau champ, CCK nous transporte sur la page *Gérer les champs* avec une liste de champs visibles. Les champs sont pour l'instant classés dans l'ordre de leur création, mais il est cependant possible via le système habituel de glisser-déposer de les déplacer pour les mettre en accord avec ce qu'attend le contributeur.

Figure 8–54
Les champs, une fois réorganisés

Étiquette	Nom	Type
+ Date de la mission *	field_date_mission	Datestamp
+ Objet de la mission	Formulaire du module node.	
+ Prestation Réalisée	field_prestation_realisee	Texte
+ Résultats Obtenus	field_resultats_obtenus	Texte
+ Acteurs	field_acteurs	User reference
+ Paramètres du menu *	Formulaire du module menu.	

Lorsque l’ordre des champs vous convient, enregistrez l’ensemble pour finir.

Affichage des champs

Nous avons jusqu’à présent travaillé sur l’aspect d’édition de notre type de contenu Référence ; il est temps d’envisager ce qui sera affiché pour les visiteurs du site. Pour ce faire, cliquez sur l’onglet *Afficher les champs*.

Sur cette nouvelle page il est possible de définir précisément la manière dont un contenu sera affiché, selon le type de rendu (basique, lors d’un résultat de recherche, dans un flux RSS), ainsi que le mode de rendu (contenu complet ou abrégé).

Les types de rendu sont sélectionnables via les sous-onglets *Basique*, *Recherche* et *RSS*. Dans un premier temps, travaillons sur le type de rendu Basique, qui correspond à notre usage pour le moment.

Nous disposons ici de la liste des champs définis via CCK. On notera et regrettera l’absence du champ *Titre* que nous avons renommé et qu’il n’est pas possible de paramétrer ici.

- La première colonne permet de changer la position de l’étiquette du champ. Nous laisserons ces valeurs à *Au dessus* sauf pour la date, un champ court, où nous préférons *Sur la même ligne*.
- La seconde colonne détermine l’affichage du contenu dans son mode abrégé. Nous allons ici tout cacher (<Hidden>) sauf pour *Prestation réalisée*.
- La colonne *Nœud Complet* définit cet affichage pour le mode complet. Laissez les valeurs par défaut pour les deux champs texte, et choisissez le format *Short* (court) pour la date.

Une fois le format d’affichage défini, il est temps de tester notre nouveau type de contenu.

Champ	Étiquette	Résumé	Exclure	Nœud complet	Exclure
Date de la mission	Sur la même ligne ▾	<Hidden> ▾	<input type="checkbox"/>	Short ▾	<input type="checkbox"/>
Prestation Réalisée	Au dessus ▾	Par défaut ▾	<input type="checkbox"/>	Par défaut ▾	<input type="checkbox"/>
Résultats Obtenus	Au dessus ▾	<Hidden> ▾	<input type="checkbox"/>	Par défaut ▾	<input type="checkbox"/>
Acteurs	Sur la même ligne ▾	<Hidden> ▾	<input type="checkbox"/>	Par défaut ▾	<input type="checkbox"/>

Figure 8–55 Paramétrage du mode d’affichage de chaque champ

Création d’un contenu Référence

Nous avons décidé de ne pas donner les droits sur ce type de contenu aux contributeurs, seul l’administrateur y a donc accès. Nous allons tester cela en cliquant sur *Créer un contenu*, puis *Référence*.

Le formulaire s’utilise ici de la même manière que dans le cas des autres types de contenu, avec toutefois les spécificités ajoutés par CCK et Date.

Objet de la mission : *

Recrutement d'un équipe Exploitation pour une grande banque Allemande

Date de la mission : *

06.2009

4 Juin 2009

DI	LU	MA	ME	JE	VE	SA
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30				

on en un temps record d'un équipe dédiées à l'expl

Décrivez ici la prestation réalisée

Résultats Obtenus : *

Ils sont tous très bon, et le client semble satisfait, du moins pour l'instant...

Figure 8–56
Le champ Date de la mission et son calendrier en action

Pour notre champ à valeurs multiples (*Acteurs*), nous disposons bien de tous les éléments nécessaires à sa gestion. Il est aussi possible de gérer l’ordre des champs :

Acteurs : *

+ Marine

+ Karine

+

Ajouter un autre élément

Figure 8–57
Ajout d’un nombre illimité d’acteurs

Vous pourrez être surpris à juste titre par l'absence de bouton *Supprimer*. Le fonctionnement ici est qu'une valeur vide est une valeur détruite. Si vous enlevez cette valeur, le champ sera automatiquement supprimé.

De même, on pourrait s'étonner qu'il y ait deux champs vides par défaut. C'est un comportement voulu de CCK et non pas un problème lié à votre paramétrage.

Une fois le contenu saisi, vous pouvez l'enregistrer pour juger du rendu final en mode complet.



Figure 8-58
Le contenu de type Référence
une fois enregistré

Nous avons là toutes les informations saisies dans chacun des champs supplémentaires. Nous pouvons maintenant vérifier l'affichage en mode abrégé sur la page d'accueil.



Figure 8-59
La version résumée d'un contenu
de type Référence

ATTENTION

CCK, PostgreSQL et les révisions

Dans la version courante de Drupal 6, il y a un bug encore non corrigé pour PostgreSQL et quelques configurations de MySQL, qui empêche la sauvegarde d'un contenu CCK avec les révisions activées. Plus d'informations sur cette anomalie ici :

► <http://drupal.org/node/261258>

Tous les champs que nous avons décidés de cacher sont effectivement invisibles : l'ensemble est satisfaisant.

Les limites de CCK

CCK est un excellent module, indispensable à l'écosystème Drupal. Malgré tout, il est important de garder la tête froide et, avant de se lancer, de mettre son utilisation en balance avec son alternative naturelle, la rédaction de vos propres modules de gestion d'un contenu. Dans cet esprit, et avant tout pour des sites de taille importante, les points suivants sont à prendre en compte.

Performances

Pour les personnes qui auraient un mauvais souvenir de ce module sous Drupal 5, les problèmes de performance de CCK sont aujourd'hui une chose réglée avec Drupal 6.

Déploiement

CCK vous oblige à passer par une procédure manuelle de copier-coller pour exporter vos types de contenu, puis pour les importer sur votre serveur de production. Contrainte supplémentaire, les types de contenu ne sont pas exportables d'un coup, vous obligeant à opérer type par type. Enfin, si un nouveau champ peut être ajouté par ce moyen, il ne peut en aucun cas être supprimé ou modifié. Ceci explique que la majorité des administrateurs font ce type d'opérations eux-mêmes, en répétant leurs manipulations sur le serveur de production.

Si le type de contenu était géré par un module, il aurait suffi de recopier le code de sa nouvelle version et d'opérer une mise à jour de la base Drupal pour que le déploiement soit réalisé (voir le chapitre 15 sur le déploiement).

Mise à jour

CCK n'autorise pas la modification du type d'un champ (par exemple, changer un entier en chaîne de caractère), ce qui dans certains cas se révélera très gênant, vous amenant là aussi à gérer la migration de vos données « à la main ».

A contrario, pour un module, il suffit de développer des procédures de mise à jour, de les tester sur la plate-forme de développement avant de répéter l'opération sur le serveur de production.

Maîtrise des données

Comme nous l'avons vu avec les champs de type Date, il faut être très prudent en choisissant un type de champ CCK, et toujours s'assurer de la pertinence des types auxquels ils correspondent en base de données. Un mauvais choix (comme le type Date au lieu d'un Timestamp) peut impliquer une perte importante en termes de performance. De plus, certains types de champs fournis par des modules externes à CCK peuvent stocker les données dans un format difficilement exploitable par une requête SQL. L'important est donc de toujours vérifier ce que CCK fait sur la base de données, et si cela vous convient.

Créer des listes de contenus

Lorsque l'on conçoit un site, un des besoins récurrents est l'affichage de listes de toutes sortes et à tout crin :

- liste des derniers commentaires ;
- liste des dernières activités (création de nœuds, de commentaires, d'utilisateurs) ;
- liste des contenus non publiés, etc.

Rien ne permet en standard dans Drupal de créer de telles listes, mais comme souvent, la fourniture de ce type d'affichages est en réalité assurée par des modules, et les nombreuses listes que nous avons déjà abordées sont issues de tels modules (liste des utilisateurs, des commentaires, etc.). Vous avez ici la possibilité d'écrire votre propre module pour afficher des listes, et Drupal offre toute une boîte à outils permettant de le faire sans problème majeur (gestion des tris, de la pagination, etc.). Il existe également des modules de contribution spécialisés dans cette fonction, qui pourront déjà combler bon nombre de vos besoins en ce domaine. La règle est toujours la même : le module doit être le choix prioritaire, et ce choix est d'autant plus pertinent qu'il est rare que votre besoin n'ait pas déjà été celui de beaucoup d'autres personnes avant vous.

Cependant, si vous ne trouvez pas le module de vos rêves, il existe un outil capable de générer n'importe quelle liste : il s'agit du module Views, qui est à peu près aussi célèbre que CCK dans l'univers Drupal.

Comme pour CCK, il se compose de plusieurs modules :

- Views, le module principal qui gère la mécanique interne ;
- Views UI, indispensable pour créer des vues ;
- Views Exporter, qui permet d'exporter et d'importer des vues d'un serveur à l'autre.

Après l'avoir installé, activez simplement, pour l'heure, les deux premiers modules.

Création des vues Articles et Billets

Deux listes vont nous être particulièrement utiles pour implémenter nos deux sections *Articles* et *Billets*. Elles n'ont rien de bien sorcier, car il s'agit simplement de filtrer respectivement les contenus de type Article et Billet, puis de les classer par ordre décroissant de date de création. Pour créer ces listes, rendons-nous dans *Construction du site > Vues*.

Cette page donne la liste des vues actuellement disponibles. Celles apparaissant en grisé sont des vues automatiquement gérées par le module

URL Module Views

► <http://drupal.org/project/views>

Views. Nous y reviendrons un peu plus loin. Pour l’heure, nous allons créer notre première vue en cliquant sur l’onglet [Ajouter](#).

Vues Liste **Ajouter** Importer Outils

Nom de la vue :

Il s'agit du nom unique de la vue. Il ne doit comporter que des caractères alphanumériques et des espaces soulignés. Il sert à identifier la vue en interne et à générer des noms uniques pour les templates de thèmes pour cette vue. Si la vue supprime une vue fournie par un module, le nom ne doit pas être modifié sous peine de créer une nouvelle vue.

Description de la vue :

Cette description apparaîtra dans l'interface utilisateur d'administration de Views pour vous indiquer l'objet de cette vue.

Étiquette de la vue :

Saisissez une étiquette optionnelle pour cette vue ; elle sert uniquement à faciliter le tri des vues sur la page d'administration.

Type de vue :

☒ **Noeud**
Les noeuds constituent le contenu primaire d'un site Drupal.

☐ **Commentaire**
Les commentaires sont des réponses au contenu des noeuds.

☐ **Fichier**
Fichiers maintenus par Drupal et différents modules.

☐ **Locale source**
A source string for translation, in English or the default site language.

☐ **Révision du noeud**
Les révisions du noeud représente l'historique des modifications de ce noeud.

☐ **Terme**
Les termes de taxonomie sont rattachés aux noeuds.

☐ **Utilisateur**
Utilisateurs ayant créé un compte sur votre site.

Le type de vue est la table primaire pour laquelle les informations sont récupérées. Le type de vue contrôle quels arguments, champs, critères de tri et filtres sont disponibles ; par conséquent, une fois qu'il est fixé, il ne peut plus être modifié.

Figure 8–60
Définition de la vue

Ce formulaire nous permet de définir le *Nom de la vue* (un nom informatique, donc sans accents ni espaces), sa *Description* et une *Étiquette* qui ne sera utilisée qu'en interne. Ces champs prendront respectivement les valeurs `liste_des_publications`, `Liste des publications` et `Liste des publications`.

Pensez à donner des descriptions explicites, car dès qu'un nombre important de vues sera créé, elles seront vos seules aides pour ne pas vous noyer dans vos vues.

BON À SAVOIR Que signifient les sections de Views en langage SQL ?

Pour ceux qui connaissent un peu SQL, nous pouvons illustrer le rôle de ces blocs de la manière suivante :

```
SELECT <Champs>
FROM <objet de base défini plus haut>
JOIN <Relations>
WHERE <Filtres>
ORDER BY <Critères de tris>
```

Ensuite vient le *Type de vue* qui détermine sur quel type d'objets la vue portera. Attention, une fois sélectionné, le type d'objet ne pourra plus être changé. Nous allons ici afficher une liste de nœuds, sélectionnons donc *Nœud*, avant de cliquer sur *Suivant*.

Fonctionnement de Views

Même si de très bonnes idées ont été développées par l'auteur de ce module sur le plan de l'interface graphique, elles ont malheureusement aussi le défaut de ne pas suivre les règles en vigueur dans Drupal. L'interface de Views demande donc à être apprise dans ses spécificités.

Nous arrivons à présent sur la page imposante – voire impressionnante – de paramétrage de la vue. Quelques explications sur les blocs que nous y trouvons :

- *Paramètres de la vue* reprend les informations que nous venons de saisir, ou plutôt la seule que nous pouvons encore modifier, à savoir l'*Étiquette* ;
- *Paramètres de base* contient toutes les informations nécessaires à la production finale du résultat (style, mise en forme, limitation d'accès, etc.).

Les cinq blocs suivants permettent de construire la requête qui, elle-même, produira la liste :

- *Relations* définit des éléments provenant d'autres objets que celui que nous avons défini comme objet primaire sur la requête ;
- *Arguments* fait référence aux éléments variables de la requête qui seront extraits du chemin interne ;
- *Champs* désigne les champs qui seront visibles ;
- *Filtres* définit les conditions de la requête ;
- *Critères de tri* présente les champs sur lesquels un tri doit être opéré.

Construction de la liste

Puisque nous cherchons à créer une liste d'articles, la première chose à faire est donc d'ajouter un nouveau filtre. Pour cela, utilisez le bouton + du bloc *Filtres*. Vous verrez alors la zone basse du formulaire changer pour vous permettre de définir le champ à filtrer : nous allons sélectionner le type de nœud *Nœud : Type*. Si vous avez du mal à vous y retrouver, vous pouvez utiliser l'option *Groupe* qui permet de n'afficher que les champs pour un type d'objet donné (ici, *Nœud*).

Paramètres par défaut: Ajouter filtres

Groupes :

Noeud

Whether or not the node is published.

☐ Noeud : Publié ou admin
Filtre et exclut les noeuds non publiés si l'utilisateur courant ne peut pas les voir.

☐ Noeud : Publié ou commenté par l'utilisateur
Display nodes only if a user posted the node or commented on the node.

☐ Noeud : Résumé
Le champ du résumé enregistré. Ceci peut ne pas être valide ou utile pour tous les types de noeuds.

☐ Noeud : Statut des commentaires
Indique si les commentaires sont activés ou désactivés sur le noeud.

☐ Noeud : Titre
Titre du noeud.

☒ Noeud : Type
Type d'un noeud (par exemple, "billet de blog", "discussion de forum", "article", etc.).

Ajouter Annuler

Figure 8–61
Ajout d'un nouveau filtre

Cliquez ensuite sur *Ajouter*, et le bas de la page devrait changer pour permettre de modifier les paramètres du filtre.

Paramètres par défaut: Configure filtre Noeud : Type

Cet élément n'est pour le moment pas exposé. Si vous l'exposez, les utilisateurs pourront changer le filtre lorsqu'ils l'affichent.

Exposer

Opérateur :

☒ Fait partie de
☐ Ne fait pas partie de

Type de noeud :

☐ Billet
☐ Book page
☐ Page
☐ Référence
☐ Newsletter issue
☒ Article

Mise à jour Annuler Ôter

Figure 8–62
Paramétrage du filtre sur les types de contenu

En toute logique, nous choisissons ici *Article* et validons par le bouton *Mise à jour*. Ajoutons à présent le critère nécessaire pour n'afficher que les contenus publiés. Pour cela, nous renouvelons l'opération d'ajout avec cette fois-ci le champ *Noeud : Publié* et sélectionnons la valeur *Oui*.

Nous allons terminer le bloc *Filtres* en ajoutant maintenant le critère *Publié en page d'accueil*. Pour cela, nous allons ajouter un dernier filtre sur le champ *Noeud : Promu en page d'accueil*. Et pour valeur, nous utilisons *Oui*. Le bloc de filtres est maintenant complet.

Filtres

Noeud : Type = Article
Noeud : Publié Oui
Noeud : Promu en page d'accueil Oui

Figure 8–63
Présentation de la liste des filtres

Figure 8–64
Présentation des critères de tri



Notre requête ne comportant ni arguments, ni relations, ni même de champs, notre paramétrage de la requête SQL s'arrête donc ici. En effet, nous cherchons simplement à afficher une liste de nœuds, et nous la présenterons avec tous les champs que Drupal expose habituellement pour ce type d'affichage en liste. C'est dans le bloc *Paramètres de base* que nous allons à présent indiquer ces choix.

- *Titre* : nous devons changer ce champ pour une formulation plus explicite, telle que « Nos dernières publications ».
- *Style* : nous sélectionnons *Non mis en forme*, laissant cela au moteur de thème de Drupal. Vous avez ici la possibilité de générer un tableau ou une grille selon vos besoins.
- *Style de ligne* : nous avons deux possibilités, *Nœud* ou *Champs*. Nous sélectionnons *Node* pour un affichage simple des nœuds comme Drupal le fait d'ordinaire (par exemple, sur la page d'accueil). Dans les options proposées, nous choisissons *Résumé* pour n'afficher qu'une version abrégée de l'article, puis cliquons sur *Mise à jour*.
- *Utiliser la pagination* : choisissez *Pagination complète*. Vous pouvez régler le nombre de nœuds à afficher par page avec le champ suivant : *Éléments par page*.

Notre vue est maintenant prête à être utilisée et vous en avez normalement déjà un aperçu en bas de la page.

L'étape suivante consistera à créer une page à partir de ces *Paramètres par défaut*. Cette opération est réalisée par la création d'un nouvel affichage de type Page et, pour ce faire, sélectionnez *Page* dans la liste de l'onglet vertical à gauche, avant de cliquer sur *Nouvel affichage*.



Figure 8–65
Ajout d'un affichage de type Page

BON À SAVOIR Views pour les développeurs

En tant que développeur de module, il n'est pas toujours facile de savoir comment construire une requête SQL qui fasse appel aux données de nombreux modules.

Views est assez pratique dans ce cas de figure, car en dessous de l'aperçu, vous trouverez la requête SQL qui l'a amené à construire son résultat. Il suffit alors de la copier dans son code et de la nettoyer un peu. Ainsi, sans aller jusqu'à créer des vues, Views est une très bon aide au développement de modules.

Ceci fait, cliquez sur l'onglet *Page* qui vient d'apparaître.



Figure 8-66
Sélection de l'onglet de type Page

Le bloc qui nous intéresse ici est *Paramètres de la page* : il nous permettra de définir le chemin interne de la vue et éventuellement de créer un menu. Seule la première possibilité nous intéresse pour l'instant. Cliquez sur *Chemin : aucun* et saisissez maintenant le chemin *publications*. Notre vue sera ainsi disponible par l'URL <http://www.colchique-conseil.fr/publications>.

Figure 8-67
Définition du chemin d'accès à notre liste

Tester l'affichage

Cliquez sur *Mise à jour* pour valider les changements, puis sur le bouton *Enregistrer*. La vue est maintenant prête à être utilisée. Pour la tester, cliquez sur le lien *Voir « Page »* en haut du formulaire.



Figure 8-68
Présentation de la liste des articles

Nous voilà avec une liste en tout point semblable à la page d'accueil, mais pour les seuls contenus de type Article.

Figure 8–69

Notre vue `liste_des_publications` dans la liste des vues, point de départ pour la duplication.

Duplication de vues

Pour créer la même vue, mais cette fois-ci appliquée aux Billets, l'opération sera beaucoup plus rapide. Retournez dans la liste des vues, et à la ligne de notre vue Articles, cliquez simplement sur le lien *Dupliquer*.

Vue Normal Noeud : <code>liste_des_publications</code> (Liste des publications)		Modifier Exporter Dupliquer Supprimer
Titre : Nos dernières publications	Liste des publications	
Chemin : <code>publications</code>		
Page		

Cette opération a pour effet de créer une nouvelle vue identique à notre vue Articles, nous faisant ainsi gagner beaucoup de temps. Il va cependant falloir la modifier en commençant par changer les valeurs en `liste_des_billets`, `Liste des Billets` et `Liste des Billets`, avant de cliquer sur *Suivant*. Vous aboutissez alors sur la page d'édition de la vue, préchargée avec tout ce que nous avons déjà paramétré.

Les trois seules choses devant être changées sont :

- le *Titre* : dans le bloc *Paramètres de base*, à changer par « Nos derniers billets » ;
- le filtre sur *Nœud : Type* : il suffit pour cela de cliquer sur la valeur, de décocher *Article* pour cocher *Billet*, et enfin de cliquer sur *Mise à jour* ;
- le *Chemin* de la page : sélectionnez l'affichage *Page*, et changez le *Chemin* comme nous l'avons fait pour la précédente vue en utilisant la valeur `billets`.

Il ne vous reste maintenant plus qu'à enregistrer la nouvelle vue puis à la tester comme précédemment.

Les performances de Views

L'utilisation de Views sur de gros projets est à envisager en connaissance de cause. Il ne s'agit pas de critiquer gratuitement cet excellent module, mais plutôt de bien comprendre que son extrême souplesse de paramétrage se paye en termes de performances. La raison de cet état de fait est que Views est obligé à chaque fois de lire dans la base de données pour fabriquer sa requête SQL et de travailler en collaboration avec d'autres modules, pour finalement produire son résultat. Pour donner un rapport de grandeur, là où un module « simple » affichant une liste lancerait logiquement une seule requête pour produire un rendu visuel, Views va en lancer environ huit (comprenant aussi des requêtes en écriture), en mettant finalement le double du temps.

Views est donc un outil très utile et d'une grande souplesse, mais il n'en demeure pas moins lent et donc à envisager avec précaution sur des pages fortement sollicitées, et plus particulièrement si vos utilisateurs sont majori-

tairement identifiés (pour plus d’informations, se reporter à la section sur les performances dans le dernier chapitre, « Mettre son site en production »).

Étendre le modèle de droits

En standard, Drupal propose une gestion des droits d’accès fondée sur les rôles affectés à un utilisateur. Mais cela peut se révéler insuffisant dans de nombreuses situations, ou n’être tout simplement pas adapté à vos besoins. Le système standard de droits d’accès est bien évidemment totalement extensible via des modules qui apporteront d’autres visions sur le sujet. Énumérer l’ensemble de ces modules serait long et inutile, car il s’agit avant tout de couvrir vos besoins. Nous allons simplement nous attarder sur un module particulièrement représentatif, appelé Node Access.

Le module Node Access permet de définir des droits pour un contenu spécifique. Ces droits viendront compléter les droits par rôle de Drupal. Il sera ainsi possible, pour un seul contenu, non seulement d’ajouter de nouveaux rôles autorisés à le visualiser, à le modifier ou à le supprimer, mais plus intéressant encore, d’ajouter des utilisateurs qui n’avaient pas à l’origine un rôle donnant accès à ce contenu.

En allant dans *Gestion des utilisateurs*, puis *Node Access*, vous avez tout d’abord la possibilité de définir les types de contenu qui bénéficieront de cette nouvelle gestion des droits, ainsi que les permissions par défaut associées à chaque rôle pour un type de contenu donné.

Article

☒ Show grant tab for this node type

Role	Voir	Modifier	Supprimer
utilisateur anonyme	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
utilisateur identifié	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
contributeur	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
correcteur	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
validateur	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Author Settings	Voir	Modifier	Supprimer
Node author	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

The settings selected for the node author will define what permissions the node author has. This cannot be changed on individual node grants.

Figure 8–70
Paramétrage des permissions
par défaut par type de contenu

Dans cet exemple, nous ne faisons que reproduire le comportement standard de Drupal et de nos réglages de rôles.

Plus loin, vous pourrez également choisir les rôles autorisés à recevoir des droits spéciaux, par exemple *authenticated user* (Utilisateur identifié).

Allowed Roles

The selected roles will be listed on individual node grants. If you wish for certain roles to be hidden from users on the node grants tab, make sure they are not selected here. You may also provide an alias for each role to be displayed to the user and a weight to order them by. This is useful if your roles have machine-readable names not intended for human users.

Allow Role	Alias	Poids
<input type="checkbox"/> anonymous user		0
<input checked="" type="checkbox"/> authenticated user		0
<input type="checkbox"/> contributeur		0
<input type="checkbox"/> correcteur		0
<input type="checkbox"/> validateur		0

Figure 8-71
Sélection des rôles ayant des droits sur les contenus

Une fois le paramétrage enregistré, le module Node Access affiche un message indiquant que la table des droits doit être reconstruite. Suivez le lien et reconstruisez la table.

Figure 8-72
Message demandant la reconstruction de la table de droits sur les contenus

Les permissions d'accès au contenu doivent être reconstruites. Veuillez vous rendre sur [cette page](#).

Ceci fait, pour un type de contenu donné – Article, dans le cas présent –, le module va ajouter un onglet *Grant* disponible pour chaque contenu.

Figure 8-73
Apparition de l'onglet Grant, permettant de définir les droits sur un contenu

Bienvenue Voir Modifier Révisions **Grant**

Figure 8-74
Assignment des droits de modification à l'utilisateur Pierre, qui n'est pas un contributeur

Enter names to search for users:

Pierre Recherche

Utilisateur	Keep?	Voir	Modifier	Supprimer
Pierre	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Sur cette page, vous avez la possibilité d'ajouter des droits pour des utilisateurs qui n'ont pas le rôle Contributeur, ou de redéfinir les droits des rôles que nous avons activés dans la configuration précédente.

Nous avons donné ici à l'utilisateur Pierre, qui n'est pas contributeur, les droits de visualisation et de modification sur le contenu Bienvenue. Notez que l'ajout d'utilisateurs n'est pas très ergonomique, il faut entrer les premières lettres du nom, puis cliquer sur *Recherche* avant de pouvoir l'ajouter.

De la même manière, nous aurions pu modifier les droits généraux des utilisateurs authentifiés pour ce nœud.

Role	Voir	Modifier	Supprimer
authenticated user	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Figure 8-75

Définition des droits sur le contenu pour les utilisateurs authentifiés

Comme vous le voyez, le champ d'utilisation de ce nouveau modèle de droits est assez large, mais aussi beaucoup plus complexe à gérer que celui « par rôle » de Drupal. Faites toujours très attention et vérifiez systématiquement ce que vous faites ; veillez notamment à ne pas rendre public un contenu que vous cherchez à protéger.

En résumé

- Le module Book permet de hiérarchiser des contenus les uns par rapport aux autres. Une hiérarchie appelée Livre peut regrouper des contenus de tout type.
- Les révisions permettent d'historier chacune des modifications apportées au contenu. Pensez toujours à indiquer les raisons de vos modifications de contenu dans le journal des révisions.
- Le module CCK permet de créer des types de contenu complexes, dotés de nouveaux champs. CCK est un outil fondamental pour structurer les types de contenu.
- Le module Views permet de créer des listes de contenus présentées sous la forme de pages.
- Il est possible d'étendre le système de droits sur les contenus en ajoutant de nouveaux modules. Attention, ces modules sont potentiellement incompatibles les uns avec les autres.

MODULES MAGIQUES Les autres modules de droits pour les contenus

De nombreux autres modules permettent, comme Node Access de définir une stratégie de droits.

Les modules Taxonomy Access Control et Taxonomy Access Control Light (une version plus simple du précédent) permettent tous deux de spécifier des droits en fonction de la taxonomie portée par un contenu (ex. les documents avec le terme « recrutement » sont accessibles par le rôle XXX).

- http://drupal.org/project/taxonomy_access
- http://drupal.org/project/tac_lite

Le module Simple Access permet de faire la même chose que Node Access dans une version plus simplifiée.

- http://drupal.org/project/simple_access

Organic Group – un gros acteur dans le monde des modules Drupal – permet de gérer des groupes d'utilisateurs. Son modèle d'accès aux contenus s'appuie lui aussi sur des groupes.

- <http://drupal.org/project/og>

Faites cependant attention de bien systématiquement tester le fonctionnement de ces modules, surtout si vous comptez en cumuler plusieurs. Ils ne sont pas tous compatibles et il peut en résulter que des contenus soient visibles pour tous alors qu'ils ne le devraient pas.

9

chapitre

The screenshot shows a Drupal website interface with several numbered annotations:

- 1**: Points to the **administrateur** menu in the left sidebar.
- 2**: Points to the footer containing the text: "© Colchique Conseil - Tous droits réservés - Contactez nous - Ce site est propulsé par Drupal."
- 3**: Points to the **Recrutement** link in the bottom right corner of the article content area.
- 4**: Points to the top navigation bar with the logo and the slogan "Colchique , car chaque brin d'herbe est unique..."
- 5**: Points to the **Un nouvel article** section header.
- 6**: Points to the **Bienvenue** section header.

The main content area displays two articles:

- Un nouvel article**
mar, 06/03/2009 - 10:20 — [Marina](#)
Corps de l'article...
[Ajouter un commentaire](#)
- Recrutement d'un équipe Exploitation pour une grande banque Allemande**
mar, 06/03/2009 - 11:42 — [administrateur](#)
Prestation Réalisée:
La mission a consisté en la constitution en un temps record d'un équipe dédiées à l'exploitation du système informatique d'une grande banque allemande.
- Bienvenue**
mar, 06/03/2009 - 23:38 — [Marina](#)
Bienvenue sur le site de **Colchique Conseil**.
le cabinet de conseil en ressources humaine qui a l'hélique à coeur.
[En savoir plus](#) - [2 fichiers attachés](#)

Structurer les pages

Nous maîtrisons maintenant tous les aspects de la gestion de contenus ; il est donc temps de s'intéresser à leur présentation. Drupal nous permet d'envisager ce travail sous deux angles distincts : l'organisation des pages par zones fonctionnelles et le changement radical de l'apparence de ces pages à travers les thèmes. Nous aborderons le premier point dans ce chapitre, et traiterons le deuxième quelques pages plus loin, au sein du chapitre 11, « Modifier l'apparence ».

SOMMAIRE

- ▶ Comprendre les blocs
- ▶ Fabriquer ses propres blocs
- ▶ Gérer finement la visibilité des blocs
- ▶ Fabriquer des nouveaux menus

Gérer les blocs

Le modèle de présentation

La disposition des éléments dans une page fabriquée par Drupal repose sur deux concepts : les régions et les blocs.

- Un *bloc* est une zone rectangulaire contenant des informations et éventuellement un titre.
- Une *région* est une partie de la page qui est capable de recevoir une liste de blocs.

Les régions ne sont pas des zones statiques et prédéfinies par Drupal. Elles sont directement gérées par le thème que votre site utilise à un moment donné. Le nombre de régions, ainsi que leur emplacement, peut donc varier d'un thème à l'autre. Cependant, la majorité des thèmes pour Drupal définit au moins cinq régions de base, et le thème Garland, que vous utilisez sans le savoir depuis le début de votre exploration, n'échappe pas à cette règle. Nous avons ainsi la barre de gauche ①, la barre de droite ③, l'en-tête ④, le pied de page ② et le contenu ⑤.



Figure 9-1
Les grandes régions
d'un thème Drupal

L'en-tête et la barre de droite sont ici réduits à leur plus simple expression : ces deux régions sont en effet vides de blocs. En revanche, nous avons un bloc Connexion utilisateur sur la région Barre de gauche, et un bloc Propulsé par Drupal au niveau du pied de page.

La liste des blocs

Les deux questions que l'on se pose à ce stade sont : « D'où viennent les blocs ? » et « Comment en ajouter sur une page ? » Pour la première question, vous devriez maintenant avoir une petite idée de la réponse : ce sont bien entendu les modules qui sont à l'origine des blocs. Chaque module peut fournir un certain nombre de blocs, que vous pouvez ensuite positionner où bon vous semble, tant qu'il s'agit d'une région définie par le thème.

Pour passer de la théorie à la pratique, dirigeons-nous vers la page de gestion des blocs, via [Construction du site](#) > [Blocs](#). Sur cette page, vous constatez que toutes les régions que nous venons d'évoquer sont matérialisées par des zones en pointillés.

Dans le tableau qui se trouve au centre de cette page, nous avons la liste des régions fournies par le thème (en gras), et en dessous de chaque région, la liste des blocs qui la composent actuellement. Lorsque la région est vide de blocs, Drupal vous prévient par le message [Aucun bloc dans cette région](#).

Bloc	Région	Opérations
Barre de gauche		
+ Connexion utilisateur	Barre de gauche	configurer
+ Navigation	Barre de gauche	configurer
Barre de droite		
Aucun bloc dans cette région		
Contenu		
Aucun bloc dans cette région		
Entête		
Aucun bloc dans cette région		
Pied de page		
+ Propulsé par Drupal	Pied de page	configurer
Désactivé		
+ Commentaires récents	<aucun>	configurer
+ Liens primaires	<aucun>	configurer
+ Liens secondaires	<aucun>	configurer
+ Nouveaux membres	<aucun>	configurer
+ Syndication	<aucun>	configurer
+ Sélecteur de langue	<aucun>	configurer
+ Utilisateurs en ligne	<aucun>	configurer

Figure 9–2
Panneau de gestion des blocs

BON À SAVOIR

Bloc de navigation dans un livre

Nous parlions précédemment dans cet ouvrage du module Book qui nous a permis de créer notre FAQ. Ce module fournit lui aussi un bloc extrêmement pratique, nommé *Book Navigation*. En ajoutant ce bloc au sommet de la région *Barre de droite*, vous disposerez ainsi d'un système de navigation dans les livres, et donc dans la FAQ, intégrée à votre site. Ce bloc est paramétrable pour ne s'afficher que lorsque l'utilisateur est déjà sur une page de livre ; pour ce faire, cochez l'option *Show block only on book pages* (ne montrez ce bloc que sur les pages de livre) et enregistrez la configuration du bloc.

Figure 9–3
Réorganisation des blocs

Enfin, en bas de la liste, dans la section *Désactivé*, se trouvent les blocs qui ne sont présents dans aucune région.

Par défaut, Drupal a juste positionné, lors de son installation, trois blocs fondamentaux :

- le bloc *Connexion utilisateur* sur la région *Barre de gauche*, correspondant à la boîte de connexion que voient les utilisateurs anonymes et que nous avons maintenant l'habitude d'utiliser. Nous découvrons ainsi un comportement important des blocs : ils peuvent devenir invisibles selon le contexte de la page (par exemple, lorsque l'utilisateur est déjà identifié) ;
- le bloc *Navigation*, placé également sur la région *Barre de gauche* et que nous avons déjà largement utilisé, puisqu'il est notre menu *Administrateur*. Là aussi, cela nous apprend une autre caractéristique des blocs : leur contenu est dynamique et peut changer en fonction du contexte. Dans le cas présent, en fonction du rôle de l'utilisateur connecté, un menu correspondant à ses droits sera affiché ;
- le bloc *Propulsé par Drupal* est positionné dans la région *Pied de page*.

Positionner de nouveaux blocs ou placer des blocs encore désactivés dans une région de votre choix est une opération très simple. Il suffit d'utiliser le système de glisser-déposer que nous avons déjà mis en œuvre plusieurs fois. L'autre manière de faire consiste à utiliser la liste déroulante qui se trouve en face de chaque nom de bloc, mais force est de reconnaître que cela est beaucoup moins pratique.

Par exemple, pour respecter la présentation que nous avons définie lors de la conception de notre storyboard, la boîte de connexion et le menu ne doivent pas être à gauche, mais à droite. Il suffit pour cela de faire glisser ces deux blocs sous la région *Barre de droite*.

Bloc	Région	Opérations
Barre de gauche		
Aucun bloc dans cette région		
Barre de droite		
 Connexion utilisateur*	Barre de droite ▾	configurer
 Navigation*	Barre de droite ▾	configurer

Notez que vous pouvez de la même manière organiser les blocs les uns par rapport aux autres au sein d'une même région. Une fois l'opération réalisée, appuyez sur le bouton *Enregistrer les blocs* et retournez sur la page d'accueil pour juger du résultat. Vous disposez d'une nouvelle colonne de droite et celle de gauche a disparu.

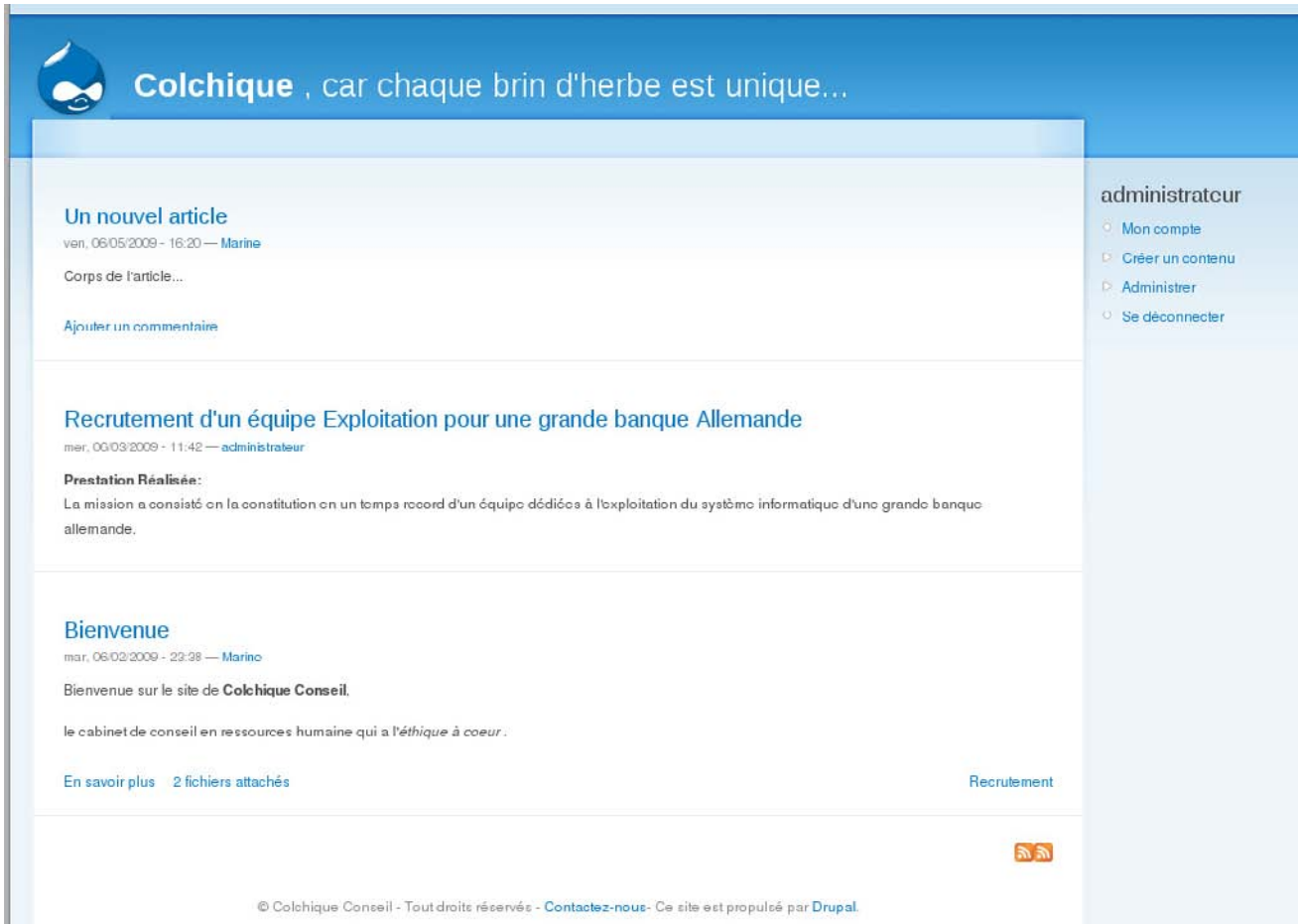


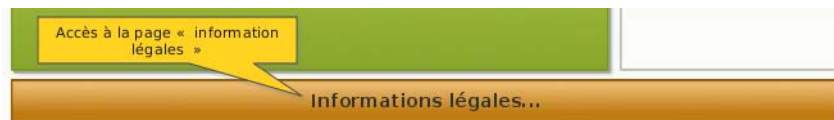
Figure 9-4 Déplacement des blocs vers la droite

Les blocs personnalisés

Création d'un bloc personnalisé

Un bloc est généralement fabriqué par un module. Les blocs disponibles en standard sont donc issus des modules installés dans Drupal par défaut. Pour ajouter de nouveaux blocs, il nous faut soit puiser dans ceux qui existent déjà mais qui sont encore désactivés, soit installer des modules supplémentaires qui nous fourniront de nouveaux blocs, soit au final rédiger un module qui fabriquera le ou les blocs répondant à nos besoins spécifiques.

Figure 9-5
Ce que nous cherchons à obtenir
pour le pied de page



BONNES PRATIQUES

Sachez décrire ce que vous faites

De manière générale – et ceci vaut pour toutes les descriptions de bloc, de type de contenu ou de vue –, soyez intransigeant quant à la qualité des descriptions que vous donnez, même – et surtout – si elles ne sont pas destinées à être affichées sur le site. Lorsque vous aurez une vingtaine de types de contenu, une centaine de vues et beaucoup plus encore de blocs, vous vous ferez plaisir d'avoir pris cette bonne habitude.

Pour fabriquer ce bloc, nous devons retourner sur la page de gestion des blocs et cliquer sur l'onglet *Ajouter un bloc*. Le formulaire qui apparaît commence par la *Description du bloc* qui sera affichée sur la page de gestion. Donnez ici une description suffisamment explicite pour repérer votre bloc facilement dans la liste, qui sera à terme très volumineuse.

Nous avons ensuite le *Titre du bloc* qui est optionnel, et encore plus lorsqu'il s'agit d'un bloc destiné au pied de page ou à l'en-tête. Nous le laisserons donc vide pour rédiger le *Corps du bloc*, qui nous intéresse davantage puisqu'il contiendra les informations légales concernant notre site.

Comme les contenus, les blocs disposent de formats d'entrée que vous pouvez sélectionner dans la section *Formats d'entrée* présente sous le champ *Corps du bloc*. Pour ne pas être embêté avec un éventuel filtrage du code HTML, nous choisissons ici *Full HTML*. Comme nous avons dans le précédent chapitre activé TinyMCE pour ce format d'entrée, nous bénéficions par conséquent de l'éditeur visuel.

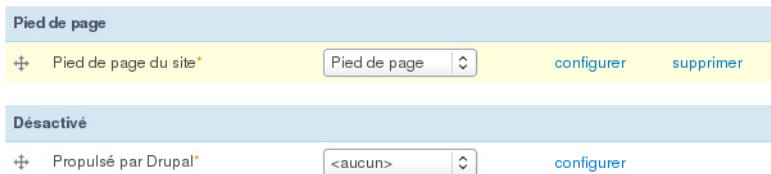
Figure 9-6
Création d'un bloc personnalisé
pour le pied de page

Ceci fait, laissons pour l'instant telles quelles les options de paramétrage qui suivent et cliquons sur *Enregistrer le bloc*. De retour sur la page de configuration des blocs, vous devez voir apparaître dans la section *Désactivé* la description du nouveau bloc.

**Figure 9-7**

Notre bloc est maintenant disponible dans la liste

Comme pour n'importe quel autre bloc, vous pouvez le déplacer dans la région de votre choix, ici *Pied de page*. Vous disposez à présent de deux blocs dans cette région, dont celui nommé *Propulsé par Drupal*. Ce dernier étant quelque peu redondant, déplacez-le dans la section *Désactivé*, puis enregistrez les blocs.

**Figure 9-8**

Déplacement du bloc dans la région Pied de page

Ceci fait, vous pouvez constater que notre nouveau bloc est bien visible en bas de la page.

Suppression d'un bloc personnalisé

Bien entendu, si un bloc personnalisé peut être créé, il peut aussi être supprimé. En tant qu'administrateur, il nous faut retourner dans l'administration des blocs. Sur la ligne de chaque bloc que nous avons créé, nous constatons la présence d'un lien *supprimer*. C'est ce lien qui nous permettra, après confirmation, de faire disparaître un bloc. De même, le lien *configurer* permet de retourner sur la page de paramétrage du bloc pour en modifier les réglages.

MODULES MAGIQUES Des blocs en onglets

Il existe un excellent module appelé Quick Tabs qui permet de cumuler plusieurs blocs en un seul sous la forme d'onglets. Il peut fonctionner aussi bien en mode Ajax (les blocs sont ainsi chargés de manière asynchrone) ou non-Ajax (les blocs sont tous chargés, puis leur visibilité successive assurée par JavaScript).

► <http://drupal.org/project/quicktabs>

Gérer la visibilité des blocs

Des blocs visibles en fonction du chemin

Le second bloc que nous allons créer est celui de présentation de la société, qui devra être affiché sur le côté droit de la page.

**Figure 9-9**

Ce que nous cherchons à obtenir sur le côté droit

À la différence du bloc précédent, il ne devra s’afficher que sur la page d’accueil et uniquement pour les utilisateurs anonymes. Comme précédemment, nous allons cliquer sur *Ajouter un bloc* et renseigner les informations de contenu. S’agissant d’un bloc à afficher sur le côté, nous allons le doter cette fois d’un titre.

Description du bloc :

Description de la société

Une courte description de votre bloc. Utilisée sur la [vue d'ensemble des blocs](#).

Titre du bloc :

Qui sommes-nous ?

Le titre du bloc tel qu'il est affiché à l'utilisateur

Corps du bloc :

Colchique est une société de conseil en management spécialisée dans les ressources humaines dans le domaine des technologies de l'information...

Figure 9–10
Création du bloc personnalisé
de présentation de la société

Cette fois, nous allons regarder de plus près les options de notre bloc en commençant par la section *Paramètres de visibilité spécifiques à l'utilisateur*.

Paramètres de visibilité spécifiques à l'utilisateur

Paramètres de visibilité personnalisés :

☒ Les utilisateurs ne peuvent pas contrôler s'ils veulent afficher ce bloc ou non.

☐ Afficher ce bloc par défaut mais autoriser l'utilisateur à le masquer.

☐ Masquer ce bloc par défaut, mais autoriser l'utilisateur à l'afficher.

Permettre aux utilisateurs de personnaliser la visibilité de ce bloc dans les paramètres de leur compte.

Figure 9–11
Spécification des paramètres de visibilité
spécifiques à l'utilisateur

Ici, il nous est possible d’offrir aux utilisateurs disposant d’un compte sur le site (par exemple, les contributeurs) la possibilité de décider s’ils veulent que le bloc soit visible ou pas. Si vous rendez ce choix possible, l’utilisateur disposera sur sa page *Mon compte*, d’une série de cases à cocher lui permettant de régler la visibilité de chacun des blocs configuré dans ce sens. Ces réglages lui sont spécifiques et n’influencent pas l’affichage défini en standard pour l’ensemble des utilisateurs ou visiteurs anonymes.

Par défaut, rien n’est sélectionné et cela signifie que le bloc sera affiché dans tous les cas ; cette option convient très bien à notre bloc *Ma société*. Nous aurions cependant pu sélectionner l’une des options suivantes :

- *Les utilisateurs ne peuvent pas contrôler s'ils veulent afficher ce bloc ou non ;*
- *Afficher ce bloc par défaut mais autoriser l'utilisateur à le masquer ;*
- *Masquer ce bloc par défaut, mais autoriser l'utilisateur à l'afficher.*

La section suivante, *Paramètres de visibilité spécifiques aux rôles*, permet d'afficher ou de cacher le bloc en fonction du rôle de l'utilisateur connecté. Dans notre cas, ce bloc ne doit être affiché que pour l'utilisateur anonyme, nous cochons donc *anonymous user*.



Figure 9-12
Définition des paramètres de visibilité par rôles

Le dernier paramétrage possible, *Paramètres de visibilité spécifiques à la page*, est de loin le plus intéressant. Il nous permet de conditionner l'affichage du bloc en fonction du chemin interne de la page en cours. Cette section propose trois choix, suivis d'une zone de texte permettant de définir la condition d'affichage.

- *Afficher le bloc sur toutes les pages exceptées celles listées* : lorsque cette option est utilisée, la zone contient une liste de chemins (un par ligne) pour lesquels le bloc ne sera pas affiché. Le format des chemins est explicité un peu plus loin (paragraphe suivant).
- *Afficher le bloc seulement sur les pages listées* : inversement, le bloc ne sera visible que pour les chemins indiqués dans la zone.
- *Afficher si le code PHP suivant retourne TRUE* : il s'agit d'une option à manier avec beaucoup de précautions, qui permet de saisir dans la zone une fonction PHP qui devra renvoyer **TRUE** pour que le bloc soit visible, ou **FALSE** pour qu'il soit invisible. Un exemple illustrera ce choix un peu plus loin (pages suivantes).

Pour les deux premiers choix, chaque chemin de la liste peut utiliser l'un des formats suivants :

- un chemin simple, par exemple `node/1` pour `http://mon_site/node/1`. Il peut s'agir ici autant d'un chemin interne que d'un alias ;
- un chemin utilisant le caractère `*` qui va correspondre à n'importe quelle chaîne de caractères. Par exemple, `node/*` rendra le bloc visible pour les chemins internes `node/1`, mais aussi `node/2`, `node/12`, `node/1/edit`, et ainsi de suite. Ici aussi les alias sont autorisés ;
- un chemin utilisant plusieurs caractères `*` sur une même ligne. Par exemple `node/*/` fonctionnera aussi bien pour le chemin `node/1/edit` que pour `node/1/print`. Mais il ne fonctionnera pas pour `node/1` tout court car il n'y a qu'un seul niveau et notre motif en demande deux. Encore une fois, les alias peuvent être de la partie ;

- un chemin prédéfini : lorsqu'on désire rendre un bloc visible à la racine, il n'est en effet plus possible d'utiliser la syntaxe précédente (le chemin serait vide car l'adresse complète serait `http://mon_site/`). La solution est alors d'utiliser le chemin prédéfini `<front>` qui permettra d'afficher le bloc lorsque l'utilisateur est sur la page d'accueil.

Grâce à ce système, il est donc possible de positionner des blocs dans n'importe quelle région, et d'afficher des blocs différents d'une section à l'autre de votre site. Dans notre cas, nous n'allons afficher ce bloc Ma société que pour la page d'accueil. Nous cochons donc le second choix et utilisons le chemin prédéfini `<front>`.

Figure 9-13
Le bloc Ma société ne sera affiché que sur la page d'accueil

Ceci fait, enregistrez votre bloc et déplacez ce nouveau bloc dans la région Barre de droite, juste au-dessus du bloc Connexion utilisateur et enregistrez.

Figure 9-14
Positionnement de notre bloc sur la région Barre de droite

Allons à présent sur la page d'accueil pour vérifier que notre bloc n'apparaît effectivement pas en haut à gauche (car nous sommes encore connectés en tant qu'administrateur). Une fois ce point validé, déconnectez-vous et le bloc devrait apparaître.

Figure 9-15
Notre bloc personnalisé en situation

Cliquons ensuite sur notre premier contenu pour quitter la page d'accueil, et constater de ce fait la disparition du bloc. Tout s'est bien passé ? Parfait !

Des blocs visibles par PHP

À présent, pour illustrer le troisième choix de visibilité – le mode PHP –, nous allons créer à titre d'exemple un bloc d'avertissement pour l'administrateur. Ce bloc sera affiché sur toutes les pages dans l'en-tête du site, pour vous rappeler que vous êtes connecté en tant qu'administrateur sur le site, et qu'il faut donc prêter attention à chacune de vos actions.

Commençons par créer un nouveau bloc comme ceci :



Description du bloc : *

Avertissement

Une courte description de votre bloc. Utilisée sur [la vue d'ensemble des blocs](#).

Titre du bloc :

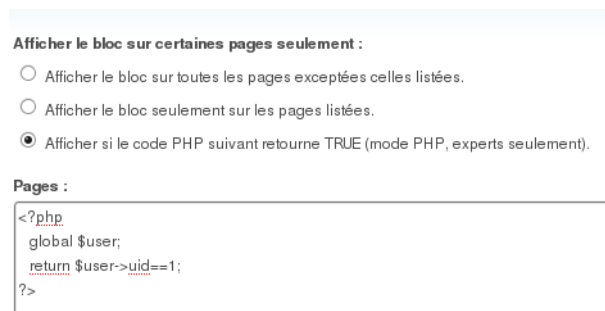
Le titre du bloc tel qu'il est affiché à l'utilisateur

Corps du bloc :

Attention !!! Vous êtes ADMINISTRATEUR

Figure 9–16
Création du bloc personnalisé d'avertissement

Cette fois, dans la section *Paramètres de visibilité spécifiques à la page*, nous allons sélectionner le troisième choix et saisir le code suivant :



Afficher le bloc sur certaines pages seulement :

☐ Afficher le bloc sur toutes les pages exceptées celles listées.

☐ Afficher le bloc seulement sur les pages listées.

☒ Afficher si le code PHP suivant retourne TRUE (mode PHP, experts seulement).

Pages :

```
<?php
global $user;
return $user->uid==1;
?>
```

Figure 9–17
Saisie de la condition PHP d'affichage du bloc

Le code PHP de la zone doit être encadré par les balises `<?php` et `?>` pour être interprété par Drupal comme une fonction pouvant renvoyer `TRUE` (vrai), pour obtenir l'affichage du bloc, ou `FALSE` (faux) lorsqu'il doit rester caché.

Le code de notre fonction s'appuie sur le fonctionnement interne de Drupal et permet de savoir si l'utilisateur qui visualise la page en cours est effectivement l'administrateur ou non. L'objet global `$user` contient toutes les informations sur cet utilisateur. Sachant que l'administrateur est le premier utilisateur créé lors de l'installation de Drupal, il a systématiquement `1` comme `ID`. C'est ici le sens de notre test.

Figure 9–18

Le message d'avertissement en situation



MODULES MAGIQUES

Blocs à visibilité programmée

Lorsqu'un site contient des centaines de blocs, il peut devenir pénible de gérer leur visibilité à la main alors qu'il serait si simple de décrire cette visibilité en code PHP.

C'est le rôle du module Block Visibility Page, qui fournit un service interne pour décrire en mode programme la visibilité de chaque bloc. L'autre avantage de cette approche est qu'il suffit de copier le code en production sans avoir à paramétrer à nouveau la visibilité des blocs.

► <http://drupal.org/project/bpv>

À présent, déconnectez-vous pour vérifier que le bloc a bien disparu, et reconnectez-vous en tant que Marine pour valider le fait qu'il ne s'affiche toujours pas.

La possibilité qu'offre Drupal de saisir ici du code PHP s'accompagne des mêmes préconisations que pour le format d'entrée PHP code. Ne mettez jamais de code long dans cette zone, juste une condition simple avec une référence à une variable ou un appel de fonction. Si vous avez besoin de plus, utilisez une fonction que vous placerez dans un module (voir le chapitre 14, « Écrire son premier module »).

Visibilité des blocs de modules

Nous avons vu comment créer un bloc et gérer sa visibilité. Il convient de noter également que cette possibilité n'est pas réservée qu'aux seuls blocs personnalisés mais qu'elle peut être étendue à l'ensemble des blocs provenant de tout module.

Très souvent, les blocs issus des modules intègrent leur propre logique d'affichage. Le bloc Connexion utilisateur, par exemple, est caché dès que l'utilisateur est identifié. Pourtant, si nous allons sur le lien [configurer](#) de ce bloc, aucune information de rôle n'a été renseignée. C'est le module lui-même qui prend en charge ce comportement.

Cependant, il est possible de définir (ou éventuellement de forcer), à travers la configuration d'un bloc de module, les paramètres de visibilité. Pour prendre un exemple, intéressons-nous au bloc Commentaires récents, dont l'intitulé est suffisamment explicite. Ajoutez ce bloc à la région Barre de droite et enregistrez. Où que vous alliez sur le site, ce bloc restera visible, et ce, quel que soit votre rôle.

À présent, retournons dans l'administration des blocs pour cliquer sur son lien [configurer](#) et changer sa visibilité afin qu'il ne soit présent que sur la page d'accueil (avec l'adresse <front> comme pour le bloc Ma société).

Enregistrez et jugez du résultat. La visibilité est maintenant spécifique et restreinte à la page d'accueil. Nous avons ainsi la possibilité de redéfinir la visibilité de l'ensemble des blocs, et pas seulement de ceux que nous avons nous-même créés.



Figure 9-19
Le bloc des commentaires récents

De manière plus anecdotique mais néanmoins très utile, il est aussi possible de changer le titre d'un bloc de module. Si « Commentaires récents » vous semble trop banal, vous pouvez y remédier en configurant le module et en ajoutant votre touche personnelle.



Figure 9-20
Définition d'un titre personnalisé pour un bloc standard

Ainsi, après enregistrement, le bloc aura un nouveau titre plus original et personnalisé que le simple « Commentaires récents ».



Figure 9-21
Affichage des commentaires récents
avec notre titre personnalisé

Vous pourriez de la même manière forcer le titre dynamique du bloc Navigation pour inscrire le texte de votre choix, ou rendre sa visibilité contextuelle, mais ce n'est pas forcément la meilleure des idées... En vidant la valeur du titre, vous retrouverez le titre par défaut tel que l'a défini le module.

Vous l'aurez compris, le maître mot qui guide le fonctionnement de Drupal est *contrôle*. Tout est modifiable, paramétrable et adaptable à vos besoins, mais ce n'est pas toujours aussi simple et direct. Lorsque vous n'arrivez pas à obtenir l'effet désiré, n'hésitez jamais à demander de l'aide sur les forums (<http://drupalfr.org/forum>).

Gérer les menus

Structure des menus

Dans le vaste panel des blocs disponibles figure une catégorie de blocs qui se révélera particulièrement utile : les blocs de type menu. Depuis le début de notre exploration, nous utilisons sans le savoir l'un d'entre eux : le bloc Navigation, qui contrôle toutes les actions au sein de Drupal.

À l'instar de la taxonomie, les menus sous Drupal fonctionnent suivant une logique d'arborescence : nous distinguons ainsi de grands ensembles appelés *menus* qui contiennent des arborescences de liens appelés *éléments de menu*.

Pour mieux visualiser ce concept, dirigeons-nous vers *Construction du site* puis *Menus* pour voir apparaître les trois menus standards de Drupal :

- le menu *Navigation* que nous avons utilisé intensivement – tous les éléments de ce menu correspondent à des actions liées aux fonctionnalités des modules activés ;
- le menu *Liens primaires* qui par défaut est vide, et qui correspond généralement aux liens que l'on trouve en haut de la page (*contact*, *se déconnecter*, etc.) ;
- le menu *Liens secondaires* qui, lui aussi, est vide par défaut et contient généralement les accès aux grandes sections du site.

Cliquez maintenant sur le menu *Liens primaires*, pour constater que Drupal affiche bien un menu vide d'éléments.

Revenez en arrière et faites de même avec *Liens secondaires* ; vous obtenez le même résultat. Enfin, cliquez sur *Navigation*, et cette fois-ci Drupal devient beaucoup plus loquace :








Elément de menu	Activé	Déplié	Opérations
 Astuces de composition (désactivé)	<input type="checkbox"/>	<input type="checkbox"/>	modifier
 Mon compte	<input checked="" type="checkbox"/>	<input type="checkbox"/>	modifier
 Créer un contenu	<input checked="" type="checkbox"/>	<input type="checkbox"/>	modifier
 Article	<input checked="" type="checkbox"/>	<input type="checkbox"/>	modifier
 Page	<input checked="" type="checkbox"/>	<input type="checkbox"/>	modifier
 Administrer	<input checked="" type="checkbox"/>	<input type="checkbox"/>	modifier
 Construction du site	<input checked="" type="checkbox"/>	<input type="checkbox"/>	modifier

Figure 9–22
Le contenu du menu Navigation

Nous y retrouvons toutes les actions que nous avons abordées jusqu'ici, chacune fournie par un module. Un point intéressant à noter ici est le

fait que toutes les entrées de menu sont affichées sans préoccupation de permissions. En effet, la visibilité d'un élément de menu fait partie intégrante de la gestion assurée par Drupal. Nous pouvons donc ajouter n'importe quel élément à un menu, il ne sera affiché que si l'utilisateur a la permission de lancer l'action correspondante.

Gestion des chemins

Pour mieux saisir la logique de fonctionnement des menus, intéressons-nous au menu *Liens primaires* et définissons les éléments qui le composent, en nous appuyant sur ce qui a été décidé dans notre storyboard, c'est-à-dire en implémentant le lien *se déconnecter*.



Figure 9-23

Ce que nous devons obtenir pour le menu primaire

Pour ce faire, retournez sur la liste des menus, et cliquez sur *Liens primaires* puis sur *Ajouter un élément*. Commençons par renseigner l'information fondamentale : le *chemin interne* du menu. Nous avons déjà abordé cette notion de chemin interne dans le chapitre consacré aux contenus. S'agissant des menus, la nouvelle notion importante à retenir est le fait que leur gestion dans Drupal est totalement déterministe. Il en découle l'obligation pour chaque module de déclarer préalablement au système cœur de Drupal la liste exacte et exhaustive des chemins internes qu'il prend en charge.

Si on examine l'exemple du module User qui effectue toute la gestion des utilisateurs, nous remarquons qu'il déclare (entre autres) deux chemins correspondant à deux actions fondamentales :

- `user/login`, qui va afficher le formulaire de connexion de l'utilisateur ;
- `logout`, qui va déconnecter l'utilisateur courant.

Faites-en l'expérience très simplement en saisissant dans la barre d'adresse de votre navigateur `http://www.colchique-conseil.fr/logout`. Vous êtes alors automatiquement déconnecté. À présent, si vous entrez l'adresse `http://www.colchique-conseil.fr/user/login`, le module User interceptera l'appel à ce chemin et affichera le formulaire de connexion. Vous pourrez ainsi vous reconnecter en tant qu'administrateur.

Certains modules déclarent des chemins plus complexes que ces simples actions. Par exemple, le module Node, qui gère toute la logique des contenus, déclare à Drupal prendre en charge le chemin `node/*`, signifiant « tous les chemins commençant par `node/` ». C'est donc ce module qui

prendra en charge l’affichage des contenus lorsque vous saisissez une adresse de la forme `www.colchique-conseul.fr/node/1`.

Le système de menus de Drupal repose donc sur ce principe et vous oblige à saisir un chemin interne (ou un alias) qui soit valide, c’est-à-dire qui corresponde à un chemin pris en charge par un module installé. Par conséquent, si vous saisissez le chemin `mauvais_chemin`, Drupal détectera le fait qu’aucun module ne le gère et vous affichera un message d’erreur.

Les seules exceptions à cette règle sont bien évidemment les chemins dits *externes*, c’est-à-dire pointant sur d’autres sites web. De tels chemins doivent alors être écrits sous une forme absolue, c’est-à-dire toujours précédé de `http://`, par exemple `http://drupal.org`.

Création d’un élément de menu

Passons à la pratique, et ajoutons le menu de déconnexion en saisissant les informations suivantes.

Chemin :

logout

Le chemin de Drupal vers lequel pointe cet élément de menu. Cela peut être un chemin interne à Entrez `<front>` pour pointer vers la page d'accueil.

Titre du lien dans le menu :

Déconnectez-vous

Texte correspond au nom du lien qui sera affiché dans le menu.

Description :

Cliquez ici pour vous déconnecter

La description affichée quand le pointeur survole cet élément de menu.

Figure 9–24
Définition d’un nouveau menu
« Déconnectez-vous »

- Le *Chemin* que nous utilisons ici est celui fourni par le module User que nous venons d’analyser.
- Le *Titre du lien dans le menu* correspond au libellé de notre élément de menu. Attention, le code XHTML n’est pas autorisé.
- La *Description* est, quant à elle, utilisée comme bulle d’aide sur le lien, il est donc important de la renseigner.
- Le paramètre suivant correspond au fait que l’élément de menu est activé ou non. Nous le laissons logiquement à *Activé*.

Les trois paramètres restants n’ont pour le moment pas d’importance, nous y reviendrons un peu plus loin. Vous pouvez dès lors enregistrer votre élément de menu, action qui renvoie à la liste des éléments du menu *Liens primaires*.

Figure 9–25
Le menu est maintenant enregistré.

Élément de menu	Activé	Déplié	Opérations
 Déconnectez-vous	<input checked="" type="checkbox"/>	<input type="checkbox"/>	modifier supprimer

Notre nouvel élément est en place et vous pouvez comme toujours le *modifier* ou le *supprimer*. Mais la chose intéressante à noter se passe en haut à droite de l'écran, avec l'apparition de notre menu primaire.



Figure 9-26
Voilà notre premier menu en place.

Vous pouvez cliquer sur ce lien (*Déconnectez-vous*) pour constater qu'il est parfaitement opérationnel, mais surtout qu'une fois l'utilisateur déconnecté, il disparaît. Comme nous l'avons mentionné plus haut, un élément de menu est soumis au système de permissions de Drupal. Ainsi, tous les liens qui n'ont pas de sens pour l'utilisateur disparaissent, comme ici une déconnexion pour un visiteur anonyme.

De la même manière, nous pouvons renseigner des sections de notre site cette fois-ci dans le menu *Liens secondaires*. Dans notre storyboard, nous avons prévu trois sections : *Nos publications*, *Les Billets* et *FAQ*.

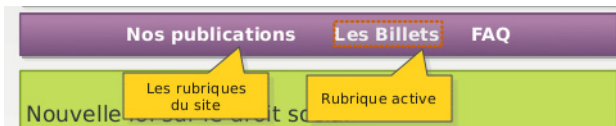


Figure 9-27
Ce que nous désirons obtenir pour les liens secondaires

Nous disposons, grâce au module Views étudié précédemment, des deux listes pour les types de contenu Article et Billet. Il nous est donc possible d'ajouter ces deux sections aux liens secondaires.

Le processus est exactement le même que pour l'ajout d'un élément dans le menu *Liens primaires*, commençant par la sélection du menu *Liens secondaires* > *Ajouter un élément*.

Le chemin que nous devons indiquer est celui que nous avons spécifié pour l'affichage Page de la vue `liste_des_publications`, à savoir `publications`.

Chemin : *

Le chemin de Drupal vers lequel pointe cet élément de menu. Cela peut être un chemin interne à Drupal comme `node/add` ou une url externe comme `http://drupal.org`. Entrez `<front>` pour pointer vers la page d'accueil.

Titre du lien dans le menu : *

Texte correspond au nom du lien qui sera affiché dans le menu.

Description :

La description affichée quand le pointeur survole cet élément de menu.

Figure 9-28
Création de l'élément de menu pour la rubrique Nos publications

Enregistrez et faites de même avec la vue Billets.

Enfin, nous allons ajouter notre dernière section FAQ en utilisant cette fois-ci l'URL associée au chemin du livre FAQ créé précédemment. Lorsque les trois éléments de menu sont enregistrés, nous pouvons en modifier l'ordre pour correspondre à ce qui a été défini dans le storyboard.

Elément de menu	Activé	Déplié	Opérations	
Nos Publications*	<input checked="" type="checkbox"/>	<input type="checkbox"/>	modifier	supprimer
Les Billets*	<input checked="" type="checkbox"/>	<input type="checkbox"/>	modifier	supprimer
FAQ	<input checked="" type="checkbox"/>	<input type="checkbox"/>	modifier	supprimer

Figure 9–29
Toutes nos sections sont en place.

Il ne reste alors plus qu'à enregistrer la configuration pour juger du résultat en haut à droite de la page.



Figure 9–30
Voilà nos sections maintenant visibles.

Comme vous le constatez, il est facile d'ajouter de nouveaux éléments à ces deux menus très spéciaux que sont les liens primaires et secondaires. Voyons à présent comment ajouter un bloc de menu similaire au menu *Navigation*.

Ajouter un bloc de menu

Les trois menus dont nous disposons pour l'instant ne sont que des éléments paramétrés en standard ; il est bien évidemment possible d'en ajouter autant que nécessaire.

Dans notre storyboard, nous avons prévu sur la barre de droite un menu permettant un accès rapide aux différentes sections de la présentation de notre société.



Figure 9–31
Ce que nous désirons obtenir pour les informations sur la société

La première étape consiste cette fois à créer un nouveau menu. Pour cela, retournez dans l'administration des menus et cliquez sur le lien *Ajouter un menu*.

Comme pour le type de contenu ou le vocabulaire, nous allons définir ici le *Nom du menu*, qui ne doit contenir que des lettres en minuscule, des nombres et des tirets, et bien sûr être unique. Ensuite viennent le *Titre* et la *Description* de votre choix.

Nom du menu : *

Le nom machine de ce menu. Ce le des nombres, des tirets, et doit être

Titre : *

Description :

Figure 9–32
Création d'un nouveau menu La société

Il suffit de cliquer sur *Enregistrer* pour que le nouveau menu devienne actif. Vous êtes alors redirigé sur la page indiquant qu'il n'y a aucune entrée dans le menu, ce qui est assez logique.

Arrive maintenant le moment d'ajouter nos éléments de menu. Nous pourrions créer les pages, et ensuite ajouter chacune d'elles en tant qu'élément de menu comme nous l'avons fait précédemment, en utilisant le chemin interne du contenu correspondant (*node/xxx*). Cependant, ce n'est pas – et de loin – la méthode la plus efficace. Cela vous obligerait à afficher à chaque fois le contenu dans un onglet avant de recopier son chemin interne dans la zone *Chemin*, ce qui est pour le moins fastidieux.

Une bien meilleure option consiste à laisser Drupal faire cela tout seul. En effet, jusqu'à présent, nous utilisons le rôle Contributeur pour créer nos contenus, donnant lieu à un formulaire de création limitée par les permissions de ce dernier. Pour créer des contenus de type Page, nous allons utiliser le rôle Administrateur pour la simple et bonne raison que c'est le seul qui dispose de ce droit. L'administrateur n'ayant aucune limitation, il aura ainsi accès à un formulaire de création beaucoup plus complet, comportant notamment une section bien pratique qui permet de définir directement un menu en même temps que la page elle-même.

Voyons cela de plus près en commençant par la création de la page La société. Allez en tant qu'administrateur dans *Créer un contenu*, puis *Page*.

Le début du formulaire est familier et nous pouvons déjà saisir le *Titre* de la page. La nouveauté se trouve en dessous du titre sous la forme d'une section à déplier, *Paramètres du menu*.

Nous pouvons ici définir directement un élément de menu en commençant par son titre (le libellé), suivi d'une zone déroulante très importante qui permet de choisir où le menu sera inséré. Dans cette liste, vous

retrouvez bien sûr tous les menus existants, y compris le menu *La société* que nous venons de créer, mais également tous les sous-éléments de chacun des menus. Il est ainsi possible de rattacher notre nouvel élément de menu à n'importe quel menu ou élément de menu existant.

Titre : *

Qui sommes-nous ?

Paramètres du menu

☐ Supprime cette entrée de menu.

Titre du lien dans le menu :

Qui sommes-nous ?

Ce texte correspond au nom du lien qui sera affiché dans le menu. Laissez vide si vous ne voulez pas créer de lien dans le menu.

Élément parent :

<La société>

La profondeur maximale d'un élément et de tous ses enfants est fixée à 9. Certains liens dans les menus ne pourront pas être utilisés comme parents si leur sélection les fait excéder cette limite.

Poids :

-50

Facultatif. Dans le menu, les éléments au poids plus élevé seront positionnés vers le bas et les éléments au poids plus faible seront positionnés plus haut.

Corps :

B *I* U ABC [Liste à puces] [Liste ordonnée] [Liste à puces] [Liste ordonnée] Styles Format [Liste à puces] [Liste ordonnée] [Liste à puces] [Liste ordonnée]

Colchique est une société de conseil en management spécialisée dans les ressources humaines dans le domaine des technologies de l'information...

Figure 9–33
Ajout de la page Qui sommes-nous ?
en créant l'élément de menu dans la foulée.

Pour l'heure, nous rattacherons le menu de notre page à *<La société>*. Laissez le champ *Poids* tel qu'il est. Il permet d'ordonner les éléments d'un même niveau les uns par rapport aux autres, mais nous verrons un peu plus loin une technique plus efficace pour réaliser la même tâche. Une fois le menu défini, saisissez le *Corps* de la page et enregistrez. Ceci fait, retournez dans l'administration des menus et cliquez sur *La société*. Nous découvrons alors que Drupal a bien créé un nouvel élément de menu actif. Cliquez sur *modifier* pour en découvrir le contenu. Outre les informations déjà saisies plus haut, nous voyons ici que Drupal a bien attribué à ce menu le chemin interne de notre nœud.

Figure 9–34
Le chemin interne du nœud

Chemin : *

node/5

Renouvelez l'opération pour les pages Historique, Chiffres et dates, L'équipe et Nos valeurs, en rattachant à chaque fois la page au menu <La société> sans vous préoccuper ni de l'ordre, ni des niveaux dans l'arborescence.

Ceci fait, retournez dans la liste des éléments du menu *La société* pour juger du résultat :

Élément de menu	Activé	Déplié	Opérations	
+ Qui sommes-nous ?	<input checked="" type="checkbox"/>	<input type="checkbox"/>	modifier	supprimer
+ Les chiffres et dates	<input checked="" type="checkbox"/>	<input type="checkbox"/>	modifier	supprimer
+ Historique	<input checked="" type="checkbox"/>	<input type="checkbox"/>	modifier	supprimer
+ Nos valeurs	<input checked="" type="checkbox"/>	<input type="checkbox"/>	modifier	supprimer
+ L'équipe	<input checked="" type="checkbox"/>	<input type="checkbox"/>	modifier	supprimer

Figure 9–35
Notre menu manque encore d'organisation.

Notre menu est complet mais l'ordre des éléments n'est pas idéal. Arrangeons cela en utilisant les petites croix sur la gauche et en faisant du glisser-déposer pour ordonner les entrées de menu (en glissant une ligne verticalement), mais aussi pour modifier la hiérarchie (en glissant horizontalement).

Une fois l'ordre correctement établi, il ne nous reste qu'à cocher la case *Déplié* de l'élément de menu *Qui sommes-nous ?*. Sans cela, ce menu ne serait en effet pas ouvert à l'affichage et demanderait à l'utilisateur de cliquer à chaque fois sur son lien pour afficher ses sous-éléments. Vous pouvez maintenant *Enregistrer la configuration*.

Élément de menu	Activé	Déplié	Opérations	
+ Qui sommes-nous ?	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	modifier	supprimer
+ Nos valeurs	<input checked="" type="checkbox"/>	<input type="checkbox"/>	modifier	supprimer
+ L'équipe	<input checked="" type="checkbox"/>	<input type="checkbox"/>	modifier	supprimer
+ Historique	<input checked="" type="checkbox"/>	<input type="checkbox"/>	modifier	supprimer
+ Les chiffres clef	<input checked="" type="checkbox"/>	<input type="checkbox"/>	modifier	supprimer

Figure 9–36
Notre menu à présent proprement structuré

Voilà qui est mieux. Il ne nous reste dès lors qu'une seule chose à faire, afficher le menu. Pour cela, retournez dans la gestion des blocs, et si vous regardez bien dans la région *Désactivé*, vous devriez y découvrir un nouveau bloc *La société*.

Désactivé		
+ La société	<aucun>	configurer

Figure 9–37
Un menu est aussi un bloc.

Figure 9–38
Donnons un titre un peu plus explicite
à notre bloc La société.

C'est une fonctionnalité très pratique du système de menu de Drupal : tout nouveau menu créé donne lieu à la création d'un bloc équivalent pour son affichage. Faites maintenant glisser le bloc jusqu'à la région Barre de droite et placez-le juste derrière le bloc Description de la société, puis enregistrez.

Nous allons maintenant modifier la configuration de ce bloc. Tout d'abord, afin qu'il bénéficie d'un titre un peu plus représentatif comme « La société Colchique », vous pouvez soit modifier ici son titre, soit retourner dans la configuration du menu *La société* pour le modifier à cet endroit. Pour plus de simplicité, nous allons changer le titre dans la configuration du bloc.

Titre du bloc :

La société Colchique

Modifiez le titre par défaut du bloc. Indiquez <enc>

Vous devez en outre paramétrer ce bloc pour qu'il ne soit affiché que sur les pages autres que l'accueil.

Figure 9–39
Définition de la visibilité du bloc La société

Afficher le bloc sur certaines pages seulement :

☒ Afficher le bloc sur toutes les pages exceptées celles listées.

☐ Afficher le bloc seulement sur les pages listées.

☐ Afficher si le code PHP suivant retourne TRUE (mode PHP, experts seulement).

Pages :

<front>

Lorsque tout est en place, vous pouvez enregistrer et vérifier que le rendu du bloc vous convient.

Figure 9–40
Le bloc-menu La société en place



Et voilà, notre menu est désormais actif et utilisable.

En résumé

- Les blocs sont des unités de contenu qui peuvent être créés dynamiquement par des modules, ou que l'on peut créer soi-même comme n'importe quel contenu.
- Une région est une zone de la page sur laquelle peuvent être disposés des blocs.
- Les régions sont définies par le thème.
- Un même bloc ne peut être dans deux régions différentes.
- Changer la structure d'une page consiste en particulier à jouer sur la visibilité des blocs qui la composent.
- Un menu Drupal est un groupe contenant des éléments de menu.
- Chaque menu peut être transformé en bloc.

chapitre

10

Activé	Code	Nom anglais	Nom natif	Direction	Par défaut	Poids	Opérations
<input checked="" type="checkbox"/>	en	English	English	De gauche à droite	<input type="radio"/>	<div>0</div>	éditer
<input checked="" type="checkbox"/>	fr	French	Français	De gauche à droite	<input checked="" type="radio"/>	<div>0</div>	éditer

Gérer les langues

Drupal est un produit développé par des personnes venant des quatre coins de la planète. Ce n'est donc pas une surprise de constater que ce CMS dispose d'un support de langues très évolué. Nous aborderons ci-après l'ensemble des dispositifs dont il dispose pour traduire dynamiquement dans une langue donnée ses interfaces, et même ses contenus.

SOMMAIRE

- ▶ Installer de nouvelles langues
- ▶ Traduire les interfaces
- ▶ Créer un site multilingue
- ▶ Amélioration des traductions existantes
- ▶ Fournir un contenu multilingue

Traduction des contenus et traduction des interfaces

La gestion des langues de Drupal se segmente en trois concepts :

- les langages utilisés pour l'interface : il s'agit là des menus, des messages d'erreur ou d'avertissement, et d'une manière générale de tous les textes que fournissent les modules ;
- les langages des contenus : ceci correspond à la possibilité de disposer d'un même contenu en plusieurs langues ;
- la liste des langues prises en charge par le site : que ce soit pour traduire l'interface ou définir un contenu dans une langue donnée, le nom de cette langue doit être préalablement ajouté à cette liste.

De ces trois points, nous pouvons déduire trois possibilités bien évidemment combinables :

- vous désirez traduire certaines expressions encore en anglais dans l'interface. Ce cas est typique lorsque l'on installe de nouveaux modules, ils ne sont pas – loin s'en faut – tous traduits ;
- vous désirez donner à vos visiteurs la possibilité de changer la langue de l'interface (les menus, les messages, etc.) ;
- vous désirez proposer des contenus dans plusieurs langues.

Tout cela est possible avec Drupal, et nous allons voir comment en commençant par explorer l'ajout de nouvelles langues, puis de nouvelles traductions d'interface, pour nous intéresser enfin à la traduction des contenus eux-mêmes.

BON À SAVOIR Qui fait les traductions ?

Le travail de traduction ne fait pas partie du développement de Drupal en lui-même mais est délégué à chaque association de *Drupaliens* dans chaque pays. La qualité de la traduction dans une langue donnée est donc directement liée au nombre de personnes qui prennent bénévolement et régulièrement de leur temps pour l'améliorer. En France, les traductions sont gérées sur le site <http://l10n.drupalfr.org/>. C'est à cet endroit que vous pourrez vous aussi contribuer si vous le souhaitez.

Traduction des interfaces

La liste des langues

Drupal a besoin de connaître la liste exhaustive de toutes les langues qui vous sont nécessaires. Quel que soit l'usage que vous en ferez, si vous avez besoin d'utiliser une nouvelle langue, vous devez avant toute chose l'ajouter à la liste des langues du site.

Pour accéder à la liste des langues, dirigez-vous vers [Configuration du site](#), puis [Toutes les langues](#). Sur cette page sont listées les langues actuellement connues du système : l'anglais et le français.

Activé	Code	Nom anglais	Nom natif	Direction	Par défaut	Poids	Opérations
<input checked="" type="checkbox"/>	en	English	English	De gauche à droite	<input type="radio"/>	0 <input type="text"/>	éditer
<input checked="" type="checkbox"/>	fr	French	Français	De gauche à droite	<input checked="" type="radio"/>	0 <input type="text"/>	éditer

Figure 10–1
Le panneau de sélection
des langues actives

Rien d'étonnant à cela car, comme nous le disions en introduction, l'anglais est la langue maternelle de Drupal. Quant au français, nous l'avons explicitement activé lors de l'installation.

Sur ce panneau, nous avons deux colonnes intéressantes, avec tout d'abord *Activé* qui indique que la langue est utilisable dans le système. La seconde colonne, *Par défaut*, indique quelle langue Drupal utilise par défaut (ici, le français).

Puisque les deux langues sont activées, cela signifie que nous pouvons basculer l'interface de l'anglais vers le français et vice versa. Pour cela, nous allons devoir instruire Drupal sur la manière de choisir une langue ou une autre.

Négociation de la langue

Plusieurs procédés permettent à Drupal de savoir s'il doit afficher des interfaces en français ou en anglais. Il peut s'appuyer sur le nom de domaine du site en utilisant le français pour `http://www.colchique-conseil.fr` et l'anglais pour `http://www.colchique-conseil.com`. Il peut également détecter la langue qui a été définie comme préférence par défaut dans votre navigateur, et l'utiliser (sur Mozilla Firefox, ce choix est défini dans les *Préférences*, onglet *Contenu* et section *Langues*). Enfin, il peut utiliser une méthode fondée sur le chemin et décider, par exemple, que tous les chemins commençant par `/en/` (ex. `http://www.colchique-conseil.fr/en/node/1`) correspondent à la langue anglaise et `/fr/` à la langue française (ex. `http://www.colchique-conseil.fr/fr/node/1`).

Le choix d'un de ces procédés a pour nom la « négociation de la langue », et son paramétrage est accessible en cliquant sur l'onglet *Configurer* de la page où nous nous trouvons.

Négociation de la langue :

- ☒ Aucun.
- ☐ Seulement le préfixe de chemin.
- ☐ Préfixe de chemin avec solution dégradée.
- ☐ Seulement le nom de domaine.

Figure 10–2
Choix du mode de négociation des langues

- Dans le mode *Aucun*, la langue ne change simplement jamais.
- Dans le mode *Seulement le préfixe de chemin*, la langue est déterminée par le début du chemin (le */en/* ou */fr/* mentionnés plus haut) qui est appelé « préfixe ».
- Dans le mode *Préfixe de chemin avec solution dégradée*, la langue est déterminée comme dans le cas précédent, mais si aucun préfixe n'est présent dans le chemin, ce sont les paramètres de l'utilisateur (dans *Mon compte*) qui seront utilisés. Si l'utilisateur est anonyme, c'est la langue du navigateur qui sera utilisée. Enfin, si le navigateur n'a pas de langue définie, c'est la langue par défaut qui sera utilisée.
- Pour finir, dans le mode *Seulement le nom de domaine*, la langue est déterminée par le nom de domaine utilisé.

Choisissons pour notre exemple la négociation *Préfixe de chemin avec solution dégradée* puis enregistrons ces paramètres.

- ☐ Aucun.
- ☐ Seulement le préfixe de chemin.
- ☒ Préfixe de chemin avec solution dégradée.
- ☐ Seulement le nom de domaine.

Figure 10–3
Choix du mode de négociation
par chemin avec solution dégradée

Réglage des préfixes

URL spécifiques

Avant de poursuivre, vous remarquerez que l'adresse de la page dans le navigateur a automatiquement changé.

 <http://www.colchique-conseil.fr/fr/admin/settings/language>

Figure 10–4
L'adresse du site prend en compte
le préfixe de langue.

Drupal a en effet pris en compte notre choix de la stratégie « par préfixe » et a en conséquence inséré le préfixe de la langue française dans le chemin.

Pour voir d'où vient ce */fr/*, retournez sur la liste des langues et cliquez sur le lien *éditer* de la langue française.

Comme vous le voyez, le préfixe de la langue française est bien *fr*. En revanche, si vous effectuez la même opération sur la langue anglaise, vous constaterez que le champ *Préfixe de chemin* est vide. Cela veut dire que lorsque aucun préfixe ne figure dans le chemin – ce qui est la situation par défaut –, il y aura de fortes chances que notre site s'affiche dans la langue de Shakespeare. Pour remédier à cela, inversons la situation en saisissant *en* comme préfixe pour la langue anglaise et en vidant le champ *préfixe* pour la langue française.

Prefixe de chemin :

Code de la langue ou n'importe quel autre chaîne de caractère à identifier dans le chemin d'accès. Si le paramètre de négociation de la langue est vide, le site sera présenté dans cette langue lorsque le paramètre de négociation de la langue est vide. Toute modification de cette valeur peut casser les URLs existantes, et doit donc être faite avec précaution. Cette valeur doit être la même pour toutes les langues. Par exemple, si vous utilisez "fr" comme domaine pour la langue française, vous devez également utiliser "fr" comme domaine pour la langue anglaise.

Domaine de la langue :

URL spécifique à la langue, incluant le protocole. Si le paramètre de négociation de la langue est vide, ce paramètre peut rester vide. Cette valeur doit être la même pour toutes les langues. Par exemple, si vous utilisez "http://fr.example.com" comme domaine pour la langue française, vous devez également utiliser "http://fr.example.com" comme domaine pour la langue anglaise.

Direction : *

☒ De gauche à droite

☐ De droite à gauche

Direction d'écriture du texte.

Figure 10-5
Paramétrage de la langue

Une fois les paramètres enregistrés, il est probable que Drupal vous renvoie une **erreur 404** (la page n'existe pas). Pas de panique, c'est tout à fait normal car le `/fr/` est resté dans l'adresse et ne correspond donc plus à rien. Retournez à l'accueil et revenez sur la page *Toutes les langues*, l'adresse devrait maintenant être correcte.

À noter que si vous avez choisi la stratégie *Seulement le nom de domaine*, c'est dans ce même formulaire que vous aurez saisi chacun des domaines pour chaque langue (ex. `www.colchique-conseil.com` pour l'anglais).

Changer de langue

Pour tester le passage d'une langue à l'autre, nous disposons à présent de deux solutions : soit nous modifions le chemin manuellement en insérant au début le `/en/`, ce qui n'est pas très pratique, soit nous ajoutons un bloc permettant au visiteur de basculer d'une langue à l'autre.

Pour réaliser cette seconde option, vous devez aller dans l'administration des blocs, ajouter à la région *Barre de droite* le bloc *Sélecteur de langue* qui existe en standard, puis enregistrer les blocs.

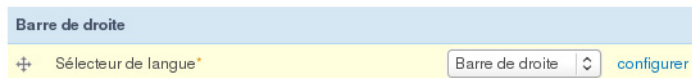


Figure 10-6
Activation du Sélecteur de langue

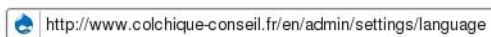
Vous devriez alors voir apparaître un nouveau bloc de gestion des langues.



Figure 10-7
Le Sélecteur de langue

Figure 10–8

L'URL contient le préfixe /en/ signifiant la langue anglaise.



L'interface du site doit être à présent entièrement traduite en langue anglaise.

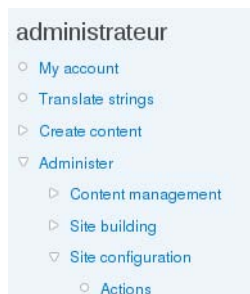


Figure 10–9

Le menu Navigation est maintenant traduit en anglais.

Le choix du mode de négociation *Préfixe de chemin avec solution dégradée* présente ici de nombreux avantages :

- s'il y a un préfixe dans l'URL, le site utilise la langue correspondant à ce préfixe ;
- si aucun préfixe ne figure dans l'URL et que l'utilisateur est authentifié, Drupal utilise la langue choisie dans *Mon compte* pour afficher la bonne interface. Pour un utilisateur anonyme, il utilisera les préférences sélectionnées dans le navigateur web ;
- enfin, sans préfixe dans l'URL et en l'absence de préférences utilisateur détectées, Drupal utilise la langue par défaut – ici le français.

Avec ce paramétrage, notre site fonctionnera comme avant tout en offrant de nouvelles possibilités, dont celle d'ajouter encore une nouvelle langue.

Ajout d'une nouvelle langue

Nous allons à présent ajouter une troisième langue, l'espéranto. Téléchargeons l'archive de cette langue sur <http://drupalfr.org>, dans la section *Downloads* puis *Translations*, comme nous l'avions fait pour le français lors de l'installation de Drupal.

Une fois l'archive en main, décompressez-la dans le dossier *Drupal*. Ensuite, cliquez sur l'onglet *Ajouter une langue* et sélectionnez *Esperanto*. Comme vous le constatez, la liste des langues connues de Drupal est longue, ce qui n'implique pas pour autant l'existence d'une traduction pour chacune d'elles.

Esperanto

By [amuzulo](#) on the 3rd of November, 2004

Drupal interface translation to the international language Esperanto.
Traduko de la interfaco de Drupal en Esperanto.

Version	Date	Links	Status
6.x-1.0	2008-May-31	Download · Release notes	Recommended for 6.x ✓
5.x-1.x-dev	2008-May-31	Download · Release notes	Development snapshot ✗

[Find out more](#) · [Bugs and feature requests](#)

Figure 10–10 Téléchargement de l’archive de langue espéranto

Langue prédéfinie

Nom de la langue :

Esperanto

Sélectionner la langue désirée et cliquez sur le bouton (Ajouter une langue).

Ajouter une langue

Figure 10–11
Ajout de l’espéranto comme langue d’interface

Cliquez maintenant sur [Ajouter une langue](#). Le moteur de traduction de Drupal se met alors en route pour traduire l’interface.

Accueil

Importation de traductions de l’interface utilisateur

Reste 17 sur un total de 25.

Figure 10–12
La barre de progression de traduction de l’interface

Une fois l’interface entièrement traduite, nous sommes renvoyé sur la liste des langues avec un nouvel arrivant, l’espéranto !

Activé	Code	Nom anglais	Nom natif	Direction	Par défaut	Poids	Opérations
<input checked="" type="checkbox"/>	en	English	English	De gauche à droite	<input type="radio"/>	0	éditer
<input checked="" type="checkbox"/>	eo	Esperanto	Esperanto	De gauche à droite	<input type="radio"/>	0	éditer supprimer
<input checked="" type="checkbox"/>	fr	French	Français	De gauche à droite	<input checked="" type="radio"/>	0	éditer

Figure 10–13
L’espéranto est maintenant une langue de Drupal.

Vous remarquerez que cette nouvelle langue a été automatiquement activée. Vous pouvez jeter un coup d’œil sur le bloc de sélection des langues pour vérifier que l’espéranto s’y trouve effectivement, et de là, cliquer sur le lien [Esperanto](#) pour traduire l’interface.

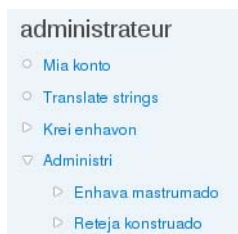


Figure 10-14

Et voici le menu de navigation en espéranto !

Vous noterez que l'URL a pris le préfixe `/eo/`, qui est le préfixe configuré automatiquement pour la langue espéranto. Dans le Sélecteur de langue, cliquez maintenant sur *Français* pour revenir à l'état standard.

MODULES MAGIQUES Réimporter des traductions mises à jour

Pour disposer d'une nouvelle langue, nous avons jusqu'à présent téléchargé une archive de traduction, nous l'avons décompressée à la racine du site et enfin ajoutée à la liste de Drupal. Ceci est parfait mais que se passe-t-il lorsque une traduction a été améliorée et qu'il existe une nouvelle version ? Comment faire pour que Drupal « relise » les traductions ? En standard, rien n'existe pour « rafraîchir » une traduction et il nous faut pour cela installer le module Localization Client.

► http://drupal.org/project/l10n_client

Un fois le module activé, la marche à suivre pour mettre à jour une langue commence évidemment par le téléchargement et la décompression à la racine de Drupal de la nouvelle version des traductions. Si votre outil de décompression demande d'écraser d'anciens fichiers, n'hésitez pas, c'est le but de la manœuvre.

Ceci fait, allez dans *Construction du site* > *Traduction de l'interface*, puis dans l'onglet *Importer* ; vous découvrirez un sous-onglet *Reimport packages*. Dans cet onglet, sélectionnez la langue à mettre à jour puis cliquez sur *Reimport packages*. Une fois le travail achevé, les nouvelles traductions sont prises en compte par Drupal.

Nous pouvons ainsi ajouter à notre site autant de langues que nécessaire, il n'y a aucune limitation. Notez également que nous aurions pu ajouter une langue sans pour autant installer un paquet de traduction. En effet, nous pourrions avoir besoin de l'espéranto pour disposer, par exemple, de la possibilité de traduire des contenus dans cette langue, sans pour autant avoir besoin d'une interface en espéranto. En réalité, cette liste présente toutes les langues disponibles et gérées par Drupal pour votre site ; il vous appartient ensuite de décider si telle langue sera utilisée pour des traductions d'interface ou des traductions de contenus, ou les deux.

Ajout d'une traduction

Jusqu'à présent, nous nous sommes contenté de décompresser des archives de traduction prêtes à l'usage. Voyons maintenant comment ajouter de nouvelles traductions, par exemple comment transcrire en

français une chaîne de caractères qui subsisterait encore seulement en anglais dans l’interface.

Lorsque nous avons abordé la gestion des droits d’accès, nous avons constaté que de nombreuses expressions étaient encore en anglais, comme *create billet content*. Pour traduire ce terme en français, dirigez-vous vers *Construction du Site > Traduction de l’interface*. Vous arrivez alors sur une page indiquant le statut de chacune des langues installées.

Langue	Interface utilisateur
Anglais (interne)	nd
Espéranto	3764/6034 (62.38%)
Français	5488/6034 (90.95%)

Figure 10–15
Statut de traduction par langue

Nous y voyons que 90,95 % des traductions françaises sont réalisées, ce qui n’est pas mal du tout pour un travail exclusivement effectué par des bénévoles. Mais il reste encore 9 % des textes en anglais, ce qui n’est pas un problème en soi car cela signifie que la majorité des instructions visibles par un visiteur ont été traduites. La plupart du temps, ce taux résiduel de textes non traduits est lié à des modules complémentaires ayant été installés et ne disposant pas de traduction française pour leur interface.

Pour traduire l’expression *create billet content*, nous devons cliquer sur l’onglet *Recherche* et la saisir dans le champ *La chaîne contient*.

Avant de valider, nous sélectionnons les options *Toutes les langues*, *Seulement les termes non traduits* et *Tous les groupes de texte*.

La chaîne contient :

Laisser vide pour afficher tous les termes. La recherche est

Langue :

☒ Toutes les langues
☐ Anglais (fourni par Drupal)
☐ Espéranto
☐ Français

Rechercher dans :

☒ A la fois les chaînes traduites et celles qui ne
☐ Seulement les termes traduits
☐ Seulement les termes non traduits

Limiter la recherche à :

☒ Tous les groupes de texte
☐ Interface utilisateur

Figure 10–16
Recherche d’une expression non traduite

BON À SAVOIR

Rechercher des expressions variables

Si vous désirez chercher un terme de manière plus étendue, comme tout ce qui commence par « create » et finit par « contenu », vous pouvez utiliser le caractère %, ce qui nous donnera *create % content*.

Figure 10-17
Résultat de la recherche
de l'expression non traduite

Groupes de texte	Chaîne	Toutes les langues	Opérations
Interface utilisateur	create billet content /admin/user/permissions	eo fr	modifier supprimer

Dans la colonne *Toutes les langues* figurent en barré les langues pour lesquelles il n'existe actuellement pas de traduction de notre expression. Nous allons en ajouter une en cliquant sur *modifier*.

Nous ne saisisons ici – faute de compétences – que la version française de ce terme, puis cliquerons sur *Enregistrer les traductions*.

Modifier le terme

Texte original :
create billet content

Espéranto :

Français :

Enregistrer les traductions

Figure 10-18
Ajout de la traduction manquante

Nous pouvons maintenant vérifier que notre nouvelle traduction a correctement été prise en compte en retournant dans le panneau de configuration des droits d'accès.

administrer les nœuds
Créer un contenu de type billet
create page content

Figure 10-19
Le texte est maintenant correctement traduit.

MODULES MAGIQUES

Un client de traduction dynamique

Le module Localization Client, que nous avons découvert dans le précédent aparté « Modules magiques », a pour rôle premier de vous permettre une traduction dynamique de l'interface. Une fois le module activé, vous disposez, en bas à droite de chaque page, d'un bouton *Translate*. En cliquant sur ce lien, un panneau de traduction de l'ensemble des termes de la page apparaît. Vous pouvez ainsi traduire ou corriger des traductions de manière simple et intuitive. Les modifications sont automatiquement sauvegardées.

► http://drupal.org/project/l10n_client

BON À SAVOIR Utiliser directement les fichiers .PO

Une méthode plus avancée de traduction consiste à utiliser la fonction d'import/export. En exportant une langue, vous obtiendrez un fichier texte au format PO. Ce format est un classique de la traduction dans le monde des logiciels libres. Il existe de très nombreux logiciels sur toutes les plates-formes (par exemple, PoEdit – <http://www.poedit.net>) permettant d'éditer facilement ces fichiers. Une fois les traductions faites, vous pouvez à nouveau importer le fichier PO dans Drupal.

Nous l'avons vu plusieurs fois déjà, les traductions de Drupal sont le travail de bénévoles, et un gros travail qui plus est. Si vous trouvez des erreurs, des manques ou des imprécisions, n'hésitez pas à contribuer. Nous disposons pour cela d'un site français totalement dédié aux traductions.

► <http://l10n.drupalfr.org/>

Traduction des contenus

Comme les interfaces, les contenus peuvent exister en plusieurs langues, et comme pour les traductions d’interface, la première chose à faire pour traduire un contenu dans une langue donnée est d’ajouter cette langue dans la liste *Toutes les langues*. Dans l’exemple qui va suivre, nous proposerons une version anglaise de l’article de bienvenue. Un ajout ne sera donc pas nécessaire puisque la langue figure déjà dans la liste des langues connues de Drupal.

Pour utiliser la traduction des contenus de Drupal, il convient tout d’abord de vérifier que le module Content translation (dans la section *Core – facultatif*) est bien activé. C’est en effet ce module qui va permettre la traduction puis l’affichage d’un nœud dans une langue donnée.

Nous devons ensuite décider quels types de contenu peuvent être traduits, et pour cela, retournons dans le panneau maintenant connu d’édition des types de contenu. Nous allons dans notre exemple modifier le type de contenu Article.

Déployez la section *Procédure de publication* pour y découvrir un nouveau champ *Gestion multilingue* proposant plusieurs choix :

- *Désactivé* est l’état standard, ce type de contenu ne sera pas traduisible ;
- *Activé* va ajouter un champ *langue* à tous les contenus de type Article ;
- *Enabled, with translation* : *oups...* voici un oubli de traduction que vous savez désormais corriger. Cette option a les mêmes effets que *Activé* mais ajoute au contenu une série de fonctions permettant de gérer ses traductions. C’est le choix que nous allons faire.

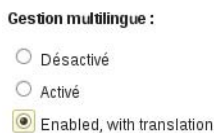


Figure 10-20

Activation du mode multilingue pour les contenus de type Article

Vous pouvez maintenant enregistrer le nouveau paramétrage du type de contenu Article.

Dernière étape, comme toujours lorsque nous ajoutons un nouveau module, nous devons à présent ajouter les droits correspondants. Ici, il s’agit d’octroyer la permission *Translate Content* au rôle Contributeur, sans quoi Marine ne pourrait pas traduire l’article de bienvenue.

Il est maintenant temps de se déconnecter pour se reconnecter en tant que Marine et de modifier l’article de bienvenue.

Figure 10–21
Choix de la langue de l'article

Langue :

Nous remarquons aussi l'apparition d'un nouvel onglet *Translate* (encore un terme en anglais à corriger) sur lequel nous cliquons pour ajouter la traduction anglaise. Apparaît alors une liste de langues indiquant celles pour lesquelles la traduction existe déjà, et pour le moment, nous n'avons évidemment que le français.

Figure 10–22
Le statut des traductions disponibles
pour un même article

Langue	Titre	Statut	Opérations
English	nd	Not translated	add translation
Esperanto	nd	Not translated	add translation
French (source)	Bienvenue	Publié	éditer

L'annotation (*source*) nous indique que cet article est le parent des futures traductions. Nous pouvons alors cliquer sur *add translation* sur la ligne *English*, et créer une version anglaise (ou espéranto si vous vous exprimez couramment dans cette langue).

Nous arrivons alors sur une page très classique de création de contenu avec la langue en grisé, ce qui est logique car Drupal a mémorisé que nous sommes en train de traduire ce nœud en anglais. Vous pouvez maintenant commencer votre version. Pour vous aider dans cette tâche, Drupal préremplit l'introduction et le corps de l'article avec la version source (ici française).

Lorsque vous avez terminé, enregistrez votre travail sans oublier de cocher *Publié* mais pas *Promu en page d'accueil*. Sinon, vous aurez en page d'accueil les deux articles dans les deux langues.

Figure 10–23
Le contenu est maintenant
disponible en anglais.

Welcome

Voir

Modifier

Translate

Le contenu 'Welcome', de type 'Article', a été créé.

ven, 06/19/2009 - 11:18 — Marine

Welcome to Colchique Conseil,

Si nous cliquons à nouveau sur l’onglet *Translate*, nous constatons le travail réalisé.

Langue	Titre	Statut	Opérations
English	Welcome	Non publié	éditer
Esperanto	nd	Not translated	add translation
French (source)	Bienvenue	Publié	éditer

Figure 10–24
Le statut de traduction pour ce contenu indique maintenant la présence de la langue anglaise.

Notre article est maintenant disponible dans deux langues, et nous allons de ce pas vérifier que la traduction des contenus a fonctionné correctement. Pour cela, cliquez sur le lien *Bienvenue*.

Bienvenue

Voir

Modifier

Révisions

Translate

mer, 06/03/2009 - 01:38 — Marine

Bienvenue sur le site de **Colchique Conseil**,

le cabinet de conseil en ressources humaine qui a l'*éthique à cœur*.

Vous trouverez sur notre site une présentation complète de la société, de ses intervenant et de **ses valeurs**. Vous pourrez aussi y découvrir nos publications ainsi que nos billets d'humeur.

Bonne visite

L'équipe de <http://www.colchique-conseil.fr>.

Fichier attaché	Taille
copie d'écran.png	105.4 Ko
Manuel utilisateur.pdf	274.03 Ko

English

Recrutement

Figure 10–25
L'article en mode complet avec le lien permettant de passer à la version anglaise

Vous remarquez sous le contenu en français l'apparition d'un nouveau lien *English*. Ce bouton agit comme un sélecteur de langue et basculera le contenu, mais aussi l'interface, en anglais. Cliquez sur ce lien pour basculer vers la langue anglaise.

Welcome

View

Edit

Translate

Fri, 06/19/2009 - 11:18 — Marine

Welcome to Colchique Conseil,

Figure 10–26
Traduction automatique du contenu en fonction du contexte de langue

L'interface et la langue du contenu ont maintenant changé. Nous remarquons également que les commentaires ont disparu. En effet, les com-

MODULES MAGIQUES

La Rolls de l'internationalisation

Le système de traduction de Drupal convient parfaitement aux usages basiques, mais montre ses limites lorsqu'il s'agit de créer un site entièrement multilingue. En effet, dans un tel cas, vous allez vouloir disposer d'une traduction non seulement pour les contenus et les interfaces, mais aussi pour vos menus personnalisés, vos blocs, vos taxonomies, etc. L'ensemble de ces tâches supplémentaires est réalisé par le module I18N.

► <http://drupal.org/project/i18n>

MODULE MAGIQUE **Une alternative pour une page d'accueil multilingue**

Pour ceux qui ont choisi d'utiliser le module I18N, cette possibilité de définir une page d'accueil n'affichant que les contenus dans la langue de l'interface est intégrée au module. Ceci étant dit, I18N est un gros module qu'il convient de n'utiliser que si l'on en a besoin, le cas échéant, la technique utilisant Views reste la plus efficace.

ASTUCE

L'utilisation de Views n'est justifiée que par le fait de vouloir reproduire en version multilingue le comportement standard de la page d'accueil de Drupal. Vous avez, si vous le désirez, la possibilité de faire exactement la même chose en remplaçant la première étape par la création de deux nœuds de type Page et en utilisant leurs permaliens (`node/XXX`) à la place de `frontpage_en` et `frontpage_fr` pour les deux étapes suivantes.

mentaires sont liés au contenu dans une langue donnée. Avec deux langues, vous aurez donc deux jeux de commentaires.

Page de garde multilingue

Si nous cliquons à présent sur le lien *Home* (anciennement *Accueil*), nous constatons que la page de garde est un mélange d'anglais et de français. Ce n'est pas étonnant car le fonctionnement standard de Drupal est d'afficher en page d'accueil tous les contenus possédant le statut *Promu en page d'accueil*. Voyons comment nous pouvons créer, à partir de ce que nous disposons, deux pages d'accueil, l'une en français et l'autre en anglais, chacune avec une liste de contenus spécifiques à la langue, et surtout comment faire pour que le changement de langue implique le changement de page d'accueil.

La solution pour créer ces deux pages d'accueil commence par l'utilisation du module Views. Il s'agira alors de créer deux vues, la première affichera la liste des nœuds de langue anglaise, et l'autre celle de langue française.

Pour fabriquer ces vues, vous pouvez simplement dupliquer la vue Articles que nous avons précédemment créée, et la renommer `frontpage_fr`. Ceci fait, nous allons commencer par supprimer le filtre sur le *Type de nœud* et le remplacer par un filtre sur la langue du contenu, *Node translation*, avec comme critère *Langue = Français*.

Cliquez ensuite sur l'onglet *Page*, et changez le chemin pour `frontpage_fr`. Changez enfin le titre pour simplement le vider de son contenu. Une fois tout cela fait, enregistrez la vue.

Dupliquez cette nouvelle vue et appelez la cette fois-ci `frontpage_en` ; changez la langue du filtre pour l'anglais et le chemin pour qu'il devienne cette fois `frontpage_en`.

L'étape suivante consiste à ajouter deux alias sur les chemins de nos vues en passant par *Construction du site*, puis *Alias d'URL*. Cliquez sur l'onglet *Ajouter un alias*. Notez un point important : la présence d'un menu déroulant avec une liste de langues, qui nous permettra de rendre notre alias dépendant de la langue en cours.

Pour cela, nous allons créer un premier alias de langue française prenant `frontpage_fr` comme *Chemin système existant* et `frontpage` comme *Alias de chemin*.

Langue :
 Français

A path alias set for a specific language will always be used when displaying this page in that language, and:

Chemin système existant : *
 http://www.colchique-conseil.fr/ frontpage_fr

Spécifier le chemin existant que vous souhaitez renommer par un alias. Par exemple : node/28, forum/1, tax

Alias de chemin : *
 http://www.colchique-conseil.fr/ frontpage

Spécifier un chemin alternatif par lequel cette ressource peut être accédée. Par exemple, taper "about" qu'on ajoute de slash final ou l'alias d'URL ne fonctionnera pas.

Créer un nouvel alias

Figure 10–27
 Saisie de l'alias pour la langue française

Cliquez maintenant sur *Créer un nouvel alias*, et renouvelez l'opération avec cette fois-ci une langue anglaise, un chemin `frontpage_en` et – c'est là toute l'astuce – le même alias `frontpage`. Une fois enregistré, nous disposons donc de deux alias strictement identiques, juste différenciés par la langue.

La dernière étape pour compléter le dispositif consiste à se diriger vers *Configuration du Site > Information*. Nous y trouvons un réglage particulièrement utile à la fin du formulaire : *Page de garde par défaut*. Comme l'intitulé le suggère, ce réglage permet de déterminer quel est le chemin à utiliser lorsque le visiteur cherche à visualiser la page d'accueil. Attention, c'est un peu délicat car il ne s'agit pas de rediriger la page d'accueil vers un chemin mais bien d'utiliser le chemin *comme* page d'accueil. En d'autres termes, l'URL ne changera pas, mais le contenu de la page, si.

Par défaut, la valeur est `node` signifiant que la page d'accueil affiche la liste des nœuds. Changez cette valeur pour la remplacer par notre alias `frontpage`. Ainsi, la page d'accueil utilisera le chemin `frontpage`, un alias dépendant de la langue, comme contenu d'affichage.

Enregistrez le paramétrage et cliquez sur *Accueil*. Vous obtenez bien la vue `frontpage_fr` car nous sommes pour l'instant en interface française. Cliquez maintenant sur le lien *English* du Sélecteur de langue, et l'accueil passe comme prévu en anglais.

En résumé

- Les traductions disponibles sont le fruit d'un travail communautaire, n'hésitez pas à y participer en créant un compte sur <http://10n.drupalfr.org/>.
- Drupal en standard prend en charge une traduction des interfaces et des contenus. Si vous souhaitez aller plus loin, le module I18N est un passage obligé.

chapitre 11

#4120 v-traxxx veh...



GO!



V-Traxxx Vehicle Tracking System, solutions for Vehicle Tracking System with modern and dy ...

more + comments (0)



#3704 wtai



GO!



Music festival in Sweden. This Drupal project is produced by the leading Drupal company in ...

more + comments (0)



#4097 children of ...



GO!



The new site of NGO Children of Africa Foundation. Powered by Drupal 6.x and created by me ...

more + comments (0)

Modifier l'apparence

Comme nous l'avons mentionné dans l'introduction du chapitre « Structurer les pages », le second axe de travail pour personnaliser l'apparence d'un site Drupal passe par l'utilisation de son moteur de thème. Ce moteur d'une grande souplesse permet à notre site de se plier à toutes les contraintes de charte graphique. Mais qui dit grande capacité dit aussi apprentissage, et pour rendre apprentissage le plus progressif possible, nous avancerons graduellement en trois étapes, en partant de la simple configuration d'un thème existant pour arriver à la fabrication d'un thème qui vous est propre et entièrement personnalisé.

SOMMAIRE

- ▶ Comprendre les thèmes, les moteurs de thème et les modèles
- ▶ Paramétrer les thèmes existants
- ▶ Créer son propre thème
- ▶ Construire ses propres modèles
- ▶ Ajouter des fonctions avancées

Qu'est-ce qu'un thème ?

Un thème, ça change tout !

Une manière simple et visuelle pour comprendre ce qu'est un thème Drupal consiste à se rendre sur le site <http://www.drupalsites.net/drupal-top-sites>. Vous avez ici un ensemble de sites web, avec une vignette donnant pour chacun un petit aperçu de son apparence visuelle. Cliquez sur le bouton **GO!** de quelques-unes de ces vignettes au hasard...

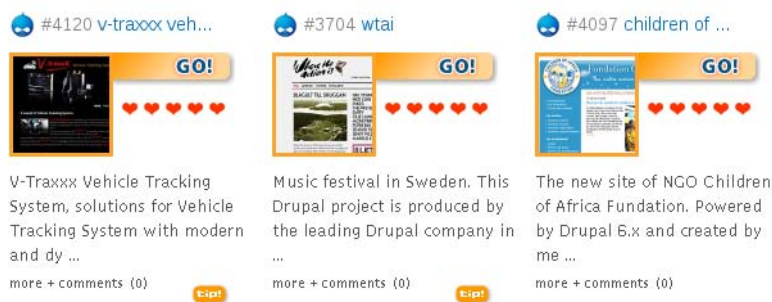


Figure 11-1
Les vignettes des différents sites utilisant Drupal

Au-delà des goûts et des couleurs, pouvez-vous imaginer qu'il s'agisse à chaque fois du même Drupal ? Et pourtant, la construction de l'ensemble de ces sites a suivi un processus semblable à celui que nous avons adopté jusqu'ici ; la seule différence réside dans la personnalisation du thème.

Pour se donner une meilleure idée de ce qu'il est possible d'obtenir avec un même Drupal, voici quelques autres sites :

- <http://research.yahoo.com/>
- <http://www.mediapart.fr/>
- <http://myplay.com/>
- <http://www.laquadrature.net/>
- <http://www.observer.com/>

Thème et moteur de thème

Le système de thèmes de Drupal se compose de deux éléments : un thème et un moteur de thème.

Un moteur de thème est un élément qu'il est possible d'installer dans Drupal à la manière d'un module, mis à part que le seul fait de le décompresser dans le dossier `themes/engines` l'activera spontanément. Le premier rôle du moteur de thème est de permettre une séparation entre données et présentation des données, en fournissant au thème toutes les

informations dont celui-ci a besoin pour construire une page (titre du site, contenus, etc.). En retour, l'autre rôle du moteur est de comprendre la syntaxe utilisée par le thème pour fournir au cœur de Drupal une page en XHTML correctement formée. On comprendra alors aisément qu'un thème développé pour un moteur ne fonctionnera qu'avec ce moteur.

Un thème est un élément qui s'installe exactement comme un module, mais avec une interface d'administration différente. Il est composé de ressources (feuilles de style, images, etc.) et d'un ensemble de fichiers permettant de générer les pages, et il utilise les données ainsi que la syntaxe imposées par le moteur de thème.

Dans la pratique, Drupal est fourni avec un moteur de thème standard appelé PHPTemplate qui utilise la syntaxe PHP. Une large majorité des thèmes que vous pourrez télécharger sur drupal.org utilisent ce moteur, aussi c'est lui que nous étudierons par la suite dans ce chapitre.

Développeur, Administrateur et Thèmeur

Avant de poursuivre sur l'installation d'un nouveau thème, prenons le temps d'examiner quelques points concernant l'organisation et les différents rôles d'une équipe chargée de mettre en œuvre un site sous Drupal.

Drupal peut être adapté à vos besoins selon trois axes qui correspondent à trois types d'utilisateurs avancés différents :

- le développeur PHP/Drupal dont le rôle est d'étendre les fonctionnalités de Drupal. Il construira ainsi de nouveaux modules pour adapter Drupal au plus près des besoins du projet ;
- l'infographiste (ou « thèmeur » dans le vocabulaire drupalien) qui développera l'identité visuelle et l'ergonomie du site. Il sera amené à créer un ou plusieurs nouveaux thèmes pour le site ;
- l'administrateur qui gèrera l'ajout de nouvelles fonctionnalités. Son utilisation de Drupal consiste à installer de nouveaux modules ou de nouveaux thèmes, et de les paramétrer.

Le danger classique lors de la réalisation d'un gros projet avec Drupal est de mélanger ces rôles, soit parce que vous ne disposez pas de toutes les compétences dans votre équipe, soit parce que la frontière entre ces trois domaines est mal tracée. Le résultat est, par exemple, l'obtention d'un site contenant des dizaines de modules récupérés sur drupal.org qui seraient trop évolués ou mal adaptés au besoin, et tout cela par manque de développeur compétent. Ou alors de voir apparaître des fonctions de traitement de l'information propres aux modules se retrouvant dans le thème, parce que cette fois nous aurions un infographiste compétent sous Drupal mais toujours pas de développeur.

Bien évidemment, cette problématique va croissante avec la taille du projet, et devient particulièrement critique pour un projet visant à construire un site à fort trafic.

Pour bien comprendre et respecter les frontières entre les trois domaines de compétence évoqués, il est possible de se fonder sur des règles simples.

- N'activez que les modules que vous utilisez complètement et dont vous maîtrisez le fonctionnement. Méfiez-vous des usines à gaz qui font la pluie et le beau temps, alors que vous n'auriez besoin que de trois gouttes d'eau. Tout est libre dans Drupal, ne vous privez donc pas d'extraire une fonctionnalité d'un gros module pour la *coller* dans un module que vous maîtrisez. Votre seule obligation sera de reverser le code d'un tel *nouveau* module à la communauté, car tous les projets disponibles sur drupal.org sont sous licence GPL v2.
- Un thème ne doit jamais contenir une requête SQL. Si vous rencontrez une situation où il est nécessaire de faire appel à ce genre de solutions, cela signifie que vous avez besoin d'un module spécifique, qu'il soit développé par vos soins ou obtenu via la communauté Drupal. Toutefois, une solution dégradée peut consister à utiliser le fichier de thème `template.php` que nous découvrirons dans ce chapitre.
- Interdisez-vous de manière stricte d'utiliser la fonctionnalité Drupal permettant de mettre du code PHP dans les contenus, que ce soit des blocs ou des nœuds. Tout code ajouté de cette manière sera très difficile à maintenir car il sera stocké dans la base de données, et non pas dans un fichier PHP comme il se doit. Tracer des anomalies éventuelles dans ces circonstances – surtout lorsque les auteurs ne sont plus sur le projet – sera extrêmement consommateur de temps, et donc d'argent. L'ajout de code PHP ne peut servir que pour la gestion de la visibilité des blocs ; veillez alors dans ce cas à ne jamais dépasser une seule ligne de code (typiquement un « return vrai/faux »), et ne faites jamais de requêtes SQL. Là aussi, une solution dégradée consisterait à créer une fonction dans le fichier de thème `template.php` que vous pourrez appeler dans le champ de visibilité du bloc.

Ces avertissements sont le fruit d'expériences multiples et ne sont pas limités à Drupal. D'un contexte à l'autre, les pièges de tout projet informatique restent bien souvent les mêmes :

- les compromis que vous faites aujourd'hui par manque de temps seront payés au centuple dans la suite de votre projet ;
- il est illusoire de croire que des mauvais choix sont rattrapables. Une fois les mauvaises directions prises, il sera toujours compliqué et surtout coûteux de revenir en arrière ;

- l'économie faite d'un développeur compétent en début de projet se paiera plus tard et beaucoup plus cher, lorsque vous aurez besoin de faire évoluer votre site ;
- lorsqu'on démarre un projet informatique, 30 % du temps prévu pour le développement doit être alloué à la conception, c'est-à-dire au travail sur le papier. Ne minimiser jamais cette étape qui est seule garante des bons choix ;
- et n'oubliez jamais que le coût final complet d'un projet informatique se décompose statistiquement en 20 % pour le développement et la mise en œuvre, et 80 % pour la maintenance et l'exploitation au quotidien. À vous de deviner où se trouvent les *vrais* gisements d'économie.

Où allons-nous ?

À partir de maintenant, nous allons sortir du champ théorique pour prendre la casquette du deuxième des trois personnages que nous venons d'évoquer : le thèmeur. Ce qui suit sera par conséquent plus technique que ce que nous avons pu aborder jusqu'à présent.

Pour une bonne compréhension, il est donc nécessaire de disposer d'un bagage en XHTML, CSS et PHP, qui forment à eux trois les compétences de base du thèmeur.

Nous allons commencer par voir ce que sont concrètement des thèmes pour Drupal, et comment les installer, les activer et les paramétrer.

Ensuite, nous allons créer notre propre thème en nous aidant d'un thème existant fourni dans Drupal et voir ainsi qu'avec un minimum de connaissances en CSS, PHP et XHTML, nous pouvons déjà obtenir une apparence radicalement différente.

Enfin, nous aborderons des techniques plus avancées du thème pour entrer ainsi dans les rouages de PHPTemplate.

Gérer les thèmes

L'administration des thèmes

Drupal dispose d'une série de six thèmes standards que vous pouvez découvrir en vous rendant dans l'administration des thèmes ([Administrer > Construction du site > Thèmes](#)).

Le thème activé par défaut que nous utilisons donc depuis le début est Garland. Comme pour les modules, vous pouvez activer plusieurs thèmes en même temps. La différence réside ici dans le fait qu'un seul thème peut être utilisé *Par défaut*.

Figure 11–2
Le panneau de configuration des thèmes

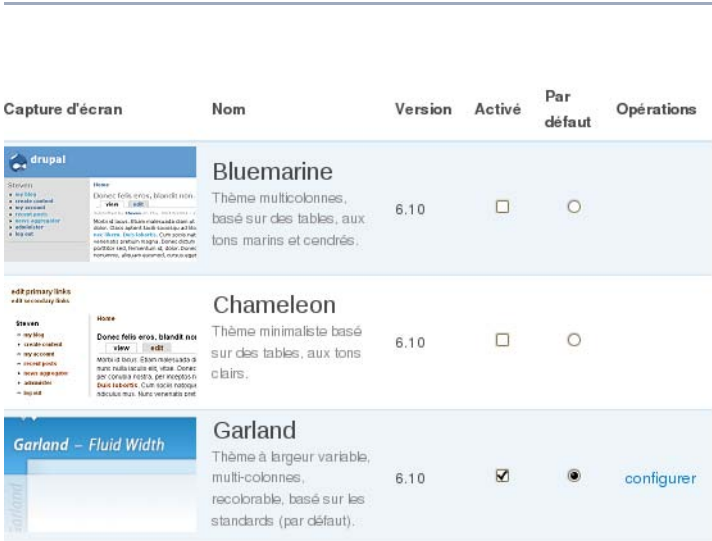


Figure 11–3
Sélection du thème Bluemarine



Figure 11–4
Notre site avec le thème Bluemarine



Paramétrer son thème

Un thème n'est pas qu'une simple apparence, il s'agit d'un simili-module disposant de ses caractéristiques propres, et donc d'une capacité de configuration. Pour étudier cela, revenons dans l'administration des thèmes et sélectionnons à nouveau le thème Garland comme thème par défaut avant d'enregistrer ce changement.

Comme vous le voyez, chaque thème est doté d'un lien [configurer](#) qui apparaît lorsqu'il est activé et qui correspond au paramétrage spécifique du thème. En haut de la page et à droite de l'onglet [Liste](#), vous apercevez un autre lien [configurer](#) correspondant cette fois-ci au paramétrage pour l'ensemble des thèmes.

Nous nous intéresserons au paramétrage spécifique un peu plus tard ; concentrons-nous pour l'instant sur le paramétrage global en cliquant sur l'onglet [Configurer](#).

Nous trouvons dans cette page un ensemble de réglages destinés à modifier l'apparence des thèmes. En premier lieu vient une série de cases à cocher contrôlant l'affichage de certaines informations.

- **Logo** : il s'agit de l'image qui se trouve en haut à gauche de la page. Par défaut, c'est la mascotte de Drupal. Activez cette option qui est importante pour l'identité visuelle de notre site, nous verrons un peu plus loin comment changer ce logo.
- **Nom du site** : c'est le texte « Colchique » qui se trouve à la droite du logo. Il paraît judicieux de le laisser activé.
- **Slogan du site** : c'est un message court qui s'affichera dans l'en-tête du site. Le slogan est celui que nous avons paramétré au chapitre 6, via [Configuration du site > Informations](#). À laisser activé ou à activer le cas échéant.
- **Objectif du site** : dans le même esprit que le slogan et paramétrable au même endroit, c'est une phrase qui s'affichera entre l'en-tête de la page et le contenu.
- **Portrait des utilisateurs dans les contributions** et **Portrait des utilisateurs dans les commentaires** : ce sont là deux options qui permettent au thème d'afficher le portrait que chaque utilisateur aura paramétré dans sa section [Mon compte](#), et ceci que ce soit pour un article ou pour un commentaire. Décochez cette option pour les articles car ils sont presque tous écrits par Marine, et cochez-la pour les commentaires.
- **Champ de recherche** : le champ de recherche permet aux utilisateurs de retrouver facilement des contenus à travers une zone de saisie qui s'affichera généralement en haut à droite de la page. Cochez cette option (il faut aussi que le module Search soit activé !).

BON À SAVOIR Qu'est-ce qu'un favicon ?

Un favicon est une icône qui s'affiche à côté de l'URL d'un site, dans la barre d'adresse du navigateur, ou à côté du titre d'un site dans les marques-pages.

À l'origine, c'est une idée de Microsoft Internet Explorer qui imposait que cette ressource soit nommée `favicon.ico` et soit au format `.ico` de Microsoft.

Aujourd'hui, tous les navigateurs le prennent en charge, le nom du fichier est libre et l'image peut aussi bien être un PNG qu'un JPG. La taille du favicon n'est plus imposée, le navigateur la redimensionnera selon ses besoins, généralement de 16 × 16 à 32 × 32 pixels. Enfin, il est possible d'utiliser de la vraie transparence (PNG) dans de telles images.

Figure 11-5
Paramétrage du logo du site

- *Icône de raccourci* : plus souvent appelée *favicon*, il s'agit de l'icône du site qui apparaîtra dans la barre d'adresse du navigateur ou dans ses marques-pages. C'est généralement une bonne idée de laisser cette option cochée.
- *Liens primaires* et *Liens secondaires* : il s'agit des deux menus que nous avons déjà abordés lors des chapitres précédents, et nous pourrions ici les rendre invisibles. Bien entendu, laissez ces deux options cochées.

Dans la section suivante, dans *Afficher l'auteur et la date de la contribution pour les types*, nous pouvons spécifier quels types de contenu doivent disposer d'une ligne indiquant son auteur et sa date de rédaction. Dans notre cas, ce genre d'information n'a d'intérêt que pour les contenus de type Article et Billet. Nous pouvons donc le décocher pour tous les autres types.

Plus loin apparaît la section *Paramètres du logo*. Nous pouvons grâce à elle remplacer la mascotte de Drupal par l'image de notre choix. Pour cela, il faut commencer par décocher l'option *Utiliser le logo par défaut*. Ensuite, deux possibilités s'offrent à nous :

- soit indiquer le chemin vers un logo personnalisé (une option que nous détaillerons un peu plus loin) ;
- soit télécharger directement un logo comme nous l'avons fait pour le portrait d'un utilisateur. C'est l'option que nous prendrons pour l'instant. Pour cela, laissez le champ *Chemin vers le logo personnalisé* vide et sélectionnez une image de votre choix avec le bouton *Parcourir*. L'image sera enregistrée sur le serveur lors de l'enregistrement du formulaire.

Paramètres du logo

Si cette case est cochée, le logo suivant sera affiché.

☒ Utiliser le logo par défaut

Cocher si vous voulez que le thème utilise son logo par défaut.

Chemin vers le logo personnalisé :

Chemin vers le logo que vous voulez utiliser à la place du logo par défaut.

Envoyer le logo :

Parcourir...

Si vous n'avez pas un accès direct aux fichiers sur le serveur, utilisez ce champ pour transférer votre logo.

La section suivante concerne le favicon, et fonctionne suivant la même logique que celle traitant du logo. Si vous disposez déjà d'une telle icône, vous pouvez procéder de la même manière pour la sélectionner.

Le paramétrage de base étant maintenant terminé, nous pouvons sauvegarder pour juger du résultat.



Figure 11–6 Un sympathique logo, n'est-ce pas ?

Nous voyons ici que le logo a changé, que le slogan est apparu à droite du titre du site et qu'un champ de recherche est à présent visible. De même, l'avatar de Marine est également visible dans ses commentaires.



Figure 11–7
Les portraits des utilisateurs enfin en action

Paramétrages par thème

Paramétrages spécifiques d'un thème

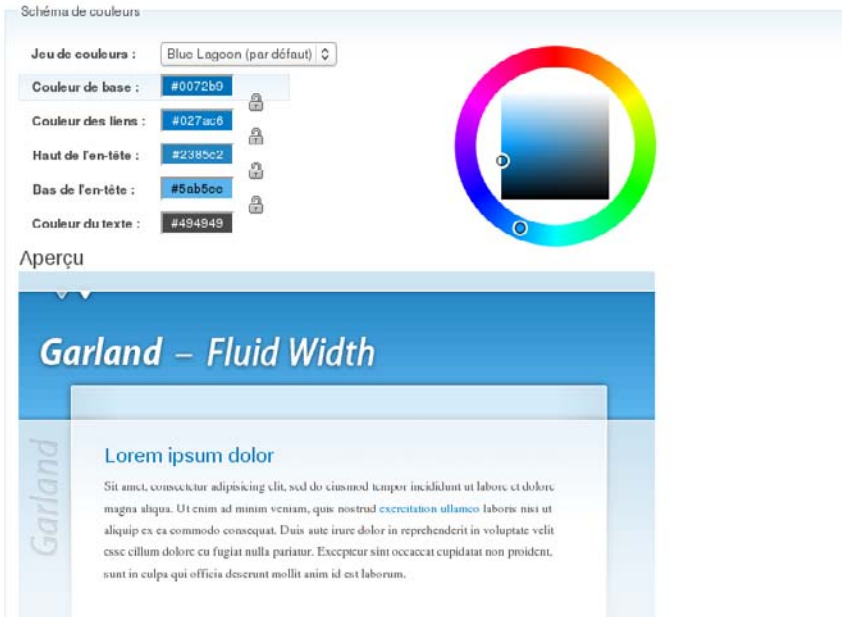
Avant de poursuivre, il est important de retenir que tous ces réglages sont soumis à l'interprétation de l'auteur du thème. Par exemple, le slogan qui est à droite du titre dans le thème Garland sera en dessous du titre avec le thème Marvin. De même, les liens primaires peuvent changer de place en fonction des choix du créateur du thème.

Par ailleurs, certains thèmes plus évolués comme Garland disposent de possibilités de paramétrage supplémentaires au *tronc commun* de Drupal. Nous trouvons donc ici l'usage du lien [configurer](#) spécifique à chaque thème, dont nous avons constaté la présence précédemment. Pour bien comprendre les possibilités qu'il offre, retournons sur le panneau d'administration des thèmes et cliquons sur le lien [configurer](#) de Garland.

Sur la page qui s'affiche, nous constatons tout d'abord que la fin du formulaire contient les mêmes informations que la configuration générale. Nous en déduisons qu'il est possible de redéfinir pour chacun des thèmes les valeurs par défaut que nous avons précédemment choisies.

Nous constatons aussi que le début du formulaire contient une nouvelle et imposante section [Schéma de couleurs](#). Il s'agit là d'un paramétrage propre à Garland que vous ne retrouverez pas sur tous les thèmes.

Figure 11–8
La sélection des couleurs du thème Garland



Grâce à cette nouvelle section, nous pourrions modifier l’ambiance complète de Garland. La liste *Jeu de couleurs* vous permet de choisir un profil prédéfini de coloris, et les champs suivants de régler spécifiquement chaque teinte à votre convenance. Lorsque la sélection de couleurs vous convient, il suffit de sauvegarder pour juger du résultat.

Figure 11–9
Le thème Garland en mode Olive



Paramétrage des blocs par thème

Pour un thème donné, d'autres possibilités de paramétrage existent et s'avèrent plus épineuses à manipuler : elles concernent notamment les régions, notions que nous avons étudiées au chapitre précédent, et qui se présentent différemment suivant le thème sélectionné.

En effet, si nous retrouvons des régions classiques comme Barre de gauche et Barre de droite quels que soient les thèmes, elle n'ont pas forcément le même nom et leur liste n'est pas toujours la même d'un thème à l'autre.

Par exemple, si vous activez le thème Marvin, vous constaterez en allant dans la page d'administration des blocs que vous ne disposez pas de région En-tête ni de Pied de page. En activant ce thème, vous serez ainsi sans doute surpris de voir notre ligne d'informations légales et notre avertissement à l'administrateur apparaître brutalement sur la barre de gauche.



Figure 11–10
Les mouvements de blocs avec le thème Marvin

Il n'y a cependant rien d'étonnant à cela : ayant un bloc activé pour une région qui n'existe pas, Drupal a choisi de le déplacer sur la première région disponible.

Il faut donc partir du principe que chaque thème dispose de ses propres régions. Nous pouvons confirmer ce fait en allant dans le panneau d'administration des blocs.

Bloc	Région	Opérations
Barre de gauche		
⊕ Avertissement	Barre de gauche	configurer supprimer
⊕ Pied de page du site	Barre de gauche	configurer supprimer
Barre de droite		
⊕ Sélecteur de langue	Barre de droite	configurer

Figure 11–11
Le thème Marvin a un nombre très restreint de régions.

Figure 11-12
La disposition des blocs peut être spécifique à un thème.

MODULES MAGIQUES

Faire un site pour mobiles ? Très simple !

Cette notion de paramétrage spécifique des blocs peut être très facilement mise en application si vous désirez rendre votre site compatible avec les unités mobiles (téléphones Android, iPhone et iPod, Blackberries, etc.). Il vous suffit pour cela de disposer de deux thèmes, l'un pour les ordinateurs de bureau, l'autre pour les petits écrans, et de configurer une absence de blocs de par et d'autre du contenu pour le thème Mobile. Ensuite, vous n'avez plus qu'à installer le module Mobile Tools. Une fois activé, ce module vous permet de sélectionner un thème différent du thème par défaut, soit pour une URL donnée (par exemple, <http://mobile.colchique-conseil.fr>), soit pour la même URL mais en identifiant le client comme étant un mobile.

► http://drupal.org/projet/mobile_tools

Avec ce thème, nous n'avons que deux régions, ce qui explique la migration de nos deux blocs.

La localisation variable des régions selon le thème peut s'avérer très vite gênante, si vous avez, par exemple, plusieurs thèmes pour votre site dont vous laissez le choix d'activation à vos utilisateurs. Pour répondre à cette problématique, vous avez fort heureusement la possibilité de définir la disposition de blocs pour chacun de ces thèmes.

Par défaut, lorsque vous arrivez sur l'administration des blocs, vous travaillez sur la configuration de bloc du thème courant. Nous avons donc depuis le début travaillé sur la configuration du thème Garland. En activant un nouveau thème, et surtout en le positionnant comme thème par défaut, nous avons implicitement recopié la configuration de bloc de Garland vers Bluemarine, avant de le faire pour Marvin avec le déplacement de certains blocs comme nous l'avons vu.

Il est aussi possible de modifier chacun des thèmes actifs sans avoir à les mettre successivement par défaut. Pour cela, vous noterez que sous les onglets *Liste* et *Ajouter un bloc* se trouve une liste de sous-onglets, un par thème activé.



Vous pouvez donc cliquer sur un autre thème, et ainsi retrouver l'apparence et la configuration de bloc du thème sélectionné pour en faire le paramétrage. Il est également possible de désactiver certains blocs par thème, une fonction qui complète très bien le paramétrage de visibilité de chaque bloc.

Installer un nouveau thème

Après avoir bien exploré les thèmes fournis en standard avec Drupal, il est maintenant grand temps d'en installer de nouveaux. Cette opération s'effectue de la même manière que pour un module.

Il nous faut tout d'abord choisir un thème sur le site drupal.org, notre point d'entrée étant toujours la page des téléchargements de ressources, <http://drupal.org/project>. Une fois arrivé sur cette page, cliquez sur *Themes*.

Vous pouvez y choisir un thème en ayant une idée du résultat final grâce à l'aperçu qui l'accompagne. Ici, nous avons par exemple le thème *Acquia Marina*.

Acquia Marina

By **jwolf** on the 1st of October, 2008

Acquia Marina is an advanced theme developed by **TopNotchThemes** in partnership with **Acquia** for their commercially-supported Drupal distribution. It contains the same kinds of features you'll find in our other **Drupal themes**.

Features

- 1, 2, or 3 column layout
- **15 collapsible block regions**
- **Drop-down** primary links menu (optional)
- Advanced **theme settings** to customize without coding!
- Includes **icons** for core and Views blocks
- High **attention to detail** on things like forms, lists, comments, quotes, and Drupal maintenance page
- **Cross-browser** tested in IE6/7, Opera, Safari, and Firefox
- **Layered Photoshop PSD files now available!**

This theme is also available in **Acquia Drupal**.

Show your support for this theme, get additional help, show off your site, and discuss with other users – join our group!

Version	Date	Links	Status
6.x-1.5	2008-Dec-22	Download · Release notes	Recommended for 6.x

[Find out more](#) · [Bugs and feature requests](#)



Figure 11–13
La page de projet
du thème Acquia Marina

Une fois le thème sélectionné, vérifiez qu’il existe bien pour la version de Drupal que vous utilisez : regardez dans la zone des téléchargements du thème et identifiez une version, par exemple **6.x** pour un Drupal 6. Comme pour les modules, privilégiez les versions stables en vert. L’utilisation d’un thème non stable (en rouge) est cependant beaucoup moins risquée que celle d’un module.

Lorsque vous disposez de la bonne version, cliquez sur le lien **Download** pour télécharger l’archive du thème au format **tar.gz**.

Comme pour les modules, nous allons décompresser notre thème dans le dossier **sites**, mais cette fois-ci dans le sous-dossier **sites/all/themes/contrib**.

Ceci fait, vous pouvez retourner dans l’administration des thèmes où vous devriez voir apparaître votre ajout.



Figure 11–14
Activation du thème Acquia Marina

Il ne reste alors plus qu'à [activer](#) ce thème, le sélectionner *Par défaut* et *Enregistrer la configuration* pour faire disparaître le thème Marvin.

Pour observer le rendu de manière plus complète que sur la page de configuration des thèmes, retournons à la page d'accueil.



Figure 11-15 Notre site paré du très sympathique thème Acquia Marina

Nous constatons encore une fois que nos blocs ne sont plus à la même place. Cette fois-ci, cela s'explique par le fait que dans ce thème, les régions pour les colonnes n'ont pas les mêmes noms que dans le thème Garland. Nos blocs de droite ont donc été déplacés vers la région *sidebar first*. Comme ce thème ne dispose pas d'une véritable colonne de droite, nous allons les y laisser.

Enfin, comme pour Garland, ce thème dispose de paramétrages spécifiques très étoffés que vous pouvez explorer.

À vous maintenant de faire votre marché et de trouver le thème qui vous convient le mieux. Et si aucun ne vous satisfait, nous verrons bientôt comment créer le vôtre.

Changer les pages d'erreur

Avant de nous lancer dans le développement d'un thème personnel, voyons d'abord comment régler les derniers aspects visuels qu'il nous est encore possible de modifier via la simple administration.

L'un de ces aspects visuels correspond aux pages d'erreur que Drupal affiche lorsqu'un document est introuvable (erreur 404) ou quand un document n'est pas accessible avec le rôle de l'utilisateur (erreur 403).

BON À SAVOIR 404/403, qu'est-ce donc ?

À chaque fois qu'un navigateur affiche une page, il effectue une requête sur un service HTTP. Le service analyse la requête et fournit une réponse au navigateur. Dans cette réponse se trouve un code de statut qui permet de savoir si tout s'est bien passé ou pas.

Le statut renvoyé est sur trois chiffres, le premier indiquant la catégorie dans laquelle se trouve la réponse :

- **1XX** : ce statut indique une information ;
- **2XX** : ce statut indique un succès. Ainsi, toute page renvoyée avec succès est accompagnée du statut **200 OK** ;
- **3XX** : ce statut indique que le navigateur doit aller chercher ailleurs sa réponse (redirection). Ainsi, le statut **301 Moved Permanently** indique que la page a changé d'adresse et que la nouvelle adresse est plus loin dans la réponse ;
- **4XX** : ce statut indique qu'il y a eu un problème. **404 Not Found** correspond à une page qui n'existe pas, **403 Forbidden** indique quant lui que le navigateur n'a pas le droit de la visualiser ;
- **5XX** : ce statut indique aussi un problème mais généralement beaucoup plus grave, avec le classique **500 Internet Server Error** qui correspond généralement à un *plantage* sur le service HTTP lui-même. C'est typiquement ce que renverra Apache si vous vous trompez dans la syntaxe d'un fichier `.htaccess`.

Lorsqu'une telle erreur est détectée, Drupal fabrique une page générique ressemblant à ceci :

Page non trouvée

La page demandée n'a pu être trouvée.

Figure 11-16

Le message standard de Drupal pour une page non trouvée

Si ces pages telles qu'elles sont générées par Drupal ne vous conviennent pas, vous avez la possibilité d'y substituer n'importe quel contenu, de préférence de type Page. Pour tester cela, créons un contenu Page avec un message spécifique pour une erreur 404.

Figure 11-17
Création d'une page 404 personnalisée

Un fois cette page enregistrée, notez le permalien – ici [node/17](#) –, et faites ensuite de même pour une page d'erreur « accès interdit », qui aura alors comme permalien [node/18](#).

Ceci étant fait, allez dans la configuration du site, puis dans *Rapports d'erreur*. Vous avez là deux champs dans lesquels vous pouvez saisir les deux alias avant d'enregistrer l'ensemble.

Figure 11-18
Changement des chemins pour les pages d'erreur 403 et 404

MODULE MAGIQUE

Des blocs sur les pages d'erreur

Vous aurez remarqué que les pages d'erreur 403 et 404 sont dépouillées de leur bloc, et ceci avant tout pour des raisons de performance. Si vous avez cependant besoin de blocs dans ces pages, vous pouvez utiliser le module 404 Blocks. [▶ http://drupal.org/project/blocks404](http://drupal.org/project/blocks404)

Figure 11-19
Notre nouveau message d'erreur

Pour vérifier que cela fonctionne, saisissez un chemin erroné – par exemple http://www.colchique-conseil.fr/pas_bon.

Régler le thème d'administration

Si vous avez pour objectif de concevoir ou de faire concevoir un thème très graphique, il se peut que celui-ci se révèle inutilisable d'un point de vue ergonomique pour l'administration du site, à moins d'y investir du temps et/ou un budget conséquent pour le rendre facilement exploitable.

C'est la raison pour laquelle Drupal vous propose la possibilité de définir un thème spécifique pour la section *Administration*. Dans ce cas, dès que

vous entrerez dans le menu *Administrer*, vous constaterez que votre thème est remplacé par un autre plus exploitable, typiquement Garland. Le choix de ce thème se fait en allant dans la *Configuration du site*, puis *Thème d'administration*.

Thème de l'administration

Thème de l'administration :

Garland

Choisi le thème que les pages d'administration utilisent. Si vous choisissez "Thème par défaut du système"

☒ Utiliser le thème d'administration pour les formulaires de modification du contenu
Utiliser le thème d'administration lors de la modification de contenus existants ou lors de leur création.

Enregistrer la configuration

Rétablir par défaut

Figure 11–20

Réglage du thème d'administration

Ici, vous pouvez non seulement changer le thème à utiliser (Garland reste un bon choix), mais imposer également ce thème lors de l'édition d'un contenu, ce qui est là aussi une bonne idée.

Notez qu'il y a une exception à ce réglage : la page d'administration des blocs qui, comme nous l'avons vu, utilise le thème que l'on cherche à paramétrer.

MODULES MAGIQUES Des thèmes spécialisés pour l'administration

Il existe sur drupal.org de nombreux thèmes dédiés à l'administration. Citons dans cette catégorie le magnifique RootCandy déclinable aux coloris de votre choix (comme Garland).

► <http://drupal.org/project/rootcandy>

Créer son propre thème

Notre premier niveau d'apprentissage des thèmes de Drupal est achevé et nous savons maintenant les télécharger, les installer et les paramétrer. Mais il est rare que nous puissions nous arrêter là. Chaque site web a sa propre personnalité et le thème en est le reflet. Nous allons donc à présent apprendre à créer notre premier thème, de manière douce, c'est-à-dire en partant d'un thème existant que nous adapterons progressivement à nos besoins.

Qu'est-ce qu'un modèle ?

À son origine, PHPTemplate n'était qu'un moteur de thème parmi d'autres et il n'a été ajouté à l'installation standard de Drupal qu'à partir de la version 4.7. À la version 5.0, il a été intégré encore plus profondément dans le cœur du CMS. Aujourd'hui, même s'il est toujours possible d'utiliser d'autres moteurs pour des besoins spécifiques (de compatibilité avec d'autres CMS, par exemple), PHPTemplate reste le passage obligé d'un thème sous Drupal.

Le succès de PHPTemplate (qu'on peut traduire littéralement par « modèle PHP ») tient avant tout à sa souplesse et à sa rapidité de mise en œuvre. En effet, l'approche minimale requiert seulement quatre

fichiers que l'on appelle des modèles (ou *templates* en anglais) pour transformer l'apparence de tout un site.

Les modèles sont des fichiers portant l'extension `.tpl.php` et stockés à la racine du thème. Leur rôle est de produire le rendu visuel soit d'un élément spécifique de la page (un commentaire, un bloc ou un nœud), soit de la page elle-même. Leur contenu est un mélange de code XHTML pour la mise en forme, et de code PHP pour l'insertion des données à présenter. Ainsi, lorsque PHPTemplate utilise un modèle, il en résulte la production d'un fragment de code XHTML. Les balises de ce code disposeront de classes et d'identifiants CSS définis par le thèmeur pour permettre à une ou plusieurs feuilles de style incluses dans le thème d'opérer la présentation finale (couleurs, images, bordures, etc.). Enfin, des comportements écrits en JavaScript peuvent venir compléter l'apparence et l'ergonomie résultant des modèles (onglets dynamiques, menus déroulants, effets, etc.).

PHPTemplate fournit les données au modèle par l'intermédiaire de variables PHP. Chaque modèle dispose ainsi d'un jeu de variables spécifiques au travail à réaliser. Le thèmeur n'a plus qu'à utiliser ces variables grâce à du code PHP inséré au sein de sa mise en forme XHTML.

À titre d'exemple, le modèle qui permet de mettre en forme un commentaire s'appelle `comment.tpl.php`. Pour chaque commentaire à formater, PHPTemplate lui fournit un jeu de variables comme `$content`, qui contient le texte du commentaire à formater, ou encore `$submitted`, qui contient le nom de l'auteur et la date d'envoi de ce commentaire. Une fois réalisé, le modèle pourrait avoir la forme suivante.

Un exemple de modèle pour les commentaires

```
<div class="commentaire">
  <div class="auteur"><?php print $submitted; ?></div>
  <div class="content">
    <?php print $content; ?>
  </div>
</div>
```

Le jeu d'instructions PHP utilisables dans un modèle n'est pas limité, et c'est là que réside le danger de cette approche. Il est en effet possible d'insérer dans ce type de modèle un code PHP complexe, avec par exemple des requêtes à la base de données. Cette pratique est fortement déconseillée. Si vous ressentez la nécessité d'écrire un code complexe, la raison probable est que PHPTemplate ne vous a pas fourni la variable dont vous aviez besoin. Dans ce cas, reportez-vous à la fin du chapitre pour découvrir la bonne manière d'étendre un modèle en y ajoutant vos propres variables.

Le code PHP d'un modèle doit rester extrêmement simple et ne contenir que deux types d'instruction : `print` pour insérer le contenu d'une variable, et éventuellement `if/else` pour implémenter des conditions.

Attention aussi à la forme courte de la balise PHP `<?>` qui n'est pas reconnue par de nombreux serveurs. Utilisez toujours sa version longue `<?php`. Pour les mêmes raisons, utilisez toujours `<?php print $variable ?>` et évitez à tout prix sa version courte `<?= $variable ?>`. À elles seules, ces deux règles vous éviteront énormément de problèmes lors d'une mise en production.

L'imbrication des modèles

Comme nous l'avons dit plus haut, quatre modèles au minimum permettent de mettre en forme l'ensemble d'un site Drupal : `node.tpl.php`, `comment.tpl.php`, `block.tpl.php` et `page.tpl.php`. Pour mieux comprendre comment ces quatre modèles fonctionnent, prenons comme exemple un thème simple comme Bluemarine.

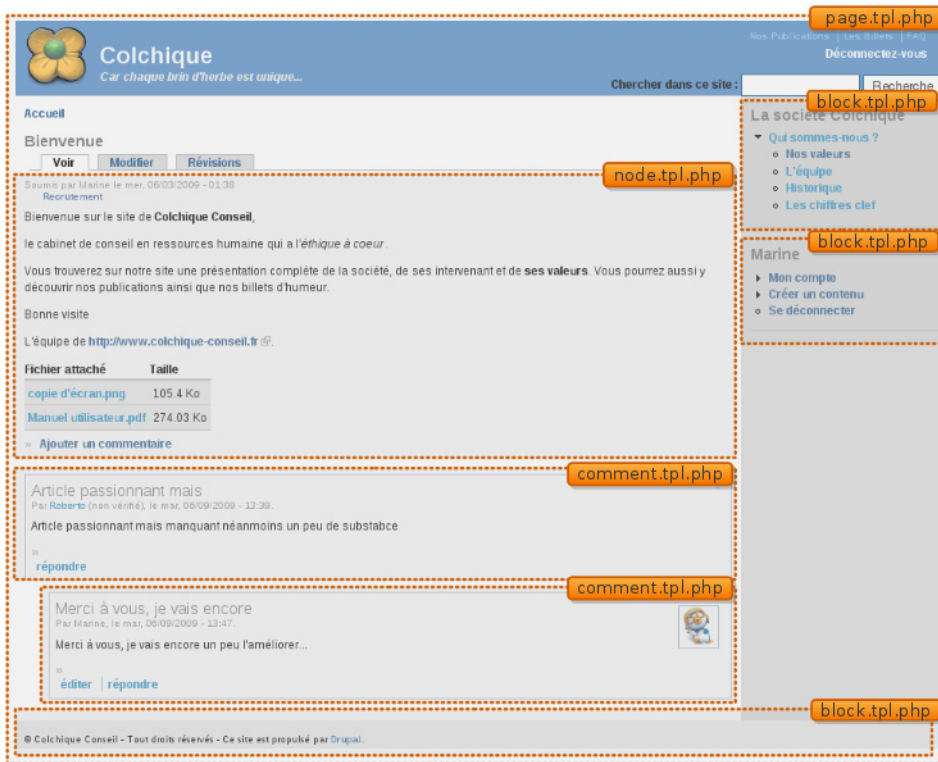


Figure 11-21
La responsabilité des modèles dans la construction d'une page

Pour construire cette page, PHPTemplate utilise les quatre modèles en les imbriquant les uns dans les autres.

- En tout premier lieu est utilisé le modèle `node.tpl.php`. PHPTemplate lui fournit par l'intermédiaire d'un jeu de variables spécifiques tout le contenu du nœud à mettre en forme (titre, corps, etc.). Il récupère ainsi une version du contenu mise en forme en XHTML.
- PHPTemplate effectue la même opération, mais cette fois avec chacun des commentaires du nœud en utilisant le modèle `comment.tpl.php`. En retour, PHPTemplate dispose d'une liste de commentaires formatés en XHTML qu'il ajoute au rendu XHTML du nœud.
- Ensuite, PHPTemplate va procéder au rendu XHTML de chacune des régions du thème. Il constitue pour cela une liste de blocs visibles par région pour la page et utilise pour chacun d'eux le modèle `block.tpl.php`.
- Dernière étape, PHPTemplate fournit au modèle de la page, `page.tpl.php`, le rendu XHTML du nœud et de ses commentaires, et celui de chacune des régions. Le modèle de page lui renvoie alors un document XHTML complet et prêt à être renvoyé par Drupal au navigateur web.

Le modèle Page est, vous l'aurez compris, le garant de la structure visuelle du site. C'est en effet lui qui détermine la disposition de chacune des régions dans le document XHTML final.

Création d'un thème personnalisable

Pour passer de la théorie à la pratique, le plus simple est de démarrer en effectuant des modifications à partir d'un thème existant, par exemple le très basique thème Bluemarine disponible dans le dossier `themes/bluemarine`.

Commençons par regarder ce que nous trouvons dans ce dossier :

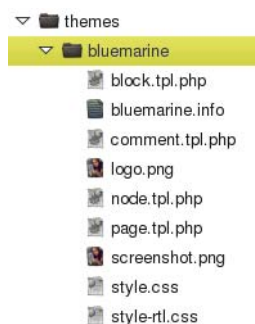


Figure 11-22
L'arborescence et le contenu d'un thème

Comme vous le voyez, un thème est finalement un simple dossier contenant les types de fichiers suivants :

- des modèles portant l'extension `.tpl.php` et présents à la racine du thème, chargés de produire la mise en forme XHTML des données fournies par PHPTemplate à chacun d'eux. Comme nous l'avons vu précédemment, ils sont au nombre minimum de quatre : `block.tpl.php`, `node.tpl.php`, `comment.tpl.php` et `page.tpl.php` ;
- un ou plusieurs fichiers CSS pour styler la présentation XHTML. Par défaut, PHPTemplate utilise pour cela le fichier `style.css` à la racine du thème. Le fichier `style-rtl.css` est quant à lui utilisé en plus de `style.css` lorsque Drupal doit afficher une langue dont l'écriture se fait de droite à gauche (`rtl` signifiant *right to left*) ;
- éventuellement, un ou plusieurs fichiers JavaScript pour ajouter une couche comportementale. Par défaut, PHPTemplate utilise, s'il existe, le fichier `script.js` à la racine du thème. Pour ce thème très simple, nous n'avons aucun comportement JavaScript à ajouter ;
- des ressources graphiques (images, polices, etc.) généralement utilisées par les feuilles de style ;
- un fichier portant le nom du thème (qui est aussi le nom du dossier du thème) et l'extension `.info`, contenant toutes les informations descriptives de ce thème.

La première étape consiste à dupliquer ce thème, qui se trouve dans le dossier `themes/bluemarine`, vers un nouveau dossier `sites/all/themes/mes_themes`, et à le renommer en `mon_theme`. Attention, ne commettez jamais l'erreur de modifier directement un thème (ou n'importe quel autre fichier standard) de Drupal. Vous vous exposeriez alors à de sérieux maux de crâne lorsque viendra le temps de mettre à jour votre installation.

Ensuite, afin d'éviter toute confusion avec le thème d'origine, renommez le fichier `bluemarine.info` en `mon_theme.info`, puis ouvrez le fichier avec l'éditeur de texte de votre choix (gedit, Notepad, etc.) pour en modifier le contenu.

Comme nous le disions plus haut, le fichier `.info` est la carte d'identité du thème. Sa syntaxe se compose d'une succession de couples `variable = valeur` avec éventuellement des lignes de commentaire précédées du caractère `;`.

```
2; $Id: bluemarine.info,v 1.4 2007/06/08 05:50:57 dries Exp $
3name = Bluemarine
4description = Table-based multi-column theme with a marine and ash color scheme.
5version = VERSION
6core = 6.x
7engine = phptemplate
8
9; Information added by drupal.org packaging script on 2009-05-13
10version = "6.12"
11project = "drupal"
12timestamp = "1242243950"
13
```

THÈMES MAGIQUES Les starter kits

Le thème Bluemarine a le mérite d'être fourni en standard avec Drupal. C'est pour cette raison que nous allons l'utiliser comme base de notre apprentissage pour la personnalisation d'un thème.

Pour la conception d'un thème réel, il existe de nombreux thèmes appelés « *starters* » (thèmes de démarrage). Ces thèmes n'ont aucune identité visuelle particulière et sont conçus dans le but unique d'être adaptés à vos besoins. Une liste à peu près exhaustive de ces thèmes de démarrage est disponible à cette adresse :

► http://www.chapterthree.com/blog/squiggy_rubio/review_drupal_6_starter_themes

Notez que le thème Zen, souvent cité, est loin d'être le plus simple à utiliser. Pour débiter, préférez-lui un thème plus simple comme Basic.

► <http://drupal.org/project/basic>

Figure 11-23
Extrait du fichier `mon_theme.info`

Commencez par supprimer tout ce qui se trouve après la ligne 8, et prenons chaque entrée une à une.

- Le champ `name` définit le nom en clair du thème. Notez que le nom Drupal du thème est celui donné par le fichier `.info`. Nous saisissons donc ici un nom facilement lisible comme `Mon Thème`, tandis que Drupal utilisera quant à lui le nom interne `mon_theme`.
- Le champ `description` est un texte d'aide destiné à être affiché dans le panneau de configuration des thèmes. Saisissez par exemple `Ma version modifiée du thème BlueMarine pour le site Colchique-Conseil`.
- Le champ `version` est mal renseigné dans le thème d'origine, nous allons changer cela en remplaçant la valeur `VERSION` par `1.0`. C'est le numéro de version de notre thème, qui a une valeur consultative et informative.
- Le champ `core` est lui plus important car il indique dans quelle version de Drupal ce thème est censé fonctionner. Avec `6.x`, nous indiquons que ce thème fonctionnera avec tous les Drupal version 6. En effet, Drupal fonctionne pour les thèmes suivant la même logique que pour les modules, et ne cherche jamais à garder une compatibilité ascendante. Ce paramètre permet donc d'éviter d'utiliser des thèmes non compatibles avec la version de Drupal mise en œuvre.
- Le champ `engine` nous ramène à la conversation précédente. Ce champ permet de définir quel moteur de thème est utilisable pour ce thème. Ici il s'agit donc bien de PHPTemplate.

Une fois modifié, le fichier `mon_theme.info` doit ressembler à ceci :

Fichier `mon_theme.info`

```
name = Mon Thème
description = Ma version modifiée du thème BlueMarine pour le
site Colchique-Conseil
version = 1.0
core = 6.x
engine = phptemplate
```

Après l'avoir enregistré, retournons dans la configuration des thèmes pour l'activer.



Figure 11–24
Activation de notre premier thème

Voilà, c'est aussi simple que cela. Nous disposons à présent d'un thème bien à nous que nous pourrions personnaliser sans risque pour le reste du système.

Premier niveau de personnalisation : styles et scripts

Utiliser les feuilles de style

Comme nous l'avons vu, les modèles d'un thème utilisent un mélange de code PHP pour insérer les données, et de code XHTML pour leur mise en page. Les balises XHTML sont, quant à elles, dotées de classes et d'identifiants CSS permettant d'en modifier l'apparence sur le navigateur. Le premier degré de modification de notre thème consistera donc à exploiter les styles CSS pour changer autant de choses que possible.

Par défaut, PHPTemplate détecte automatiquement la présence dans le dossier du thème de la feuille de style `style.css`. De manière générale, vous n'avez jamais besoin de lier vous-même une feuille de style à une page. Ces tâches sont prises en charge directement par PHPTemplate.

S'agissant d'un thème qui n'a pas été écrit par nos soins, il sera difficile de savoir quels identifiants ou quelles classes CSS ont été utilisés pour modifier le style d'une balise. Pour pallier cette méconnaissance, nous allons utiliser un outil absolument vital, une extension pour Mozilla Firefox nommée Firebug. Firebug permet, entre autres choses, de découvrir très facilement les identifiants CSS en cliquant simplement sur un élément de la page.

Une fois Firebug installé et Firefox redémarré, vous devriez voir apparaître une icône en forme d'insecte dans la barre de statut du navigateur (voir ci-contre). Cliquez dessus pour faire apparaître le panneau Firebug.

Dans la barre d'outils de ce panneau se trouve un bouton *Inspector* que nous ne cesserons d'utiliser dans les travaux à venir. Une fois que l'on a cliqué dessus, il permet d'afficher dynamiquement le code HTML de l'élément que votre souris survole, ainsi que les sections des feuilles de style qui en déterminent l'apparence.

Si nous voulons modifier la couleur de fond de l'en-tête, nous apprenons ainsi que cette zone est gérée par une table qui a pour identifiant `#header`, et que cet identifiant est défini dans la feuille de style `style.css` à la ligne 80 avec un fond bleu (`background-color: #6699CC`).

Nous allons donc modifier le fichier `style.css` pour changer cette couleur bleue en un fond image de type dégradé. Pour créer ce fond, nous utiliserons un outil d'édition d'images comme Gimp, Inkscape, ou encore Adobe Photoshop.

URL Firebug

► <https://addons.mozilla.org/fr/firefox/addon/1843>



Figure 11-25
Icône de Firebug

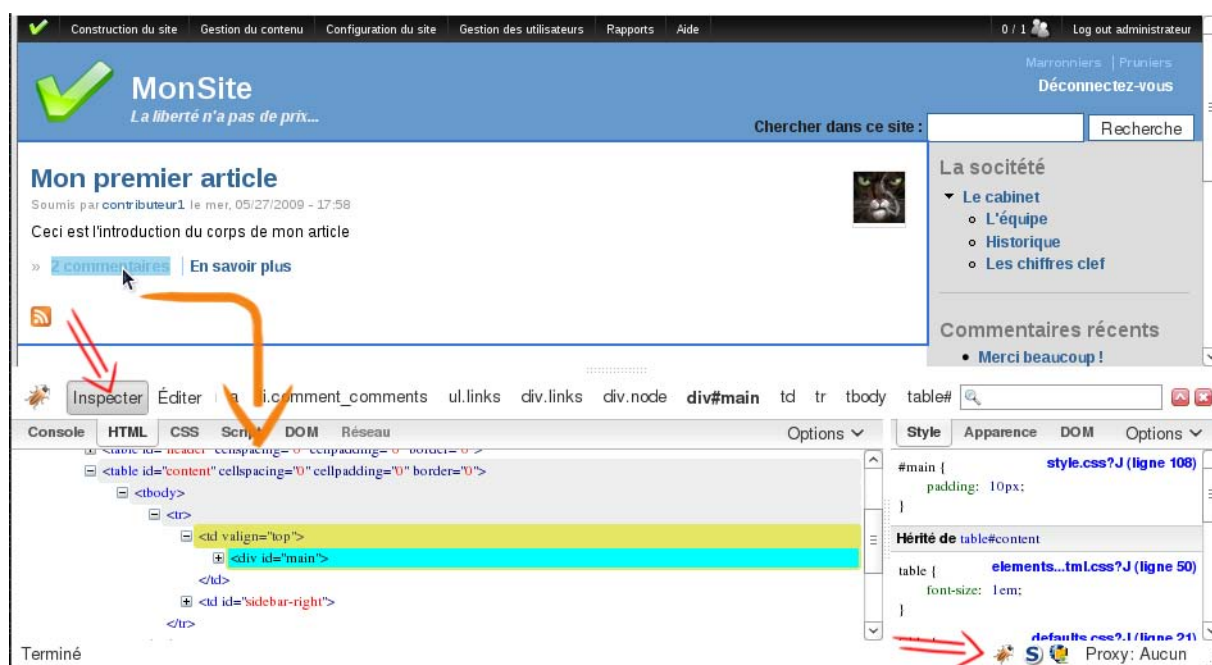


Figure 11–26 Aperçu du panneau Firebug en action

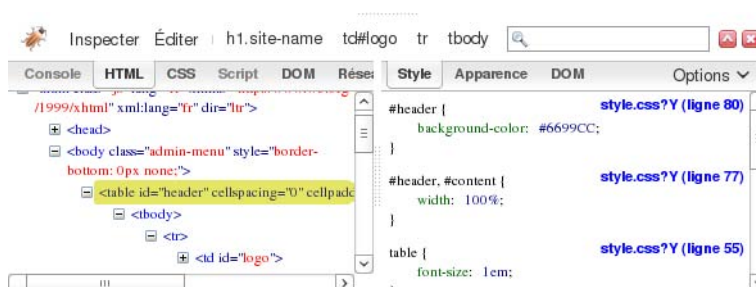


Figure 11–27
L'en-tête vue par Firebug

Une fois cette image générée, elle devra être placée dans un dossier `mon_theme/images/fond_header.png`. À nous de modifier en conséquence la feuille de style `style.css` à la position indiquée par Firebug :

Portion de `style.css` à modifier pour le thème de l'en-tête

```
#header {
  background-color: black;
  background-image: url("images/fond_header.png");
  height: 79px;
}
```

Après un rafraîchissement de la page sur le navigateur, nous pouvons juger du résultat.

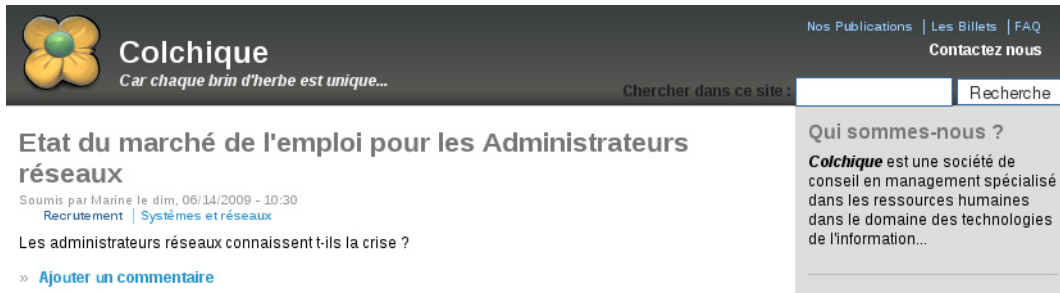


Figure 11-28
Notre thème
avec son en-tête

Ce n'est là qu'un premier pas. Avec un peu de travail sur le fichier `style.css`, il est possible de changer radicalement l'apparence de la page sans écrire une seule ligne de PHP.

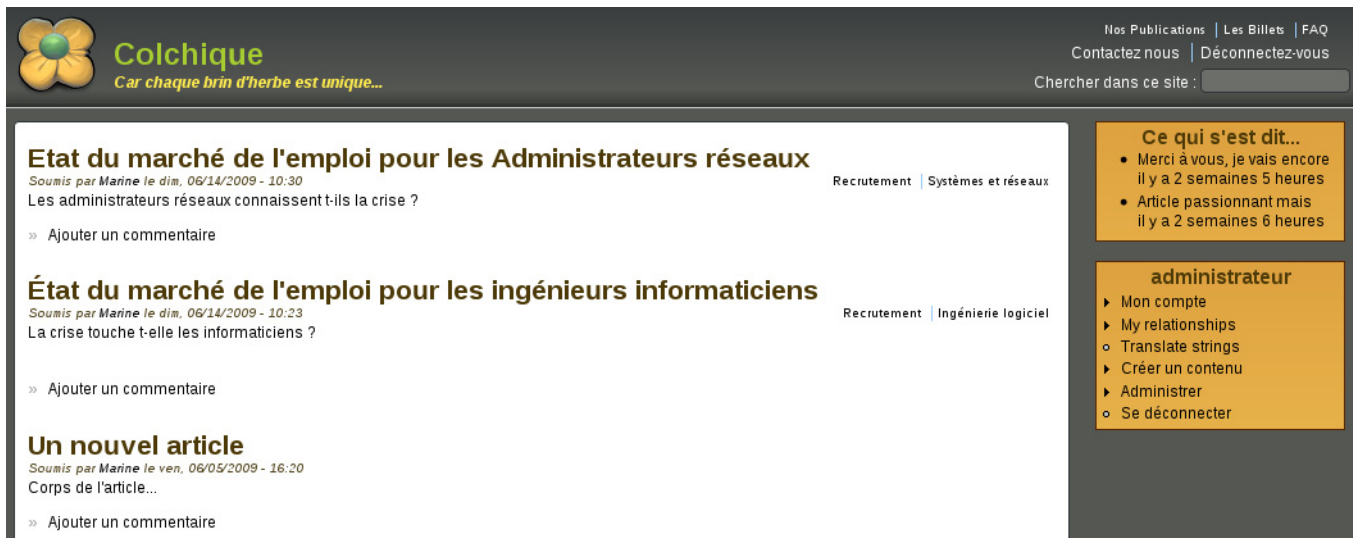


Figure 11-29 Et maintenant un peu plus travaillé

Pour mieux organiser les balises CSS, vous allez rapidement avoir besoin de disposer de plusieurs feuilles de style. Il suffit pour cela de modifier le fichier `mon_theme.info` et d'y ajouter un nouveau champ `stylesheets[all][]=ma_feuille.css`. Mais attention, en procédant ainsi, `style.css` ne sera plus reconnue comme feuille de style par défaut. Pour la conserver, il est nécessaire d'ajouter un champ supplémentaire `stylesheets[all][]=style.css`.

Ajout de nouvelles feuilles de style dans `mon_theme.js`

```
; les feuilles de style spécifiques
stylesheets[all][] = style.css
stylesheets[all][] = ma_feuille.css
stylesheets[print][] = ma_feuille_pour_impression.css
```

L'identifiant `all` signifie « pour tous les médias » et implique que le navigateur utilisera la feuille de style en toutes circonstances. Il est possible de remplacer `all` par n'importe quel média CSS. Par exemple, pour une feuille à utiliser uniquement pour l'impression, nous utiliserions `stylesheets[print][] = ma_feuille_pour_impression.css`.

Enfin, dans cet exemple, nous disposons les feuilles de style à la racine du dossier du thème. Il est bien évidemment possible de les placer dans un sous-répertoire en indiquant cette fois comme valeur au champ un chemin relatif au dossier du thème, par exemple `stylesheets[all][] = styles/ma_feuille.css`.

Pour que Drupal prenne en compte la nouvelle définition de notre thème, il nous faut retourner sur l'administration des thèmes. Un simple affichage de cette page suffit à forcer la mise à jour des définitions ; une fois ceci fait, l'utilisation des deux feuilles de style est alors activée. C'est ainsi que vous pouvez ajouter autant de feuilles de style que vous le désirez sans jamais avoir à les insérer vous-même dans le code XHTML. D'ailleurs, à noter, vous ne devez jamais les insérer vous-même dans une page. En effet, Drupal dispose de mécanismes d'optimisation des feuilles de style qui ne fonctionneraient pas si vous insériez une balise `<link>` dans un des modèles.

Utiliser les scripts

Lorsque vient le moment de mettre en place des comportements dynamiques (onglets, effets, menus déroulants, etc.), il est nécessaire de pouvoir ajouter de nouveaux fichiers JavaScript. Comme pour les styles, PHPTemplate cherchera dans le dossier de thèmes un fichier `script.js` qui, s'il existe, sera inclus automatiquement dans chacune des pages. Pour ajouter d'autres scripts que `script.js`, le même principe que pour les feuilles de style s'applique, à savoir une modification du fichier `mon_theme.info`.

Illustrons cette possibilité avec un script bien pratique qui arrondit automatiquement les coins des boîtes. Le script que nous allons utiliser pour cette opération est un greffon pour jQuery.

jQuery est une bibliothèque très légère de manipulation du DOM (*Document Object Model*) écrite en JavaScript, et extensible via des cen-

URL jQuery

- ▶ <http://jquery.com/>
- ▶ <http://plugins.jquery.com/>

taines de greffons qui lui ajoutent des comportements particuliers : effets, animations, boîtes de dialogue, etc. Par chance, cette merveilleuse bibliothèque est aussi la bibliothèque JavaScript de Drupal en standard.

Le greffon qui nous intéresse ici se nomme jQuery Corner, et vous pouvez le télécharger à partir de l'adresse <http://blue-anvil.com/archives/anti-aliased-rounded-corners-with-jquery>.

Dans l'archive [zip](#) se trouve un fichier `jquery.curvycorners.packed.js` qui est la version compressée du greffon. Placez-le dans le dossier du thème, et créez dans ce même dossier un fichier `script.js` dans lequel nous allons écrire le code en utilisant jQuery Corner pour arrondir les angles de nos blocs. Nous aurons préalablement étudié ces blocs avec Firebug pour découvrir que ceux se trouvant sur la région Barre de droite ont pour chemin CSS `#sidebar-right .block`.

Contenu du fichier `script.js`

```
Drupal.behaviors.myModuleBehavior = function (context) {
  $('#sidebar-right .block .content').corner({
    tl: { radius: 5 },
    tr: { radius: 5 },
    bl: { radius: 5 },
    br: { radius: 5 }});
};
```

L'objet `Drupal.behaviors` utilisé ici est fourni par Drupal pour collecter l'ensemble des comportements dynamiques, de manière à ne les activer *que* si JavaScript est lui-même activé par Drupal. Utilisez toujours cette méthode lorsque vous désirez lancer un script juste après le chargement de la page sur le navigateur client.

Le corps de la fonction est assez explicite pour qui connaît JavaScript : il s'agit de rechercher le chemin CSS `#sidebar-right .block`, correspondant, nous le savons à présent, aux styles CSS de nos blocs de droite, pour y appliquer le greffon d'arrondissement des coins.

Pour ceux qui ne connaissent pas encore jQuery, son apprentissage vaut le détour et l'Internet regorge de tutoriels bien construits, y compris en français.

À ce stade, nos deux fichiers JavaScript ne sont pas encore connus du thème. Comme pour les styles, si nous n'avions eu que le fichier `script.js`, il aurait été reconnu automatiquement sans avoir à renseigner `mon_theme.info`. Mais, comme nous avons ici deux scripts, nous devons, comme pour les feuilles de style, les référencer tous les deux dans `mon_theme.info`.

Ajout des scripts dans `mon_theme.info`

```
; déclaration des scripts
scripts[] = script.js
scripts[] = jquery.curvycorners.packed.js
```

Et comme pour les styles, une fois `mon_theme.info` sauvegardé, nous devons repasser par le panneau d'administration des thèmes pour que notre thème soit rechargé par Drupal. Ceci fait, et comme par magie, vous constaterez que les coins des blocs se sont arrondis.

Figure 11-30
Les coins des blocs sont automatiquement arrondis.

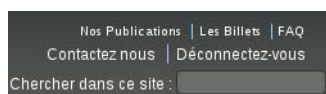


Second niveau de personnalisation : les modèles

Lorsque les besoins deviennent très spécifiques, ajouter des feuilles de style ne nous sera plus d'aucune utilité car c'est l'organisation du contenu propre au thème qui s'avère inadaptée.

Lorsque nous regardons notre « nouveau » thème, nous constatons par exemple que les liens secondaires sont placés au-dessus des liens primaires. Ceci est déroutant si l'on désire y placer les grandes sections de notre site, et n'est de toute façon pas conforme à ce que nous avons décidé.

Figure 11-31
Mauvaise disposition des rubriques



Vous l'aurez compris, ce sont les modèles qu'il va nous falloir modifier car ce sont eux qui mettent en forme chaque élément fondamental de la page. Dans le cas présent, c'est le modèle principal, `page.tpl.php`, qui est responsable de la disposition des liens primaires et secondaires.

Modifier le modèle de page

Avant de procéder aux modifications nécessaires au changement de disposition des liens primaires et secondaires, observons quelques instants le contenu de notre `page.tpl.php` :

En-tête (simplifié) du modèle page.tpl.php

```
<html xmlns="http://www.w3.org/1999/xhtml" lang="<?php print
$language->language ?>" xml:lang="<?php print $language-
>language ?>" dir="<?php print $language->dir ?>">

<head>
  <?php print $head ?>
  <title><?php print $head_title ?></title>
  <?php print $styles ?>
  <?php print $scripts ?>
</head>
```

Nous trouvons ici le squelette complet de notre page codée en XHTML et nous observons les insertions en PHP de variables fournies par PHP-Template.

Dans la première partie du modèle est défini l'en-tête de la page. Détaillons un peu les variables dont nous disposons ici.

BON À SAVOIR Connaître le contenu des variables de modèle

Si vous ne savez pas ce que contient une variable particulière, par exemple `$language`, vous pouvez insérer dans le code (de préférence dans le `<body>` s'il s'agit de `page.tpl.php`) la fonction PHP `var_dump(...)`.

```
14 <body>
15 <?php var_dump($language); ?>
16 <table border="0" cellpadding="0" cell
17 <tr>
```

Figure 11–32

Cette astuce vous permettra, après sauvegarde et rafraîchissement de la page sur le navigateur, de lister le contenu complexe d'une variable et ainsi de déterminer les informations que vous pouvez en retirer.

```
object(stdClass)[5]
  public 'language' => string 'fr' (length=2)
  public 'name' => string 'French' (length=6)
  public 'native' => string 'Français' (length=9)
  public 'direction' => string '0' (length=1)
  public 'enabled' => string '1' (length=1)
  public 'plurals' => string '2' (length=1)
  public 'formula' => string '($n>1)' (length=6)
  public 'domain' => string '' (length=0)
  public 'prefix' => string '' (length=0)
  public 'weight' => string '0' (length=1)
  public 'javascript' => string 'fddcbd4c68ca6a11e685c2a8b9b9f2c0' (length=32)
  public 'dir' => string 'ltr' (length=3)
```

Figure 11–33

- `$language` contient la langue courante utilisée par Drupal. Sa valeur est un objet et dispose à ce titre de nombreux champs dont `$language->language`, qui contient la forme abrégée de la langue `fr`.

BONNE PRATIQUE Chemin vers une ressource

Utilisez toujours `$base_path` dans votre thème dès que vous avez à indiquer le chemin vers une ressource.

La manière idéale pour référencer une ressource sans se préoccuper du paramétrage du serveur est de l'utiliser en conjonction avec la fonction `path_to_theme()` qui donne le chemin du thème.

```
<?php
print $base_path.path_to_theme()
    ."/mon_image.png" ?>
```

- `$head_title` contient le titre de la page qui a été paramétré dans *Configuration du site > Informations*.
- `$head` est à inclure systématiquement pour permettre à Drupal d'ajouter des éléments dans l'en-tête selon ses besoins.
- `$styles` et `$script` contiennent respectivement la liste des balises d'inclusions des feuilles de style et des scripts pour la page. À ce titre, l'inclusion de ces variables est absolument obligatoire.
- `$base_path` contient le chemin de base de Drupal. Il s'agit la plupart du temps d'un simple slash, mais peut aussi correspondre à une valeur plus longue si votre site fonctionne sur une URL du type `http://monSite/monDrupal/`. Dans ce cas, `$base_path` aura pour valeur `monDrupal/`.

Plus loin dans la page, nous trouvons d'autres variables :

- `$front_page` prend une valeur booléenne avec `TRUE` si le modèle est utilisé pour la page d'accueil du site et `FALSE` dans le cas contraire.
- `$logo` contient le code XHTML (essentiellement une balise `img`) permettant d'afficher le logo du site qui a été configuré dans le thème.
- `$site_name`, `$site_slogan` et `$mission` contiennent respectivement le nom, le slogan et la mission du site qui ont été paramétrés dans la section *Informations*.
- `$title` contient le titre du contenu, à ne pas confondre avec le titre du site. Lors de l'affichage d'un nœud, le titre du contenu est le titre du nœud.
- `$breadcrumb` contient le fil d'Ariane de la page.
- `$messages` contient les éventuels messages d'erreur renvoyés par Drupal.
- `$help` contient un éventuel message d'aide contextuel renvoyé par Drupal.
- `$tabs` contient le code XHTML des onglets affichés dans les pages.
- `$primary_link` et `$secondary_link` contiennent respectivement le code XHTML pour la liste des liens primaires et secondaires.
- `$closure` contient les éléments XHTML que Drupal doit mettre à la toute fin du document XHTML. Cet élément est obligatoire, comme celui composant la balise `body` en début de document et utilisant la variable `$body_classes`.
- `$search_box` est le code XHTML de la boîte de recherche dans les contenus.
- `$content` est une variable on ne peut plus importante car elle reçoit le contenu XHTML d'un nœud (articles, pages, etc.).

Dans de nombreux cas, vous remarquerez que l'inclusion de ces variables est conditionnée par le fait qu'elles ont un contenu (par exemple, `<php if`

(`$logo`): `?>`). Ce mécanisme est utilisé pour rendre optionnel l'affichage de tel ou tel élément en fonction de la configuration du thème, comme abordé précédemment. Ainsi, si l'affichage du logo est désactivé, la variable sera vide et le code correspondant non inclus.

Les dernières variables que nous trouvons dans ce modèle sont celles correspondant aux régions que nous avons étudiées dans le chapitre « Structurer les pages » :

- la région En-tête représentée par la variable `$header` ;
- la région Pied de page représentée par la variable `$footer` ;
- la région Barre de droite représentée par la variable `$right` ;
- la région Barre de gauche représentée par la variable `$left`.

Fort de cette connaissance, nous avons à présent tous les outils nécessaires pour modifier la présentation des liens primaires et secondaires. Pour ce faire, il nous faut avant toute chose repérer les variables `$primary_links` et `$secondary_links` dans le modèle, et modifier comme suit en déplaçant le bloc de code faisant référence à `$secondary_links` après la table.

Modification de `page.tpl.php` pour positionner les liens secondaires

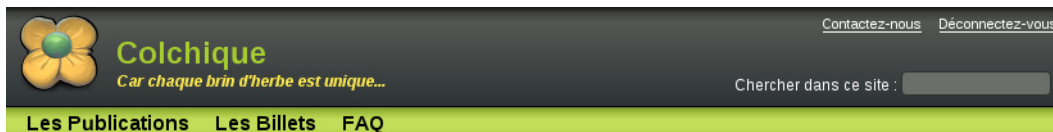
```
<?php if (isset($primary_links)) { ?>
<?php print theme(
  'links',
  $primary_links,
  array('class' => 'links', 'id' => 'navlist')) ?><?php } ?>
<?php print $search_box ?>
</td>
</tr>
<tr>
  <td colspan="2"><div><?php print $header ?></div></td>
</tr>
</table>
<?php
  if (isset($secondary_links)) { ?>
<?php print theme(
  'links',
  $secondary_links,
  array('class' => 'links', 'id' => 'subnavlist')) ?>
<?php } ?>
```

Il ne reste qu'à sauvegarder le modèle pour observer le résultat sur la page d'accueil.



Figure 11–34
Les liens secondaires
correctement placés

Figure 11–35
Nouvelle version
de l'en-tête



Modifier le modèle des blocs

Nous irons un peu plus vite sur ce modèle, car il est beaucoup plus simple que celui concernant la page.

Le modèle bloc.tpl.php

```
<div class="block block-<?php print $block->module; ?>"
id="block-<?php print $block->module; ?>-<?php print $block-
>delta; ?>">

    <H2><?php print $block->subject; ?></H2>

    <div class="content"><?php print $block->content; ?></div>

</div>
```

S'agissant des variables, ce modèle reçoit juste l'objet `$block` de la part de PHPTemplate. Cet objet contient cependant un certain nombre de champs qu'il convient de détailler.

`$block->module` et `$block->delta` forment à eux deux un identifiant unique pour le bloc. En effet, comme nous l'avons vu, tous les blocs sont fabriqués par un module. Il en est de même pour les blocs que nous créons manuellement, qui sont en réalité générés par le module Block. `$block->module` contient le nom du module dont est issu le bloc et `$block->delta` contient l'identifiant du bloc généré par ce module. En effet, un module peut générer plusieurs blocs et le nom du module seul ne suffirait donc pas à l'identifier.

Dans le code de ce modèle, Bluemarine utilise ces deux identifiants pour associer à chaque `<div>` représentant un bloc :

- la classe `block`, commune à tous les blocs ;
- la classe `block-<nom du module>`, commune à tous les blocs issus d'un même module ;
- l'identifiant `block-<nom du module>-<delta du bloc>`, qui est spécifique à ce bloc uniquement.

Pour connaître les modules et les deltas d'un bloc, que vous visez à travers un élément de votre feuille de style, le plus simple consiste à l'afficher et à utiliser le bouton *Inspecter* de Firebug.

Les deux derniers champs relatifs à l'objet `$block` sont `$block->subject`, qui contient le titre du bloc, et `$block->content`, son contenu.

Modifier le modèle des contenus

Le modèle `node.tpl.php` est utilisé pour transformer un contenu (article, page, CCK, etc.) en un fragment de code XHTML, qui sera inséré dans `page.tpl.php` à travers la variable `$content`. Ici aussi, nous sommes loin de la complexité de `page.tpl.php` :

Le modèle `node.tpl.php`

```
<div class="node"><?php if ($sticky) { print " sticky"; } ?>
  <?php if (!$status) {
    print " node-unpublished"; } ?>">

  <?php if ($picture) {
    print $picture;
  }?>

  <?php if ($page == 0) { ?>
  <h2 class="title">
    <a href="<?php print $node_url?>"><?php print $title?></a>
  </h2><?php }; ?>

  <span class="submitted"><?php print $submitted?></span>
  <div class="taxonomy"><?php print $terms?></div>

  <div class="content"><?php print $content?></div>

  <?php if ($links) { ?>
    <div class="links">&raquo; <?php print $links?></div>
  <?php }; ?>
</div>
```

Les variables utilisables dans ce modèle sont les suivantes :

- `$sticky` et `$status` indiquent respectivement si les cases à cocher *épingler en haut de la page* et *publier* ont été activées pour le contenu que l'on cherche à formater ;
- `$teaser` indique si le contexte d'affichage du modèle est une vue résumée (en page d'accueil, par exemple) ou une vue complète ;
- `$picture` est un fragment de code XHTML contenant la référence à l'avatar de l'auteur du contenu. Si cette option n'est pas activée dans le paramétrage du thème, son contenu est vide ;

- `$page` est très importante car lorsqu'elle est à `TRUE`, elle nous prévient que l'on cherche à afficher le contenu dans son entier ; à `FALSE`, c'est le résumé du contenu qui est traité ;
- `$submitted` contient le code XHTML décrivant l'auteur et la date d'écriture de l'article ;
- `$taxonomy` contient un fragment de code XHTML donnant la liste des termes de taxonomie qui sont rattachés à ce contenu. Cela permet, en cliquant dessus, de faire ressortir les contenus traitant des même thèmes que l'article ;
- `$content` contient le code XHTML du corps de la page ;
- `$link` contient le code XHTML de la liste des liens associés au contenu. Vous y trouverez, par exemple, le lien *Ajouter un commentaire* ou *En savoir plus*.
- `$node` contient le nœud au grand complet. Elle n'est pas utilisée dans notre modèle mais mérite d'être citée car elle contient toutes les informations de base de contenu (ex. `$node->title` pour le titre).

Sur ce modèle, nous allons combler une lacune du thème Bluemarine. En effet, d'un point de vue CSS, rien ne nous permet de distinguer si notre nœud est affiché dans le contexte d'une vue résumée ou d'une vue complète, alors que cette information nous serait très utile pour définir un affichage différent dans chacun des deux cas.

Modifier le modèle des commentaires

C'est le modèle `comment.tpl.php` qui est chargé de formater chaque commentaire, de la même manière que `block.tpl.php` formate chaque bloc. Lorsque PHPTemplate a effectué le rendu du nœud, puis de chacun des commentaires, il fusionne l'ensemble de ces fragments XHTML pour le fournir dans la variable `$content` à `page.tpl.php`.

Voyons le contenu du modèle `comment.tpl.php` d'un peu plus près :

Le modèle `comment.tpl.php`

```
<div class="comment">?php print ' '. $status; ?>
    <?php if ($picture) {
        print $picture;
    } ?>
    <h3 class="title">?php print $title; ?></h3>

    <?php if ($new != '') { ?>
        <span class="new">?php print $new; ?></span>
    <?php } ?>

    <div class="submitted">?php print $submitted; ?></div>
```

```

<div class="content">
  <?php print $content; ?>
  <?php if ($signature): ?>
    <div class="clear-block">
      <div>--</div>
      <?php print $signature ?>
    </div>
  <?php endif; ?>
</div>
<div class="links">&raquo; <?php print $links; ?></div>
</div>

```

Nous avons ici des variables qui ont le même rôle que celles utilisées pour le modèle de contenu : `$status`, `$picture`, `$title`, `$submitted`, `$content`. Mais nous y trouvons aussi quelques nouveautés :

- `$signature` qui contient, si l'utilisateur l'a définie et si l'administrateur l'a autorisée, la signature de l'utilisateur ;
- `$new` qui contient le message « nouveau » lorsqu'un commentaire a été posté sur ce contenu après la dernière visite de l'utilisateur courant ;
- `$comment` qui contient un objet Commentaire complet, comme `$node` pour le modèle `node.tpl.php` ou `$block` pour `block.tpl.php`.

Nos seuls changements consisteront ici à corriger un petit problème dans le modèle lié à ce double chevron (`»`) qui s'affiche au-dessus de chaque ligne, puis à supprimer le titre du commentaire qui n'apporte pas grand-chose.

Nous en profiterons aussi pour améliorer la feuille de style de manière à mettre le portrait de l'utilisateur à gauche, à installer une bordure au-dessus des liens que l'on alignera à droite, et enfin à supprimer la bordure bleue autour de chaque commentaire (en jouant sur l'attribut `border` de l'identifiant CSS `comment`).

Ceci fait, nous pouvons rafraîchir la page du navigateur pour admirer le résultat.



Figure 11-36

Les commentaires avec les portraits correctement positionnés

Nous voilà arrivé au terme du second niveau de personnalisation de notre thème qui, en l'état, est déjà très présentable. Nous pouvons cependant aller encore plus loin et modifier encore plus finement certains aspects qui nous échappent avec les deux premiers niveaux.

Troisième niveau de personnalisation : la fonction `theme()`

Pourquoi aller plus loin ?

Nous avons utilisé pour le moment tout ce qui se trouvait dans notre dossier `mon_theme` sans nous poser trop de questions. Cette approche, alliée aux feuilles CSS, nous permet de modifier énormément de choses, mais nous risquons d'être rapidement bloqué lorsqu'il s'agira d'effectuer des tâches plus complexes, par exemple :

- faire un modèle différent pour chaque type de contenu ;
- fournir à un modèle d'autres variables que celles qui nous sont données en standard ;
- modifier des éléments qui ne sont pas présents dans le thème, comme l'apparence des liens ou des images.

Dans le domaine des thèmes Drupal, les notions abordées jusqu'ici correspondent au périmètre des connaissances généralement maîtrisées par la majorité des themeurs. Lorsqu'il y a nécessité d'aller plus loin, certains se lancent alors dans le « bidouillage » de modèles existants, mais rendent par la même occasion le système fragile et difficile à maintenir. Pourtant, à chacune de ces questions existe une réponse et/ou une solution précises dans le système Drupal de gestion de l'apparence.

Vision générale du système de thèmes de Drupal

Chaque donnée de Drupal est transformée en code XHTML par ce qu'on appelle un « hook de thème ». Ne confondez pas ce *hook de thème* avec le *thème de site* que nous sommes en train de réaliser. Un hook de thème est une opération de transformation entre une donnée (par exemple un objet PHP `lien` qui contient des champs `href`, `title`, etc.) et un fragment de code XHTML.

Les hooks de thème sont définis par les modules activés dans Drupal. Chaque module peut définir un nombre infini de hooks. Par exemple, le hook de thème qui transforme un objet `block` en un bloc XHTML est défini par le module standard de Drupal, `Block`.

Les hooks de thème s'appellent les uns les autres pour créer des codes XHTML complexes. Ainsi, à la création d'une page, c'est le hook de thème `Page` qui est appelé ; il va à son tour appeler le hook `Node`, et plusieurs fois le hook `Blocks` pour chacune des régions du thème. À son tour, le hook `Blocks` appellera plusieurs fois le hook `Block` (sans s), avec en paramètres les différents blocs de cette région.

Pour appeler ce hook, Drupal utilise une seule et unique fonction appelée `theme()` qui prend en premier paramètre le nom du hook, et pour les suivants les données à formater. La construction d'une page est donc une longue série d'appels imbriqués à la fonction `theme()`.



Figure 11-37 L'enchaînement des appels de la fonction `theme()`

Jusque-là, nous retrouvons une logique que nous connaissons déjà. Mais en réalité, les appels à `theme()` descendent beaucoup plus loin dans le détail qu'il n'y paraît de prime abord. `theme('block', ...)`, par exemple, devra afficher des liens, et ainsi faire appel à `theme('link', ...)`. Les commentaires auront besoin d'images et ainsi faire appel à `theme('image', ...)`.

Vous l'aurez compris, la moindre donnée de Drupal est tôt ou tard formatée en XHTML par un appel à la fonction `theme` et au hook qui convient pour cette tâche.

La question qui se pose légitimement est : où se trouvent les modèles pour un lien ou une image ? La réponse dépend du choix opéré par le module lorsqu'il a défini son hook. Il dispose en effet de deux méthodes :

- soit il décide que le hook doit être géré par un autre modèle ;
- soit il préfère confier ce travail de transformation à une simple fonction PHP.

La raison pour ne pas utiliser uniquement des modèles est avant tout une question de performances. En effet, une fonction `theme('link')` est appelée pour chaque lien présent sur une page et ce, à chaque fois que la page est affichée. Sachant qu'un modèle est en moyenne cinq fois plus lent qu'une fonction PHP, utiliser des modèles pour de petites données à formater est très préjudiciable sur le temps de fabrication de la page.

Utiliser les noms de modèles éligibles

Que le hook soit défini par une fonction ou par un modèle, vous avez toujours les pleins pouvoirs pour modifier ce comportement par défaut.

Lorsque le module a défini le hook comme étant géré par un modèle, le modèle par défaut se trouve toujours dans le dossier du module. La première action possible consiste à dupliquer ce fichier dans votre dossier thème. Drupal utilisera systématiquement ce qui se trouve dans le dossier du thème en priorité par rapport à n'importe quel autre emplacement. C'est en réalité ce que nous avons fait depuis le début avec les modèles `node.tpl.php`, `block.tpl.php`, `page.tpl.php` et `comment.tpl.php`, dont les versions par défaut se trouvent dans le dossier des modules, respectivement, `modules/node`, `modules/block`, `module/system` et `module/comment`. Vous pouvez y jeter un coup d'œil pour vérifier cela par vous-même.

Mais l'extension des modèles par défaut ne s'arrête pas là. Pour répondre aux besoins que nous avons énoncés en introduction de ce chapitre, vous avez aussi la possibilité de définir des modèles dérivés à partir d'un modèle principal, qui ne seront utilisés que dans certains cas particuliers.

Il est ainsi possible de rendre un modèle dépendant de certaines conditions, par exemple un modèle de nœud qui ne serait utilisé que pour les contenus de type Billet, ou encore un modèle de page qui ne s'activerait que si l'URL est de type rubrique. Le principe est que pour un même hook, il peut y avoir une série de noms de modèles éligibles, le nom par défaut portant systématiquement le nom du hook.

D'un point de vue pratique, il s'agit simplement de dupliquer le modèle de base du hook (ex. `node.tpl.php`) et de le renommer avec un nom compatible avec une éligibilité et correspondant à votre besoin. Voyons maintenant pour chaque grande catégorie de hook, les noms de modèles dérivés éligibles.

Noms de modèles éligibles pour le hook Page

Pour ce hook, la liste des noms de modèles éligibles est déterminée à partir du chemin interne de la page. Ainsi, une page avec pour chemin interne `chemin/12/page` aura la liste de modèles éligibles suivante :

- `page-chemin-page.tpl.php` : ce nom de modèle est construit à partir de tous les éléments non numériques du chemin. En effet, pour Drupal, un élément numérique est très souvent un élément variable faisant référence à un ID en base de données. Par exemple, `node/1/edit` signifie « la page d'édition du nœud ayant 1 pour ID en base de données ». Le modèle portant le nom `page-node-edit.tpl.php` permettrait ainsi de fonctionner pour toutes les pages d'édition quel que soit l'ID du nœud ;

- `page-chemin-12-page.tpl.php` : c'est-à-dire chacun des éléments du chemin ;
- `page-chemin-12.tpl.php` : tous les chemins commençant par `chemin/12` (incluant donc `chemin/12` tout court) ;
- `page-chemin.tpl.php` : tous les chemins commençant par `chemin` ;
- `page.tpl.php` : le modèle de base du hook, qui sera utilisé si aucun autre n'existe.

À noter que si vous désirez créer un modèle spécifique pour la page d'accueil, vous devez simplement le nommer `page-front.tpl.php`.

Enfin, dernier détail, les chemins dont nous parlons ici étant les chemins internes, il n'est donc pas question d'utiliser les alias dans ces noms de modèle. Si vous donnez un alias `societe` à `node/6`, le modèle `page-societe.tpl.php` ne fonctionnera pas, il faudra utiliser `page-node-6.tpl.php`.

Noms de modèles éligibles pour le hook Node

Les noms de modèles éligibles pour le hook Node permettent de définir un modèle spécifique à un type de contenu donné. Il est possible de créer un modèle différent pour un contenu de type Page ou Article, tout en gardant le modèle standard pour tous les autres types.

Pour construire le nom d'un modèle de nœud éligible, il faut connaître l'identifiant du type de contenu. La manière la plus simple est d'aller dans l'administration des types de contenu, l'identifiant qui nous intéresse étant indiqué à la colonne *Type*.

Nom	Type	Description	Opérations
Article	story	Un <i>article</i> , similaire dans la forme à une <i>page</i> , est idéal pour créer et afficher du contenu qui informe et pousse les visiteurs à réagir. Revues de presse, actualité, billet de blog peuvent être créés avec un <i>article</i> . Par défaut, un <i>article</i> est automatiquement publié sur la page d'accueil de votre site et permet aux visiteurs d'ajouter des commentaires.	<div>modifier</div> <div>gérer les champs</div> <div>supprimer</div>

Figure 11-38
Découverte du type interne d'un type de contenu

Une fois le type connu, vous pouvez construire votre nom de modèle sous la forme `node-type.tpl.php` où `type` est l'identifiant interne du type de contenu. Par exemple, pour un modèle spécifique à un contenu de type Article, le nom sera `node-story.tpl.php`.

Notez que le même principe est applicable aux noms de modèles éligibles pour le hook Comment. Ainsi, le nom de modèle `comment-story.tpl.php` sera utilisé uniquement pour les commentaires liés à un contenu de type Article.

BON À SAVOIR Plus de modèles ?

Il existe encore d'autres hooks proposant une liste de noms de modèles éligibles. Vous trouverez une liste exhaustive dans la documentation de Drupal, à l'adresse :

► <http://drupal.org/node/190815>

Noms de modèles éligibles pour le hook Block

Nous avons vu qu'un bloc est défini par le *nom du module* qui le génère et son *delta*. Ces informations seront utilisées pour définir la liste des modèles éligibles :

- `block-module-delta.tpl.php` : ici nous avons toutes les informations pour un modèle spécifique à un bloc unique. `module` et `delta` sont à remplacer par le nom du module et la valeur du delta, par exemple `block-comment-0.tpl.php` sera le modèle éligible pour le bloc de delta 0 (ce sont les commentaires récents), fourni par le module Comment ;
- `block-module.tpl.php`, qui sera utilisé pour tous les blocs issus d'un même module. Par exemple, `block-comment.tpl.php` sera le nom de modèle éligible pour tous les blocs du module Comment ;
- `block-region.tpl.php`, qui sera utilisé pour tous les blocs d'une région donnée. La valeur de la région sera en général `left`, `right`, `header`, `footer` ou `content` ;
- `block.tpl.php` : le modèle par défaut du hook Block si aucun autre n'a été trouvé.

Modifier les modèles invisibles

Jusqu'à présent, nous avons travaillé sur des modèles liés aux hooks Node, Block, Comment et Page. Ce n'était pas trop difficile car n'importe quel thème de site fournit une version de ces modèles qu'il suffit de modifier. Mais qu'en est-il des modèles cachés au cœur de chaque module ?

Prenons un exemple. Vous avez peut-être remarqué que depuis l'activation dans le thème de la prise en charge des portraits dans les commentaires, nous avons un problème : l'absence de portrait pour les utilisateurs anonymes. Ce n'est pas vital, mais pas très esthétique non plus.

Drupal propose un réglage pour cela, dans *Gestion des utilisateurs* > *Paramètres des utilisateurs*, à la section *Portraits* et dans le champ *Portrait par défaut*. Il serait ainsi parfaitement possible de mettre une image dans le thème, par exemple, `sites/all/themes/mes_themes/mon_theme/utilisateur-inconnu.png`, et de renseigner exactement ce chemin dans le champ *Portrait par défaut*.

Le problème de cette approche est son manque de pérennité. En effet, que se passera-t-il si l'on supprime ce thème pour en mettre un autre ? Ou si on déplace le thème dans un autre dossier ? L'image disparaîtrait en même temps que le thème, et il y a fort à parier que l'on perdrait du temps à comprendre d'où vient cette anomalie.

Une autre option consisterait à modifier le modèle `comment.tpl.php`, ce qui ne serait pas bon non plus car nous y ajouterions du code n'ayant pas grand-chose à faire à cet endroit. En effet, l'image d'un utilisateur n'est pas un élément spécifique aux commentaires, car nous avons vu qu'il était possible de l'activer pour les contenus. Si nous faisons une modification sur ce point dans `comment.tpl.php`, il faudra alors reproduire tout ce code dans `node.tpl.php`. Même si l'on peut s'accorder sur le fait qu'il est inconcevable qu'un utilisateur anonyme écrive un contenu, ce serait oublier que certains utilisateurs authentifiés n'auront peut-être pas défini leur portrait.

La *bonne* approche consiste donc à trouver quel hook est responsable de la fabrication du portrait utilisateur. Cet exercice serait fastidieux si l'on ne disposait pas d'un outillage approprié, et en l'occurrence, de l'indispensable module Devel.

Une fois que vous avez installé ce module, n'activez que les sous-modules Devel et Theme Developer. Ceci fait, retournez sur l'article de bienvenue. Vous devriez voir apparaître en bas à gauche de la page une étrange petite zone avec une case à cocher *Themer info*. Cochez la case pour faire apparaître l'outil d'aide sous la forme d'un panneau.

Vous avez ici un outil très proche de Firebug en termes d'utilisation. En passant la souris sur les différentes zones de la page, un cadre rouge apparaît. Allez sur le portrait de l'utilisateur Marine dans le commentaire qu'elle a rédigé et cliquez dessus. Le panneau vous affiche alors toutes les informations à connaître pour modifier l'apparence du portrait.

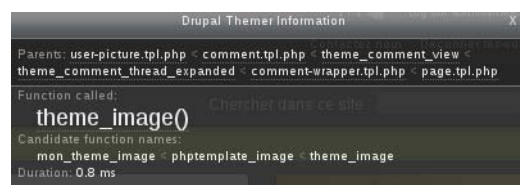


Figure 11–39

Découverte des fonctions de thèmes avec le module Devel

Nous y apprenons que l'image a été générée par le hook `Image` à travers la fonction `theme_image()`, mais ce n'est pas l'image qui nous intéresse ici.

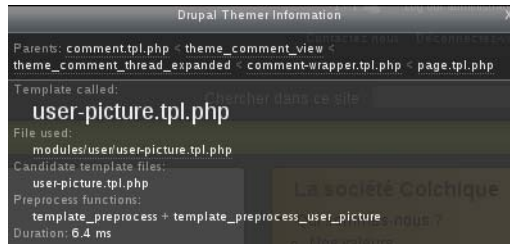
En regardant en haut du panneau, vous avez la liste de tous les modèles qui ont été appelés en partant de `page.tpl.php` jusqu'à `comment.tpl.php`, puis `user-picture.tpl.php` : là réside l'information intéressante. Il existe donc un modèle `user-picture.tpl.php` qui génère le code XHTML de notre portrait.

L'étape suivante consiste à comprendre où se trouve ce modèle et pour cela, cliquez sur le lien `user-picture.tpl.php`. Le panneau affiche à présent des informations plus précises sur ce modèle.

URL Module Devel

► <http://drupal.org/project/devel>

Figure 11-40
Découverte des fonction responsables
de l’affichage du portrait de l’utilisateur



Nous voyons ici que le modèle se trouve dans le dossier `modules/user`. Ce hook est donc défini par le module User, ce qui est logique puisque ce dernier gère tous les aspects liés à l'utilisateur.

L'étape suivante consiste à recopier ce modèle dans le dossier de notre thème et d'en examiner le contenu.

Le contenu d'origine de `user-picture.tpl.php`

```
<div class="picture">
  <?php print $picture; ?>
</div>
```

Comme vous le voyez, il n'est pas très complexe et nous pouvons le modifier pour prendre en compte l'absence d'image.

Notre version de `user-picture.tpl.php`

```
<div class="picture">
  <?php
    if ($picture) {
      // L'image est disponible, il suffit de l'utiliser
      print $picture;
    } else {
      // L'image n'est pas disponible, nous prenons celle par
      défaut
      print theme('image', path_to_theme()."/utilisateur-
      inconnu.png");
    }
  ?>
</div>
```

Le code ainsi modifié traitera le cas où l'image du portrait est inexistante. Il ne présente aucune difficulté notable, à l'exception de l'appel à la fonction `theme()` que nous commençons à connaître.

- `theme()` est utilisée ici pour faire appel au hook de thème image avec en second paramètre le chemin vers l'image à transformer en balise XHTML.
- `path_to_theme()` est une fonction que vous utiliserez continuellement. Elle renvoie tout simplement le chemin vers le thème actif. Il

est très important de toujours utiliser cette fonction pour être certain que, quel que soit l'endroit où vous mettez votre thème, les ressources que vous définissez sont toujours bien accessibles.

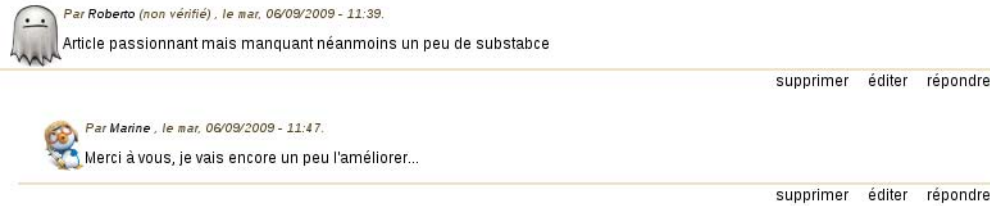


Figure 11–41
Nos anonymes ont maintenant leur portrait.

Le résultat de l'appel de thème est donc une balise XHTML `` correctement formée et entièrement prise en charge par Drupal. Nous pouvons maintenant rafraîchir notre article pour vérifier que les commentaires anonymes sont à présent bien dotés d'une image par défaut.

Personnaliser une fonction de thème

Comme évoqué dans la section « Vision générale du système des thèmes de Drupal », lorsqu'un module définit un hook de thème, il a le choix entre l'associer à un nom de modèle ou à une fonction PHP. La plupart des petits éléments (liens, images, libellés, etc.) sont transformés par la seconde méthode. Comment modifier un comportement non désiré dans un tel cas de figure ?

Pour illustrer la solution à ce problème, nous allons tenter de modifier la manière dont le nom des utilisateurs est présenté dans les commentaires. En effet, vous avez sans doute remarqué l'apparition du libellé (*non vérifié*) à la droite du pseudo des utilisateurs anonymes.

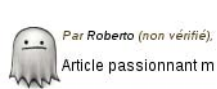


Figure 11–42
Ce « non vérifié » est disgracieux, nous allons le faire disparaître.

Cette petite phrase indique simplement que le contributeur Roberto n'est pas un utilisateur authentifié du site. Nous pouvons décider que cette « fonctionnalité » n'apporte pas grand-chose et nous voulons la supprimer.

À l'aide du module Devel, nous allons donc pointer ce texte pour trouver l'identité du hook et du modèle qui s'y cachent.



Figure 11–43
Recherche de la fonction de thème responsable de l'affichage du nom d'utilisateur

Dans ce cas de figure, nous sommes face à une petite nouveauté : il semble qu'aucun modèle ne prenne directement en charge le rendu visuel de ce nom d'utilisateur, le dernier modèle évoqué en haut du panneau étant notre classique `comment.tpl.php`.

En revanche, nous apprenons que l'intégralité du libellé *Nom d'utilisateur* a été généré avec la fonction `theme_username()`. Nous sommes donc dans le cas de figure d'un hook (`username`) associé non pas à un modèle, mais à une fonction PHP. Pas de panique : comme pour les modèles, Drupal permet de remplacer n'importe quelle fonction de thème par votre propre version.

Si nous nous intéressons à la section *Candidate function names* (littéralement « Noms de fonction éligibles »), nous découvrons une liste de noms de fonction classés par priorité (de la plus forte à la plus faible). Comme pour les modèles, nous pourrions fabriquer nous-même l'une de ces fonctions pour prendre à notre charge l'apparence du nom d'utilisateur.

Lorsque la fonction `theme('username', ...)` sera lancée, elle cherchera l'une de ces fonctions en choisissant en priorité la première de la liste, ici `mon_theme_username`. De manière plus générique, lorsque `theme('nom_du_hook', ...)` est appelé, les fonctions suivantes sont recherchées :

- `<nom_du_thème>_<nom_du_hook>` : comme notre thème a pour nom `mon_theme` et que le hook s'appelle `username`, il s'agit pour nous de la fonction `mon_theme_username`, comme l'indique le panneau du module Devel ;
- `phptemplate_<nom_du_hook>` : ce qui nous donne pour le hook `username` le nom de fonction `phptemplate_username`. Cette fonction est à utiliser avec prudence, car rien n'empêche deux thèmes actifs de l'employer. Beaucoup de thèmes utilisent ce format de nom de fonction, avant tout pour des raisons historiques ; préférez donc la version précédente ;
- `theme_<nom_du_hook>` : c'est l'implémentation par défaut du rendu XHTML pour le nom d'utilisateur, utilisé par Drupal si aucune des fonctions précédentes n'existe.

Pour créer notre propre fonction `mon_theme_username`, nous allons nous inspirer de la version d'origine `theme_username`. Pour ce faire, vous pouvez la rechercher dans les sources de Drupal, ou plus simplement cliquer sur le nom de cette fonction dans le panneau du module Devel. Vous êtes alors dirigé automatiquement sur la documentation en ligne (en anglais) de l'API Drupal, un site qui est la référence de tout ce qui touche au développement de ce CMS.

URL Documentation en ligne Drupal

► <http://drupal.org/handbooks>

theme_username

[Drupal 4.7](#)
[Drupal 5](#)
[Drupal 6](#)
[Drupal 7](#)

includes/**theme.inc**, line 1601

Versions

4.7 – 7 theme_username(\$object)

Format a username.

Parameters

\$object The user object to format, usually returned from [user_load\(\)](#).

Return value

A string containing an HTML link to the user's page if the passed object suggests that this is a site user. Otherwise, only the username is returned.

Related topics

Default theme implementations Functions and templates that present output to the user, and can be implemented by themes.

Code

```
<?php
function theme_username($object) {

  if ($object->uid && $object->name) {
    // Shorten the name when it is too long or it will break many tables.
    if (drupal_strlen($object->name) > 20) {
      $name = drupal_substr($object->name, 0, 15) . '...';
    }
    else {
      $name = $object->name;
    }
  }
}
```

Figure 11–44

L'aide en ligne sur la fonction theme_username

Sur cette page se trouvent des explications sur le rôle de la fonction, sur ses paramètres et valeurs retournés, ainsi que sur le code source de la fonction. Ne faites pas toujours confiance à ce code qui est là à titre d'exemple. L'information intéressante ici est le nom de fichier ainsi que la ligne où se trouve le code de la fonction dans les sources de Drupal. Ouvrez donc avec un éditeur le fichier `includes/theme.inc`, à la ligne 1 601 pour trouver la fonction `theme_username`. (Notez qu'elle ne sera peut-être pas exactement à cette ligne, preuve qu'il ne faut pas accorder une confiance aveugle en la documentation.) À la fin de la fonction, vous trouverez le code qui nous pose problème.

La fonction theme_username, la partie ajoutant (not verified)

```
} else {
  $output = check_plain($object->name);
}

$output .= ' ('. t('not verified') .)';
} else {
  $output = variable_get('anonymous', t('Anonymous'));
```

BON À SAVOIR Les traductions dans le code

Notez au passage dans le code que nous venons de commenter l'utilisation de la fonction `t()` dans `t('not verified')`. Cette fonction est au système de traduction de Drupal ce que `theme()` est au système de thèmes. Le principe général est que tous les textes d'un module ou d'un thème contenus dans du code PHP doivent être d'une part écrits en anglais, d'autre part toujours passés dans cette fonction. C'est cette fonction qui traduit les termes anglais dans n'importe quelle langue prise en charge par Drupal. C'est aussi elle qui rend possible l'extraction de toutes les chaînes de caractères à traduire pour un thème ou un module donné, afin de les fournir à un traducteur.

```
}
    return $output;
}
```

Nous allons à présent créer notre propre fonction `mon_theme_username`. Pour cela, il nous faut commencer par créer un fichier très important appelé `template.php` dans le dossier du thème.

Ce fichier est le seul qui doit contenir du code PHP complexe dans votre thème : n'en mettez donc jamais ailleurs, et surtout pas dans les modèles.

Une fois le fichier créé, ajoutez la balise PHP en tête du fichier (`<?php`) et recopiez à sa suite la fonction `theme_username`, avant de renommer cette fonction `mon_site_username`. Nous pouvons dès lors tranquillement désactiver cette ligne de code qui ajoutait (*non vérifié*) en la mettant en commentaire.

Modification de la fonction `mon_theme_username()`

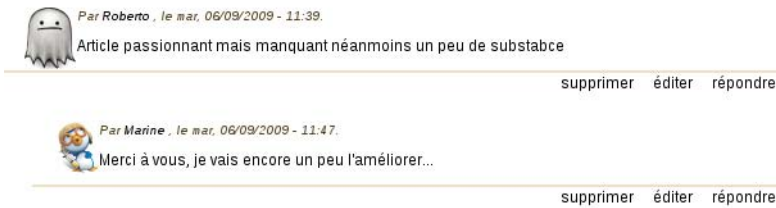
```
} else {
    $output = check_plain($object->name);
}
// $output .= ' (' . t('not verified') . ')';
} else {
    $output = variable_get('anonymous', t('Anonymous'));
}
return $output;
}
```

Si nous testions notre thème dès maintenant, cela ne fonctionnerait pas encore. En effet, Drupal ne cherche pas – pour d'évidentes raisons de performances – à relire le code de tous les modules et de tous les thèmes pour découvrir de nouveaux hooks de thème ou fonctions de thème. Il effectue cette opération appelée « construction du registre de thème » uniquement lors de l'activation d'un thème.

Pour procéder aux tests, deux solutions se présentent à nous. La première consiste à aller dans le panneau d'administration des thèmes, et le seul fait d'afficher cette page provoque la reconstruction du registre des thèmes.

L'autre méthode – à manier avec précaution – consiste à forcer la reconstruction en ajoutant en tête de votre fichier `template.php` l'appel à la fonction `drupal_rebuild_theme_registry()`. Ainsi, lorsque vous rafraîchirez votre page (via la touche *FR*), le registre sera automatiquement reconstruit.

Vous pouvez maintenant rafraîchir la page pour juger du résultat et de la disparition du libellé (*non vérifié*).

**Figure 11–45**

Nos commentaires maintenant « parfaits »

Voilà, nos commentaires sont maintenant parfaits. Attention, pensez à toujours mettre en commentaire l'appel à la fonction `drupal_rebuild_theme_registry()` une fois la base reconstruite, sans quoi vous nuirez fortement aux performances de votre Drupal, car le registre sera systématiquement reconstruit pour chaque page affichée.

Enrichir les variables d'un modèle

Pour le moment, nous avons uniquement compté sur les variables qui nous étaient fournies en standard dans un modèle. Il ne s'agit là en aucun cas d'une limitation, car Drupal nous offre bien entendu la possibilité d'en ajouter autant que nécessaire.

BONNE PRATIQUE Utilisez intensivement l'enrichissement des modèles

L'enrichissement en variables d'un modèle est un aspect très important, car c'est ce qui assurera une bonne indépendance entre la présentation (les modèles) et les données. Ne sachant pas comment enrichir les variables d'un modèle, beaucoup de thèmeurs ajoutent du code métier directement dans les modèles PHP, les rendant ainsi très difficiles à lire, pour ne pas dire illisibles, et surtout très dépendants des données.

L'exemple critique de cette mauvaise pratique est l'utilisation de requêtes SQL lourdes dans les modèles, dupliquées sur des dizaines de modèles différents : un véritable enfer à traquer pour effectuer des opérations de maintenance ou d'évolution. Lorsque nous avons besoin d'afficher une nouvelle donnée non prévue dans le modèle d'origine, la seule chose qui doit apparaître dans le modèle est une variable `$ma_nouvelle_donnee`, et rien d'autre. Le modèle doit toujours rester une chose simple. Quelques tests, des variables, et c'est tout.

Le cas pratique que nous allons aborder pour illustrer cette possibilité consistera à ajouter à notre thème une nouvelle région. Pour l'instant, héritant de Bluemarine, `mon_theme` utilise par défaut les cinq régions standards de PHPTemplate. Imaginons que nous voulions mettre en place une sixième région qui se placerait juste au-dessus du contenu, et juste en dessous de l'en-tête.

La première chose à faire est d'indiquer à Drupal la présence de cette région. Pour cela, nous allons ajouter de nouvelles informations au fichier `mon_theme.info`. Le principe de `mon_theme.info` est que tout ce qui n'y est pas spécifié – ici les régions – bénéficie de valeurs par défaut (les cinq régions standards). En revanche, si nous ajoutons une nouvelle information, celle-ci remplacera toutes les valeurs par défaut qui ne seront alors plus utilisées ; il nous faut donc dans `mon_theme.info` définir six régions, sans quoi nous perdrons l'usage des cinq régions fournies au départ en standard.

Ajout des six régions à `mon_theme.info`

```
regions[left] = Left sidebar
regions[right] = Right sidebar
regions[content] = Content
regions[header] = Header
regions[footer] = Footer
regions[content_top] = Content Top
```

Nous utilisons pour cela la syntaxe `regions[identifiant_region] = Nom de la région`. Vous constaterez que les identifiants de région correspondent aux noms des variables de région que nous avons étudiées avec le modèle `page.tpl.php` : `$left`, `$right`, `$content`, `$header` et `$footer`. Il s'agit là d'une convention permettant aux personnes qui utilisent le thème de s'y retrouver facilement. En ajoutant la région `content_top`, nous indiquons ainsi que nous allons mettre à disposition une variable `$content_top`.

Notez seulement que ces identifiants doivent suivre les mêmes règles qu'un identifiant PHP (pas d'espace, d'accent, de tiret, etc.).

Une fois le fichier `mon_theme.info` sauvegardé, l'étape d'enrichissement à proprement parler peut commencer. Comme pour la personnalisation d'une fonction de thème, l'enrichissement des variables d'un modèle passe par la création d'une fonction dans `template.php`. La différence ici réside dans le fait que toutes les fonctions éligibles seront exécutées les unes après les autres. C'est pour cela que nous parlons d'enrichissement et non de redéfinition. Vous mettrez ainsi dans le corps de ces fonctions uniquement les variables que vous souhaitez ajouter.

Comme pour les fonctions de thème, deux noms de fonction sont utilisables, `phptemplate_preprocess_<nom du hook>` et `<nom du thème>_preprocess_<nom du hook>`. Dans notre exemple avec le hook `Page`, cela nous donnera en guise de noms `phptemplate_preprocess_page` et `mon_theme_preprocess_page`. Comme pour les fonctions de thème, nous donnerons préférence à la seconde possibilité.

Cette fonction va prendre en paramètre un tableau associatif (`clé => valeur`). Ce paramètre utilise logiquement un passage par référence

(`&$variables`) pour en permettre la modification par le corps de la fonction. Lorsque ce tableau devra être donné au modèle, chaque clé deviendra une variable prenant la valeur correspondante, ce qui nous donnera le code suivant à mettre à la suite dans `template.php`.

La fonction d'enrichissement des variables du thème Page

```
function mon_theme_preprocess_page(& $variables) {
  $variables['content_top'] = theme('blocks', 'content_top');
}
```

Nous retrouvons ici la fonction `theme` – utilisée cette fois-ci sur le hook `Blocks` – qui prend en paramètre l'identifiant de la région. Ce hook cherchera automatiquement tous les blocs présents sur cette région et renverra leur rendu XHTML. Ce rendu est stocké dans le tableau `$variables` à la clé `content_top`, qui lors du passage du modèle, donnera la variable `$content_top`.

La dernière étape est l'utilisation de la variable `$content_top` dans le modèle `page.tpl.php`, action qui se fait très simplement en insérant le code PHP correspondant :

Ajout de la nouvelle région dans le modèle

```
<?php print $help ?>
<?php print $content_top; ?>
<?php print $content; ?>
<?php print $feed_icons; ?>
```

Il ne nous reste qu'à rafraîchir le registre des thèmes par l'une des deux techniques données à la section précédente, avant d'aller observer le résultat dans le panneau d'administration des blocs.



Figure 11–46
Apparition de notre nouvelle région

Choisissez un bloc dans le stock des blocs désactivés – par exemple, *Propulsé par Drupal* –, faites-le glisser dans la région *En-tête*, puis enregistrez. Comme vous le voyez, le logo Drupal apparaît bien au-dessus du contenu, la région est donc opérationnelle.

également [créer un bloc personnalisé](#).

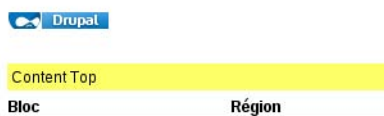


Figure 11–47
La nouvelle région en action

Nul besoin de laisser ce bloc ici, mais ce type de région peut s'avérer par exemple très utile pour insérer des blocs de publicité.

Enrichir les variables de tous les modèles

Nous savons à présent enrichir les variables d'un modèle lié à un hook particulier, mais il existe des situations dans lesquelles vous aurez besoin de transmettre une variable à tous les modèles, tous hooks confondus. Un grand classique illustrant ce besoin est de mettre à disposition de tous les modèles une variable `$is_admin` qui contiendra `TRUE` si l'utilisateur est l'administrateur, et une variable `$logged_in` qui contiendra `TRUE` si l'utilisateur est authentifié.

Le principe est ici très proche de ce que nous venons de voir pour l'enrichissement en variables des modèles par hook. Ce sont peu ou prou les mêmes noms de fonctions qui sont éligibles, et elles procèdent exactement de la même manière avec les mêmes paramètres. Ce sont `phptemplate_preprocess` et `<nom du thème>_preprocess`, ce qui dans notre exemple nous donne `phptemplate_preprocess` et `mon_theme_preprocess` : la même chose sans le nom d'un hook, en somme.

Pour ajouter nos deux variables, nous allons donc insérer dans `template.php` la fonction `mon_theme_preprocess`. Nous utilisons ici la même technique que celle employée pour conditionner l'affichage d'un bloc au fait que l'utilisateur soit un administrateur.

Enrichissement de tous les modèles avec `$is_admin` et `$logged_in`

```
function phptemplate_preprocess(& $variables) {
  global $user;
  $variables['is_admin'] = $user->uid == 1;
  $variables['logged_in'] = ($user->uid > 0) ? TRUE : FALSE;
}
```

Une fois `template.php` enregistré et le registre des thèmes rafraîchi, nous pouvons utiliser les variables `$is_admin` et `$logged_in` dans n'importe quel modèle.

Le résultat

Nous voilà arrivé au bout de notre première exploration des thèmes de Drupal qui se conclut par deux petites captures d'écran (voir double-page suivante), l'une du thème Bluemarine avant que nous commençons à y toucher, et l'autre de notre thème final.

Alors bien évidemment, pour précéder toute critique, un graphiste thèmeur professionnel, armé de toutes ses compétences en graphisme et en ergonomie, aurait produit un thème beaucoup plus beau. Mais l'objectif ici n'est pas de viser l'excellence esthétique mais de démontrer à quel point le moteur de thème de Drupal est souple et somme toute, avec un petit effort d'apprentissage, facile à mettre en œuvre.

En résumé

- Un modèle est un fichier PHP auquel le moteur de thème, PHP-Template, fournit des variables.
- Lorsque vous devez changer quelque chose dans un thème, commencez toujours par les feuilles de style. Si ce n'est pas suffisant, changez les modèles, et si cela ne répond toujours pas à votre besoin, ajoutez de nouvelles variables et/ou fonctions de thème.
- Chaque élément d'une page (et la page elle-même) dispose d'un nom de modèle, et d'une série de noms de modèles éligibles prenant en charge les cas spécifiques (ex. `node-story.tpl.php`).
- Ne mettez jamais de code dans un modèle. Les seules instructions qui doivent s'y trouver sont `print` et `if`. Si vous avez besoin de plus, cela signifie que vous devez enrichir le modèle avec de nouvelles variables.
- Le moindre aspect visuel de Drupal est modifiable par le thème, soit par un modèle, soit par une fonction de thème. Le module Devel est l'outil idéal pour découvrir ce que vous devez surcharger.
- Un thème doit fonctionner dans n'importe quel dossier. Ne mettez jamais un chemin en dur, utilisez toujours `$base.path_to_theme()` pour récupérer le chemin vers une ressource.
- Si vous avez besoin d'aide, n'hésitez pas à poser vos questions sur le forum <http://drupalfr.org>, dans la section *Développement*.



Figure 11–48 Le thème Bluemarine d'origine


Colchique
Car chaque brin d'herbe est unique...

Contactez-nous

Chercher dans ce site :

[Les Publications](#)
[Les Billets](#)
[FAQ](#)

Accueil

Bienvenue

Soumis par Marine le 8 juillet 2009 - 11:11am
 Bienvenue sur le site **Colchique Conseil**,
 le cabinet de conseil en ressources humaines qui à l'éthique au fond du coeur.

Vous trouverez sur notre site une présentation complète, des ses intervenants et de **ses valeurs**. Vous pourrez aussi découvrir nos publications ainsi que nos billets d'humeur. Bonne visite L'équipe
<http://www.colchique-conseil.fr/>

Fichier attaché	Taille
La copie d'écran	105.4 Ko
Le manuel utilisateur	274.03 Ko

Moyenne :


Votre notation : Aucun Moyenne : 3 (1 vote)

Ajouter un commentaire

Par Roberto, le 8 juillet 2009 - 12:03pm.
 Article passionnant mais manquant néanmoins un peu de substance.


Par Roberto, le 8 juillet 2009 - 12:03pm.
 Article passionnant mais manquant néanmoins un peu de substance.

répondre

Par Marine, le 8 juillet 2009 - 12:04pm.
 Merci à vous, je vais encore un peu l'améliorer...


Par Marine, le 8 juillet 2009 - 12:04pm.
 Merci à vous, je vais encore un peu l'améliorer...

répondre

La société Colchique

- Qui sommes-nous ?
 - Nos valeurs
 - L'équipe
 - Historique
 - Les chiffres clef

Ce qui s'est dit...

Merci à vous, je vais encore il y a 22 heures 7 min
 Article passionnant mais il y a 22 heures 9 min

Rejoignez-nous...

Nom d'utilisateur : *

Mot de passe : *

Connectez-moi !

- Créer un nouveau compte
- Demander un nouveau mot de passe

© Colchique Conseil - Tout droits réservés - Ce site est propulsé par [Drupal](#).

Figure 11–49 Notre thème dans sa version finale

chapitre 12

AbonnementsFluxÉlémentVueOutilsRechercheAide

Planet Libre (2)

drupal.org

planete drupal.fr (1)

Logiciels libres, linux, le web... - Logiciel...

La planète francophone de PostgreSQL...

NewsForge

Framasoft

Boîte à outils

DLFP - Dépêches (12)

Framablog (1)

MozillaZine-Fr

FreelanceInformatique.com

Développement (24)

Drupal

Gnome (4)

Linux (51)

Date	Titre
mar. 13:01	WIKI d'organisation de la réunion de prise de contact...
dim. 13:57	Mettre à jour Drupal
ven. 19:11	DrupalCon Paris: Pour que tout le monde soit là!
ven. 17:33	I'm coming to DrupalCon Paris video avalanche

DrupalCon Paris: Pour que tout le monde soit là!

Classé sous **drupalcon paris, Isabell S, Marketing**
Créateur **Drupal Groupe France**
marqueur marque-page link cosmos

La fondation DrupalCon Scholarship offre 60 bourses pour que des étudiants, des développeurs et des contributeurs qui ont besoin d'un soutien financier puissent participer à la DrupalCon Paris 2009. Grâce à nos [sponsors](#) et à tous les animateurs de [la journée Formation](#) c'est devenu possible !

Il y a deux types de bourses - **CloseTheGap** et **Local** - qui couvrent une partie ou la totalité des coûts de l'entrée, du voyage et de l'hébergement. Vous pouvez trouver [plus d'information sur le site](#) de la DrupalCon Paris (en anglais).

Nous acceptons les candidatures via le site [DrupalCon Paris 2009](#). Soumettez votre demande depuis votre profil personnel (après avoir créé votre la

506 non lus

Ajouter des fonctionnalités

Le cœur de métier de Drupal est la gestion des contenus et il s'en acquitte fort bien. Cela étant dit, un site web moderne se doit de proposer un certain nombre de fonctionnalités transversales, car l'internaute d'aujourd'hui s'attend très naturellement à les y trouver. Ainsi, récupérer un contenu en version imprimable ou en format PDF, être notifié par courriel lorsqu'un nouveau commentaire a été posté dans une de ses discussions, disposer de son Gravatar, ou encore s'authentifier par son identifiant OpenID, entre autres, forment un ensemble de services sur lesquels il n'est plus possible de faire l'impasse. Drupal fournit en standard bon nombre d'entre eux, et nous verrons qu'il suffit d'y ajouter une poignée de modules judicieusement choisis pour obtenir un site répondant à toutes les attentes, ou presque.

SOMMAIRE

- ▶ Ajouter la syndication à votre site
- ▶ Prendre en charge l'OpenID et les Gravatars
- ▶ Permettre l'impression et la génération de PDF
- ▶ Mettre en place la notation des contenus
- ▶ Créer sa propre liste de diffusion

Ajouter la syndication à votre site

Qu'est-ce qu'un flux RSS ?

De manière très simplifiée, RSS est un format de document normalisé fondé sur un autre format normalisé, le XML, qui permet l'échange d'informations entre deux programmes informatiques.

On appelle « flux RSS » un document au format XML/RSS téléchargé à partir d'un site web par un autre site web ou par une application. Ce document contient simplement une liste d'items comme une liste d'articles, ou encore une liste de commentaires pour un article. Un site web peut publier plusieurs flux RSS, et dans notre cas avec Drupal, nous pouvons publier par exemple le flux des nouveaux articles, le flux de tous les nouveaux articles rattachés à un terme de taxonomie donné, ou encore le flux de tous les nouveaux commentaires liés à un article.

Chaque flux peut contenir les contenus complets (titre, introduction et corps) ou une version allégée du contenu (le titre seul ou le titre et l'introduction). Enfin, chaque flux contient un nombre spécifié des derniers items à prendre en compte (ex. les 5 derniers articles publiés, les 5 derniers commentaires postés, etc.)

L'application ou le site qui lit ce flux RSS ira à intervalle régulier – par exemple toutes les 30 minutes – le relire pour détecter les éventuels nouveaux articles ou commentaires publiés. On dit alors que l'application ou le site est « abonné au flux RSS » ou « syndiqué au flux RSS ».

Les lecteurs de flux

Les applications des flux RSS sont nombreuses. D'un point de vue utilisateur, ils ont permis l'apparition de lecteurs de flux RSS – des applications bureautiques très proches en termes d'interface d'un logiciel de messagerie – permettant de suivre les nouveautés sur des centaines de sites en même temps. Si vous ajoutez le flux RSS de votre site à un tel outil, ce dernier ira voir ce qui s'y trouve de neuf. Ces logiciels permettent, si le flux RSS contient les articles en version complète, de se passer d'une visite sur le site réel.

Dans cette catégorie de logiciels, pour GNU/Linux, citons Liferea pour le bureau Gnome ou Akregator pour le bureau KDE, ou encore Feed-Reader sous Windows. Il existe aussi des versions web de tels logiciels comme le très célèbre Netvibes (www.netvibes.com).

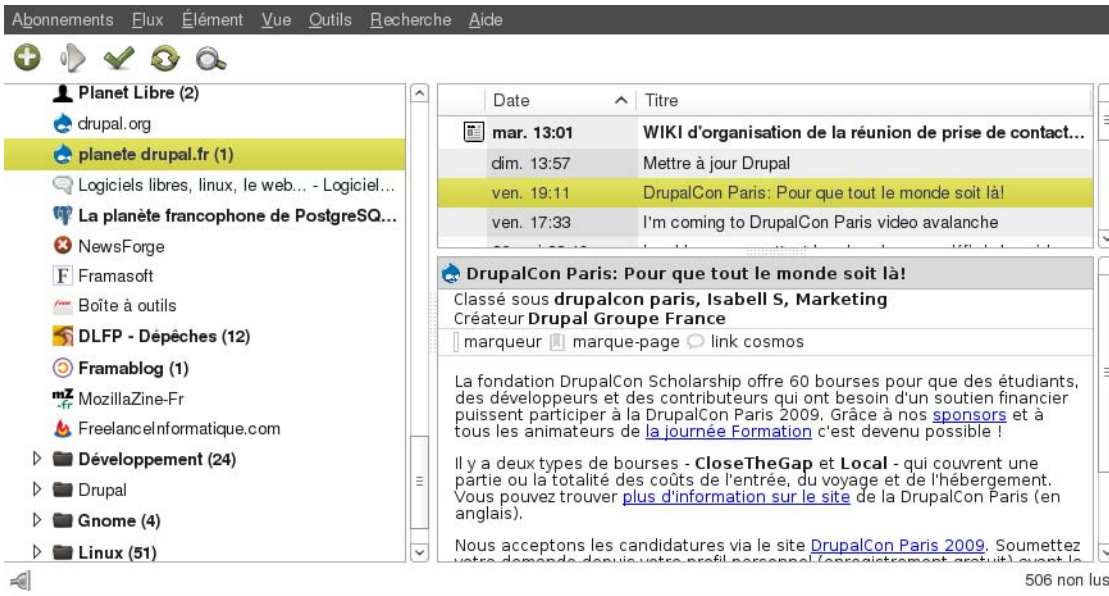


Figure 12-1 Le lecteur de flux RSS Liferea pour Gnome

Une autre utilisation des flux RSS consiste à faire communiquer les sites web entre eux. Généralement, il s'agit d'un site web qui va interroger les flux RSS de plusieurs sites et agréger le résultat en une page web, ou même un autre flux RSS.

Les sites interrogés traitant en général des mêmes sujets d'intérêt, ces agrégateurs jouent ainsi le rôle de portails thématiques permettant à ses utilisateurs d'avoir en un coup d'œil les nouveautés sur un sujet donné provenant de plusieurs sites différents.

Dans le monde du libre, de tels sites sont connus sous le nom de « planet ». Citons par exemple <http://planet-libre.org> qui regroupe les sites parlant des logiciels libres en général, mais aussi <http://planet.drupalfr.org> qui est un planet regroupant les sites français parlant de Drupal.

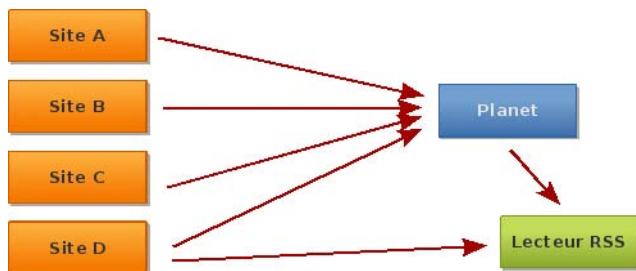


Figure 12-2
Circulation de l'information RSS entre les différents sites

Vous l'aurez compris, les flux RSS sont en quelque sorte des tuyaux, et il est bien évidemment possible de s'abonner au flux RSS d'un planet, utilisable sur un lecteur de flux RSS, comme vous le constatez sur la copie d'écran ci-dessous.



Figure 12-3 <http://planet-libre.org>, un planet dédié au monde du libre

Flux des contenus

Pour activer les flux RSS dans Drupal, il y a peu de choses à faire car cette possibilité est disponible en standard.

Pour accéder aux flux de vos articles, l'URL est toujours la même : <http://www.colchique.fr/rss.xml>. C'est cette adresse qui devra être fournie aux applications RSS et aux sites d'agrégation pour qu'ils puissent s'y abonner.

Si vous avez un outil vous permettant de télécharger ce document (par exemple `wget http://www.colchique.fr/rss.xml` sous Unix), vous obtenez alors le contenu suivant :

Contenu d'un flux RSS

```
<?xml version="1.0" encoding="utf-8"?>
<rss version="2.0" xml:base="http://dev.drupal6.karma-lab.net"
  xmlns:dc="http://purl.org/dc/elements/1.1/">
  <channel>
    <title>Colchique</title>
    <link>http://dev.drupal6.karma-lab.net</link>
    <description></description>
    <language>fr</language>
    <item>
      <title>Mon premier article</title>
      <link>http://dev.drupal6.karma-lab.net/node/4</link>
      <description>
        &lt;p&gt;Ceci est l'introduction de mon
        article&lt;/p&gt;
      </description>
      <comments>http://dev.drupal6.karma-lab.net/node/
        4#comments</comments>
      <category domain="http://dev.drupal6.karma-lab.net/
        taxonomy/term/3">Recrutement</category>
      <pubDate>Wed, 03 Jun 2009 00:13:30 +0000</pubDate>
      <dc:creator>Marine</dc:creator>
      <guid isPermaLink="false">4 at http://dev.drupal6.karma-
        lab.net</guid>
    </item>
  </channel>
</rss>
```

Ce document a été ici intentionnellement réduit à un seul article. Nous retrouvons dans la balise `channel`, le titre du site et son adresse. Et pour `item`, le titre de l'article, le lien qui y mène, sa description, la taxonomie, un lien vers commentaires, l'auteur et la date de création.

Si nous jetons un coup d'œil au code source de notre page de garde en faisant un clic droit puis en sélectionnant *Afficher le code source*, nous trouverons dans la page XHTML le code suivant :

Balise d'insertion d'un flux RSS dans une page XHTML

```
<link rel="alternate" type="application/rss+xml"
  title="Colchique RSS" href="http://www.colchique-conseil.fr/
  rss.xml" />
```

Ce code indique au navigateur qu'il existe un flux RSS nommé « Colchique RSS » disponible à l'URL <http://www.colchique-conseil.fr/rss.xml>. Lorsqu'un navigateur lit cette information dans la page qu'il affiche, il détecte automatiquement le flux et le signale au visiteur en plaçant le symbole des RSS (voir ci-contre) à l'extrême droite de la barre d'adresse. Le visiteur peut alors cliquer sur ce symbole pour s'abonner au flux RSS de notre site. S'il utilise par exemple Mozilla Firefox, le flux sera automa-



Figure 12-4
Symbole des flux RSS

tiquement transformé en « marque-page dynamique ». Il est possible de paramétrer le navigateur pour qu'une telle action ajoute automatiquement ce flux à une application client RSS qui aura été préalablement installée.

Par défaut, Drupal a limité le contenu de l'*item* au titre et à l'introduction. Pour changer ce comportement, vous devez aller dans *Gestion de contenu* > *Publication RSS*. Ici, vous avez la possibilité de choisir le nombre d'articles à placer dans le flux, ainsi que le contenu de chaque élément de ce flux, ce qui pourra être le titre seul, le titre et le résumé, ou enfin le texte complet. Nous allons sélectionner *Texte Entier*.



Figure 12-5
Notre article vu par un lecteur RSS

MODULES MAGIQUES Des flux au format Atom

Atom est un autre format de flux, lui aussi construit sur XML. Sa différence majeure avec RSS est que ce format a été normalisé par l'IETF dans la RFC 4287. Quant à choisir entre Atom et RSS ? La réponse est sûrement, les deux. Pour prendre en charge les flux Atom avec Drupal, vous avez seulement besoin d'installer le module Atom.

► <http://drupal.org/project/atom>

Une fois la configuration enregistrée, retournons sur l'agrégateur et faisons une mise à jour manuelle. Cette fois, les articles sont bien complets.

Flux par taxonomie

Toujours en standard, Drupal vous permet de disposer d'un flux de nouveaux articles pour chacun des termes d'une taxonomie. Imaginons que nous voulions créer l'URL du flux des articles traitant de « recrutement », un terme de notre taxonomie qui a par exemple pour identifiant 3. L'URL du flux associé à ce terme serait alors :

<http://www.colchique-conseil.fr/term/3/feed>.

Vous pouvez construire ainsi n'importe quelle URL pour tous les termes de taxonomie, comme nous l'avons étudié, et ajouter à la fin */feed* pour transformer cette URL en flux RSS.

Notez enfin que Drupal permet au navigateur d'identifier le flux de tous les articles associés à la page de garde, de la même manière qu'il affiche un résultat de recherche lié à un terme de taxonomie donné. Ainsi, si vous cliquez sur le lien *Recrutement* sur un de nos articles, en plus de la liste des articles traitant du recrutement, vous verrez apparaître le symbole RSS à droite de la barre d'adresse : il permettra de s'abonner au flux RSS des articles traitant de recrutement.

Flux des commentaires

Drupal gère l'extension RSS qui permet de disposer des commentaires dans les flux d'articles, mais il peut cependant être intéressant de disposer d'un flux spécifique aux commentaires. Cette tâche est réalisable par le module Comment RSS, qui va venir étendre le système de gestion des flux aux commentaires suivants :

- commentaires publiés sur tout le site par l'URL <http://www.colchique-conseil.fr/crss> ;
- commentaires liés à un nœud particulier par l'URL <http://www.colchique-conseil.fr/crss/node/XXX> (XXX étant le numéro du nœud) ;
- commentaires de tous les nœuds rattachés à un terme de taxonomie particulier par l'URL <http://www.colchique-conseil.fr/crss/term/XXX> (XXX étant le numéro du terme).

URL Module Comment RSS

► <http://drupal.org/project/commentrss>

Une fois le module installé et activé, vous disposez d'un nouvel onglet *Comment RSS settings* dans le réglage de la *Publication RSS* que nous venons d'utiliser, vous permettant de définir précisément les flux mis à votre disposition.

Par conséquent, lorsque vous cliquez dans la page d'accueil de Firefox sur le symbole du flux RSS situé à droite de la barre d'adresse, une liste des abonnements possibles apparaît :



Figure 12-6 Sélection d'un flux RSS avec Mozilla Firefox

Créer des flux personnalisés

Drupal en standard ne permet pas de créer des flux pour autre chose qu'une taxonomie ou pour l'ensemble des contenus. Mais en utilisant une fonctionnalité du module Views, il est possible de générer un flux RSS pour à peu près n'importe quel besoin.

L'un des intérêts du module Views est qu'il permet pour un même paramétrage de générer plusieurs affichages. Nous avons déjà exploité cette possibilité pour nos vues *liste_des_publications* et *liste_des_billets* avec un affichage de type Page qui nous a permis de disposer d'un chemin d'accès et d'une page permettant de naviguer dans la liste.

Pour un même paramétrage par défaut, le module Views peut aussi nous permettre de créer un affichage de type Flux, qui n'est autre chose qu'un

flux RSS. Nous avons donc là la possibilité de transformer n'importe quelle liste de n'importe quel type de contenu en flux.

Si nous retournons dans le paramétrage de la vue `liste_des_publications`, il suffit de sélectionner dans la liste déroulante en haut à gauche, l'entrée *Flux*, puis de cliquer sur *Nouvel affichage*.

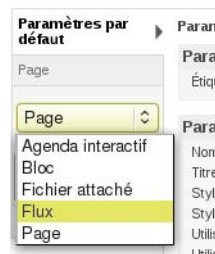


Figure 12-7
Ajout d'un affichage Flux
pour une vue existante

Une fois l'affichage créé, comme pour l'affichage Page, nous pouvons définir son chemin d'accès dans la section *Paramètres du flux*. Cliquez sur *Aucun* pour saisir le chemin `articles/rss.xml`, puis cliquez sur *Mise à jour*.

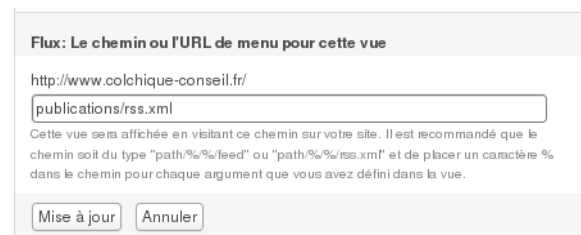
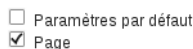


Figure 12-8
Paramétrage du chemin du flux

Nous devons ensuite définir à quelle page ce flux doit être rattaché, c'est-à-dire dans quelle page l'icône RSS correspondante devra s'afficher dans la barre d'adresse du navigateur. Pour cela, nous commençons par cliquer sur le lien *Attacher à : Aucun*, avant de cocher *Page* puis de cliquer sur *Mise à jour*.

Figure 12-9
Paramétrage de l'attachement
automatique du flux à la page



Dans le bloc *Paramètres de base*, pour la valeur du champ *Style de ligne*, vous avez normalement *Plugin de style manquant*. Pour corriger cela, cliquez sur cette valeur et sélectionnez *Node*, puis *Mise à jour*. Apparaît alors le choix du type d'affichage. Vous pouvez laisser *Utiliser le paramétrage RSS par défaut*, qui a l'avantage de reprendre la configuration que nous avons déjà choisie pour l'ensemble des flux Drupal. Cliquez sur *Mise à jour*.

Nous allons à présent modifier le titre de notre flux en cliquant sur le champ *Titre* et en saisissant « Colchique – Flux des derniers articles publiés ».



Figure 12–10
Paramétrage du titre du flux

Le paramétrage est maintenant terminé, vous pouvez *enregistrer*. Cliquez maintenant sur la section *Les publications* de la barre des liens secondaires. Vous devez alors voir apparaître dans la barre d’adresse du navigateur le symbole indiquant qu’un flux RSS est disponible. Vous pouvez vous y abonner pour juger du résultat.



Figure 12–11
Notre nouveau flux vu à travers
le lecteur RSS Liferea

Comme vous le voyez, le module Views peut constituer un outil très puissant pour générer des flux RSS totalement personnalisés, construits à partir de requêtes complexes et auxquels on pourra adjoindre toutes sortes de limitations.

Ajouter la notification sur les nouveaux commentaires

Une fonction basique et pourtant absente par défaut dans Drupal consiste à proposer au visiteur – anonyme ou authentifié – lors d’envoi d’un commentaire, la possibilité de suivre la conversation qui s’ensuit. Ce manque est comblé très facilement grâce au module Comment Notify. Plus encore, ce module permet aussi de notifier l’auteur d’un contenu qu’un nouveau commentaire a été posté.

Une fois ce module activé, dirigez-vous vers *Administrer > Configuration du site > Comment Notify*. Le module n’a pas encore été traduit, tout est donc ici en anglais.

URL Module Comment Notify

► http://drupal.org/project/comment_notify

Figure 12-12
Choix des types de contenu
qui disposeront de la notification

Tout d'abord, sélectionnez les types de contenu pour lesquels la notification doit être activée : *Article* et *Billet*. Il est d'ailleurs un peu dommage que l'auteur n'ait pas prévu de faire vérifier par son module les réglages des commentaires pour les types de contenu. De ce fait, il nous propose ici *Page* et *Référence*, pour lesquels les commentaires sont de toute façon désactivés :

Content types to enable for comment notification :

☒ Article

☒ Billet

☐ Page

☐ Référence

Comments on content types enabled here will have the option of comment notification.

Comme vous le savez, un commentaire peut être soit une réponse à l'article ou au billet, soit un réponse à un commentaire précédent. Il est donc possible de définir quel type de notification vous désirez activer, ainsi que les valeurs par défaut pour l'utilisateur anonyme ou authentifié :

Figure 12-13
Choix du mode de souscription

Available subscription modes :

☒ All comments

☒ Replies to my comment

Choose which notification subscription styles are available for users

Default state for the notification selection box for anonymous users :

Default state for the notification selection box for registered users. :

This flag presets the flag for the follow-up notification on the form that anon users will see when posting a comment

Ensuite viennent les deux modèles de courriels eux aussi en anglais, correspondant aux messages envoyés, pour le premier à un commentateur qui veut être notifié, pour l'autre à l'auteur de l'article pour le prévenir qu'un nouveau commentaire a été posté. Dans les deux cas, à l'instar des modèles de courriels que nous avons étudiés dans le chapitre 6 concernant la création des nouveaux utilisateurs, plusieurs variables peuvent être insérées pour compléter le texte. La liste de ces variables est disponible en dessous de chaque modèle.

Lorsque le paramétrage vous convient, il ne reste plus qu'à *Enregistrer la configuration* et aller dans le paramétrage des droits d'accès pour activer les permissions concernant ce nouveau module. Cela consiste à accorder le droit *subscribe to comments* (souscrire au suivi des commentaires) aux rôles Utilisateur anonyme et Utilisateur identifié, et le droit *administer comment notify* (administrer les notifications de commentaires) au seul rôle Contributeur.

Ceci étant fait, changez d'identité pour devenir Marine et cliquez sur *Mon compte*. À la fin de la page, vous devriez découvrir une nouvelle section *Comment follow-up notification settings*, permettant à Marine de recevoir des notifications à son adresse courriel dès qu'un nouveau commentaire est posté sur l'un de ses articles. Cochez juste la case *Receive node follow-up notification e-mails* et enregistrez le profil.



Figure 12-14
Réglage des souscriptions
dans le profil de l'utilisateur

Pour tester si tout fonctionne correctement, déconnectons-nous et posons un commentaire en tant que visiteur anonyme sur l'un des articles de Marine.

Dans le formulaire de création du commentaire, vous remarquerez que de nouvelles options sont maintenant disponibles :

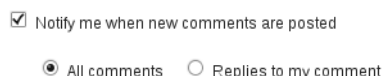


Figure 12-15
Sélection des notifications pour un article

Après la validation du commentaire, Marine reçoit un courriel la notifiant de cette contribution. Si elle répond, l'utilisateur anonyme reçoit à son tour un courriel à l'adresse qu'il a fourni dans le formulaire.

Dans ce courriel, l'utilisateur dispose d'un lien de désabonnement lui permettant de ne plus être notifié à l'avenir. Le cas échéant, il reste toujours possible de désabonner un utilisateur en saisissant son adresse courriel dans l'onglet *Unsubscribe* du paramétrage du module.

Ajouter une lettre d'informations

La gestion des bulletins d'informations peut être effectuée grâce à divers modules dont Simplenews, qui reste le plus connu.

Une fois activé, le module Simplenews a créé pour vous un bulletin par défaut que vous pouvez paramétrer en allant dans *Administrer > Gestion du contenu > Bulletins*.

MODULES MAGIQUES

Être notifié des nouveaux commentaires

Le module Comment Notify est très important pour permettre la communication avec les utilisateurs anonymes (c'est-à-dire les internautes n'ayant pas de compte sur notre site). Même s'il fonctionne également très bien pour les utilisateurs authentifiés, il existe plusieurs modules Drupal capables de gérer les notifications de façon beaucoup plus évoluée comme Subscriptions et Notifications. Le second est clairement plus riche que le premier, et permet d'aller beaucoup plus loin que le simple envoi de courriels, avec notamment la gestion des SMS, de Twitter, etc. Mais en contrepartie, il est également plus lourd à mettre en œuvre.

- ▶ <http://drupal.org/project/subscriptions>
- ▶ <http://drupal.org/project/notifications>

URL Module Simplenews

- ▶ <http://drupal.org/project/simplenews>

Figure 12-16
Activation du module Simplenews

▼ Mail			
Activé	Nom	Version	Description
<input checked="" type="checkbox"/>	Simplenews	6.x-1.0-rc6	Send newsletters to subscribed email addresses. Dépend de : Taxonomy (enabled) Requis par : Simplenews action (enabled)
<input checked="" type="checkbox"/>	Simplenews action	6.x-1.0-rc6	Provide actions for Simplenews. Dépend de : Simplenews (enabled), Trigger (enabled), Taxonomy (enabled)

Figure 12-17
Modification du bulletin

Bulletins		
Liste des bulletins		Ajouter un bulletin
Nom du bulletin	Abonnés	Opérations
Colchique newsletter	0	éditer le bulletin

Pour modifier le bulletin par défaut, cliquez sur *éditer le bulletin*. Vous pouvez ici changer le nom du bulletin ainsi que sa description. Notez que ces deux informations ne sont pas envoyées aux abonnés et sont là à des fins de classement.

Les informations importantes concernent l'expéditeur, dont vous pouvez régler le nom et l'adresse courriel.

Figure 12-18
Définition des coordonnées de l'expéditeur

Informations concernant l'expéditeur

Expéditeur :

Colchique

Adresse courriel de l'expéditeur : *

bulletin@colchique-conseil.fr

Figure 12-19
Ajout du bloc d'inscription aux bulletins

+	Commentaires récents*	Barre de droite	configurer
+	Bulletin : Colchique newsletter*	Barre de droite	configurer
+	Navigation*	Barre de droite	configurer

Vous devriez alors voir apparaître le bloc d'inscription au bulletin.

Figure 12-20
Le bloc d'enregistrement
pour les utilisateurs identifiés

Il s'agit là du bloc spécifique aux utilisateurs authentifiés ; pour rendre disponible la lettre d'informations aux anonymes, vous devez aller cocher la permission *S'abonner à des bulletins* pour le rôle utilisateur anonyme. Ceci étant fait, déconnectez vous pour vérifier que cela fonctionne.

Figure 12-21
Le bloc d'enregistrement
pour les utilisateurs anonymes

Le bloc étant maintenant en place, nous pouvons y saisir une adresse de test pour vérifier qu'il est opérationnel. Ceci fait, Drupal nous prévient que la demande a été prise en compte et qu'un courriel a été envoyé à l'adresse indiquée pour finaliser l'inscription.

Vous recevrez sous peu un courriel de confirmation contenant les instructions nécessaires pour compléter votre demande d'abonnement.

Figure 12-22
Message de confirmation d'un abonnement

Il suffit alors de cliquer sur le lien de confirmation reçu par courriel pour valider complètement l'inscription.

Êtes-vous sûr de vouloir abonner : *test@arnumeral.fr* à : *Colchique newsletter* ?

Vous pouvez toujours vous désabonner plus tard.

Figure 12-23
Page de confirmation de l'abonnement

Nous avons maintenant un nouvel utilisateur abonné à notre bulletin. Nous pouvons confirmer cela en retournant, en tant qu'administrateur, dans la section [Administrer](#) > [Gestion du contenu](#) > [Bulletins](#), puis [Abonnements](#).

Figure 12–24

La liste des abonnements en cours, les petits ruisseaux feront les grandes rivières...

BONNE PRATIQUE Des adresses qui ne vous appartiennent pas

Faites très attention aux actions que vous effectuerez avec des adresses de courriel collectées. Ne perdez jamais de vue que ces adresses ne vous appartiennent pas, qu'il est illégal de les revendre et illégal de les utiliser pour d'autres fins que celles choisies par les personnes qui vous les ont communiquées. Il n'est par exemple pas acceptable d'abonner automatiquement à des bulletins les utilisateurs qui ont créé un compte. Notez également que si la fréquence d'envoi de bulletins est supérieure à celle que vous annoncez, vous risquez d'être catalogué comme fournisseur de pourriels (ou spammeur), ce qui peut avoir des conséquences légales.

L'ergonomie de cette page est devenue pour nous classique, avec sa possibilité d'appliquer des filtres et des actions à une sélection donnée.

Si vous désirez abonner des utilisateurs à partir d'une liste d'adresses de courriel, vous devez passer par l'onglet [Abonnement en masse](#). Il vous suffit alors de saisir les adresses de courriel – une ligne par adresse dans la zone de saisie –, de cocher le bulletin sur lequel l'abonnement s'appliquera sans oublier la langue à utiliser par défaut, puis d'appuyer sur le bouton [M'abonner](#).

De la même manière, vous pouvez, avec l'onglet [Désabonnement en masse](#), désabonner une série d'adresses d'un ou de plusieurs bulletins.

Enfin, l'onglet [export](#) vous permet d'abonner automatiquement vos utilisateurs à un ou plusieurs bulletins d'informations.

Pour terminer, vous pouvez affiner le paramétrage du module Simplenews en allant dans [Configuration du site](#), puis [Simplenews](#). Les paramètres par défaut sont normalement parfaitement opérationnels, et en cliquant sur [Bulletin](#), nous allons juste modifier l'adresse courriel d'essai qui nous permettra de valider l'envoi des bulletins, sans pour autant les envoyer réellement. Dans la section [Adresses courriel pour les essais](#), saisissez une adresse de test que vous êtes sûr de pouvoir consulter, puis enregistrez.

Il est maintenant grand temps de créer puis d'envoyer notre premier bulletin. Pour le module Simplenews, un bulletin est un type de contenu comme les autres. La création d'un nouveau bulletin s'effectue donc simplement par un clic sur [Créer un contenu](#) de type [Newsletter Issue](#). Mais avant tout, nous allons cependant prendre une précaution, en allant éditer le type de contenu [Newsletter Issue](#) pour désactiver dans les options de publication la case à cocher [Placer en page d'accueil](#). En effet, les bulletins ne sont pas faits pour être visibles en première page de notre site.

Le formulaire que vous voyez apparaître ne devrait pas vous dépayser, exception faite du champ *Newsletter* qui permet de sélectionner le type de bulletin correspondant à celui que vous désirez créer.

Figure 12–25 Rédaction de notre premier bulletin

La section *Envoyer le bulletin* vous permet maintenant de contrôler l'état de votre bulletin :

- *Ne pas envoyer maintenant* indique que ce bulletin sera conservé dans les « brouillons » après enregistrement ;
- *Envoyer le bulletin à l'adresse d'essai* nous permet de tester notre bulletin sans pour autant l'envoyer réellement, en utilisant l'adresse d'essai que nous avons saisie plus haut ;
- *Envoyer le bulletin* provoque l'envoi du bulletin. Notez que cet envoi n'est pas immédiat, comme nous le verrons un peu plus loin.

Pour l'instant, laissez la valeur à *Ne pas envoyer maintenant* et enregistrez votre bulletin. Il sera alors stocké dans les brouillons. Vous pouvez avoir accès à la liste des bulletins en brouillon en allant dans *Gestion du contenu* > *Bulletins* puis *Version de Brouillon*.

Afficher les publications de

tous les bulletins

Titre	Bulletin	Date de création	Publié	Envoyé	Modifier
Bienvenue	Colchique newsletter	06/05/2009 - 01:08	<input checked="" type="checkbox"/>	<input type="checkbox"/>	modifier

Figure 12–26
L'état d'expédition des bulletins

Vous pouvez alors modifier votre bulletin jusqu'à ce qu'il vous convienne. Pour tester ensuite l'envoi, effectuez une dernière modification en sélectionnant l'option *Envoyer le bulletin à l'adresse d'essai*, puis appuyez sur le bouton *Sauvegarder et envoyer un test*. Un message Drupal vous notifie alors de la bonne expédition du bulletin, que vous devriez recevoir à l'adresse d'essai. Comme le bon usage l'impose, vous constaterez que le bulletin reçu contient à la fin un lien permettant de se désabonner.

MODULES MAGIQUES Les courriers en HTML

Comme pour les autres types de contenu, vous pouvez sélectionner le format d'entrée qui vous convient le mieux. Notez cependant que si vous désirez utiliser une notation XHTML, vous devrez aussi installer le module MimeMail, qui n'était pas encore disponible en version stable pour Drupal 6 au moment de la rédaction de cet ouvrage. Si vous tenez malgré tout à l'utiliser, vous devez après l'avoir activé aller dans *Administrer* > *Configuration du site* > *Simplenews* puis *Bulletin* pour changer le format du bulletin de texte brut à HTML.

► <http://drupal.org/project/mimemail>

Retournez maintenant dans la liste des brouillons et modifiez à nouveau le contenu pour choisir cette fois-ci l'option *Envoyer le bulletin*, puis enregistrez. Le bulletin a dès lors disparu de la liste des brouillons, mais il n'est pas pour autant envoyé réellement. Si vous allez dans *Gestion du contenu* > *Bulletins*, puis *Publications envoyées*, vous retrouvez votre bulletin avec une petite horloge en guise d'icône dans la colonne *Envoyé*.



Figure 12-27

Le bulletin est en attente d'expédition.

Cette horloge signifie que le courriel est en attente d'envoi avant le prochain passage de cron, le planificateur de tâches.

Comme nous l'avons vu lors de l'installation de notre plate-forme de développement (chapitre 5 « Administrer Drupal et ses modules »), nous n'avons pas configuré cron pour qu'il se lance automatiquement. Nous allons donc l'exécuter à la main pour compléter notre test. Pour cela, dans un nouvel onglet, entrez l'URL <http://www.colchique-conseil.fr/cron.php>. Le sablier tournera un certain temps avant de s'arrêter sur une page blanche. Les tâches planifiées, dont l'envoi du bulletin, ont donc été exécutées. Pour le vérifier, retournez sur l'onglet précédent et pressez *F5* (actualiser).

Titre	Bulletin	Date de création	Publié	Envoyé	Modifier
Bienvenue	Colchique newsletter	06/04/2009 - 23:08			modifier

Figure 12-28

Le bulletin est maintenant expédié.

L'horloge a bel et bien disparu, et la coche nous indique que le bulletin est parti. Nous pouvons le vérifier en jetant un coup d'œil à notre messagerie et en constatant que le bulletin est bien arrivé à l'adresse de notre unique abonné.

Le contenu du bulletin reçu

```
----- BIENVENUE
-----

Bienvenue sur la NewsLetter de Colchique

L'équipe.

-- Me désabonner de ce bulletin:
http://www.colchique-conseil.fr/newsletter/confirm/remove/
7ebba8ad571t4
```

Prendre en charge l'OpenID

L'explosion du Web 2.0 et le développement de l'usage de l'Internet nous ont doté d'une multitude d'identités numériques : à chaque inscription sur un nouveau site, un nouveau couple nom d'utilisateur/mot de passe s'ajoute à notre (déjà longue) liste d'identifiants à retenir.

Parce que cela peut devenir très vite difficile de s'y retrouver, OpenID fournit une réponse à cette schizophrénie numérique en vous permettant, avec les sites qui le prennent en charge, de ne plus jamais saisir d'identifiant et de mot de passe. Le principe est très simple : il vous suffit de créer un compte sur un serveur OpenID pour obtenir un identifiant OpenID que vous utiliserez sur tous les sites qui prennent en charge OpenID en lieu et place du traditionnel couple identifiant/mot de passe.

Choisir son serveur OpenID est donc une étape importante, car c'est lui qui hébergera les informations sensibles (c'est tout de même de votre vie privée qu'il s'agit). Le site <http://www.openidfrance.com> est un choix acceptable car géré par une équipe de bénévoles sûrement moins intéressés par le profit qu'ils pourraient tirer de cette somme d'informations personnelles que d'autres organismes professionnels. Voyons d'abord comment créer un compte OpenID sur ce serveur, pour ensuite examiner la façon dont Drupal gère ce type d'identifiants.

Pour commencer, allez à l'adresse <http://www.openidfrance.fr> et cliquez sur le lien *Créez votre OpenID*.

Vous devez alors saisir un nom d'utilisateur, un mot de passe et une adresse courriel valide pour recevoir la confirmation de l'inscription, puis cliquez sur *Enregistrer*.

RÉFÉRENCE Identité numérique

📖 O. Iteanu, *L'identité numérique en question – 10 scénarios pour la maîtrise juridique de son identité sur Internet*, Eyrolles, 2008

Inscription OpenID France

Choisissez un identifiant qui va devenir votre unique adresse OpenID pour vous connecter à tous les sites.

Nom d'utilisateur	<input type="text" value="Gaston"/>
Mot de passe	<input type="password" value="****"/>
Confirmez votre mot de passe	<input type="password" value="****"/>
Email	<input type="text" value="gaston@colchique-conseil.fr"/>



Recopiez exactement les lettres dans l'ordre dans ce dessin (c'est une sécurité pour vous enregistrer la première fois).

Figure 12-29

La page de création d'OpenID France

Figure 12-30
Vous avez maintenant
votre propre identifiant OpenID.

Une fois que vous avez reçu le courriel de confirmation, cliquez sur le lien qui vous est fourni, et c'est terminé !



[Accueil](#) | [Découvrir OpenID](#) | [Créez votre OpenID](#) | [Les sites OpenID](#) |

Le compte est activé.

Sur le site OpenID, connectez-vous à présent avec votre identifiant et votre mot de passe dans la boîte de connexion en haut à droite pour entrer dans votre espace personnel. Vous voyez alors apparaître votre nouvel identifiant OpenID.

Votre adresse OpenID est

?

Vous avez un site internet ou un blog, vous souhaitez l'utiliser comme identifiant OPENID ?
C'est très simple, il vous suffit de recopier le code ci-dessous entre les balises <head> </head> :

```
<link rel="openid.server" href="http://www.openidfrance.fr/index.php" />
<link rel="openid.delegate" href="http://www.openidfrance.fr/gaston" />
```

Figure 12-31
Cette page vous donne votre adresse OpenID,
ainsi que le moyen d'utiliser l'adresse de votre
serveur comme identifiant OpenID
(voir aparté).

MODULES MAGIQUES Votre propre URL comme identifiant OpenID

Par défaut, votre OpenID est celui fourni dans la première zone de votre écran de connexion. Si le site que vous êtes en train de créer est votre site personnel et que vous avez acheté un nom de domaine, il existe un module très pratique, OpenID URL, qui permet d'utiliser l'adresse de votre site comme identifiant. Une fois activé, vous pouvez le paramétrer en allant dans *Configuration du site*, puis *OpenID URL*. Là, vous devez remplir les deux premiers champs avec les valeurs données dans la seconde zone de l'écran :

- OpenID Server : <http://www.openidfrance.fr/index.php>
- OpenID Delegate : <http://www.openidfrance.fr/gaston>

Une fois cette configuration enregistrée, vous pouvez utiliser l'adresse de votre site Drupal comme OpenID à la place de <http://www.openidfrance.fr/gaston>.

Cliquez maintenant sur l'onglet *Mes identités* pour spécifier les informations que vous désirez rendre visibles aux serveurs sur lesquels vous utiliserez votre identifiant. Au minimum, vous devez bien entendu cocher le nom d'utilisateur et l'adresse courriel. Enfin, enregistrez.

Mon Compte | Mes Informations | **Mes Identités** | Mes Sites | Mes Recherches | Aide

> Mon OpenID visible

> www.wasab.dk

Ma page personnelle : <http://www.openidfrance.fr/loran>

☐ Je veux être invisible ?

Type	Information	Status
<input checked="" type="checkbox"/> Nom d'utilisateur	gaston	Optionnel
<input checked="" type="checkbox"/> Email	gaston@colchique-conseil.fr	Optionnel
<input type="checkbox"/> Prénom Nom		Optionnel
<input type="checkbox"/> Date de naissance	1975-01-01	Optionnel
<input type="checkbox"/> Genre	Femme	Optionnel
<input type="checkbox"/> Langue principale		Optionnel
<input type="checkbox"/> Adresse		Optionnel
<input type="checkbox"/> Code Postal		Optionnel
<input type="checkbox"/> Ville		Optionnel
<input type="checkbox"/> Pays		Optionnel

Figure 12-32
Vous pouvez définir précisément les informations que vous souhaitez faire connaître.

Le paramétrage de votre compte OpenID est maintenant terminé ; cliquez sur [Se déconnecter](#) et retournez sur votre site Drupal pour activer dans la foulée le module OpenID. Lorsque c'est fait, déconnectez-vous.

La boîte de connexion a quelque peu changé et vous disposez maintenant d'un lien [Se connecter avec OpenID](#).

Connexion utilisateur

Nom d'utilisateur : *

Mot de passe : *

☐ [Se connecter avec OpenID](#)

☐ [Créer un nouveau compte](#)

☐ [Demander un nouveau mot de passe](#)

Figure 12-33
Votre site est maintenant compatible avec OpenID.

Cliquez sur ce lien pour faire apparaître la boîte de connexion spécifique à OpenID, dans laquelle vous allez saisir l'identifiant qui vous a été donné par le site OpenIDfrance : <http://www.openidfrance.fr/gaston>.

Figure 12-34
Formulaire de saisie de l'identifiant OpenID

Lorsque vous cliquez sur le bouton *Se connecter*, vous êtes automatiquement redirigé sur votre page d'authentification OpenID.



Identification OpenID

Avant de vous authentifier avec votre OpenID sur un site, vous devez vous connecter une première fois avec vos identifiants

Figure 12-35
Saisie de votre identifiant et mot de passe pour activer l'OpenID

ATTENTION D'où travaillez-vous ?

Si vous utilisez cette procédure sur votre machine de développement, il se peut que cela ne fonctionne pas correctement. En effet, votre machine n'étant pas visible de l'extérieur, elle risque de ne pas pouvoir mener jusqu'au bout le dialogue entre les différents serveurs. Cette procédure est en revanche tout à fait opérationnelle pour un site en production.

Une fois l'authentification réalisée, le serveur OpenID vous demande quelles informations vous souhaitez rendre visibles par le serveur Drupal. Vous pouvez décocher ici toutes les informations qui vous semblent inutiles et choisir entre *Autoriser juste cette fois* et *Autoriser toujours*.

Passée cette étape, vous êtes renvoyé sur votre site Drupal où vous êtes à présent miraculeusement identifié, un compte ayant été créé.

Comme vous le voyez, OpenID est un formidable outil permettant de ne plus avoir à retenir des dizaines de couples identifiant/mot de passe.

Prendre en charge les Gravatars

À l'instar d'OpenID qui permet d'utiliser un seul identifiant pour tous les sites web prenant en charge cette norme, Gravatar permet d'utiliser le même avatar sur tous les sites sur lesquels vous intervenez. La différence est que Gravatar est une solution propriétaire et qu'elle risque un jour d'être supplantée... par OpenID. Toujours est-il que cette plate-forme

est aujourd'hui très appréciée des internautes, et qu'il est difficile de ne pas la prendre en compte.

Pour tester cette fonctionnalité, vous devez d'abord créer votre Gravatar. Allez sur le site <http://fr.gravatar.com/signup> et saisissez l'adresse de courriel que vous souhaitez associer à votre Gravatar.

Inscrivez-vous pour obtenir votre propre Gravatar

Tout ce qu'il nous faut est votre adresse e-mail. Vous recevrez par retour un courrier de confirmation incluant un lien pour activer votre compte, et vous permettre de choisir un mot de passe.

Adresse e-mail:


Figure 12-36
Inscription au service gravatar.com

Une fois inscrit, vous recevrez un courriel de confirmation à l'adresse indiquée, contenant un lien vous permettant d'activer votre compte. Vous sera alors demandés votre pseudo, ainsi qu'un mot de passe.

Une fois ces informations validées, le site vous invite à *Rajouter une image en cliquant ici*, ce que vous faites bien évidemment. Suivez les instructions pour ajouter votre avatar, puis sélectionnez la *catégorie G* pour indiquer que cet avatar n'est pas de nature offensante (et effectivement, il ne doit pas l'être !), et convient donc à tous les sites.

C'est terminé, cet avatar est maintenant associé à votre adresse courriel. Vous avez bien sûr la possibilité d'ajouter d'autres adresses de courriel sur le même compte avec d'autres avatars.

Sélectionner une adresse e-mail ci-dessous à modifier ou [add a new email](#)



gaston@colchique-conseil.fr(primary)

Vous avez sélectionné cette adresse, maintenant choisissez une image, ci-dessous, à associer à cette adresse

[Adresse principale \(ne peut être effacée\)](#) | [N'utilisez pas cette image](#) | [Vérifier ce Gravatar](#) | [Link](#)

Choisissez une image, ci-dessous, à associer à l'adresse sélectionnée (ci-dessus), ou [add a new image](#)



Figure 12-37
Association d'un portrait
à une adresse courriel

Passons à présent à la mise en œuvre de ce gravatar dans Drupal, en installant le module Gravatar Integration. Une fois ce dernier activé, allez dans les droits d'accès et octroyez la permission *Utiliser les gravatars* aux utilisateurs anonymes.

URL Module Gravatar Integration

► <http://drupal.org/project/gravatar>

Figure 12-38
Le Gravatar est maintenant
intégré à notre site.

La dernière étape consiste à bien vérifier que vous avez activé les portraits utilisateur – ce qui est le cas si vous avez suivi jusqu’ici toutes les instructions de ce livre –, et que dans la configuration du thème courant, ces portraits sont bien affichés.

Il ne reste alors plus qu’à se déconnecter pour saisir un commentaire en utilisant l’adresse courriel du Gravatar, et tester ainsi que tout fonctionne.



BONNE PRATIQUE **Attention à vos adresses de courriel**

D’une manière générale, faites attention à vos adresses de courriel. Le site Gravatar est géré par une société privée et, en théorie, rien ne leur interdit de revendre à l’avenir les adresses courriel qu’ils ont collectées. Il est donc fortement conseillé d’utiliser une adresse courriel spécifique que vous mentionnerez dans tous vos commentaires sur tous les sites, et que vous emploieriez pour créer votre Gravatar. Ainsi, si d’aventure cette adresse finissait dans de mauvaises mains, vous n’aurez pas compromis votre adresse principale. Créer une adresse de courriel prend environ dix secondes sur un site comme Gmail ou Yahoo!, ce qui n’est pas cher payé pour les inconvénients que vous pourrez éviter.

Ajouter un formulaire de contact

Un formulaire de contact est un élément très important, quel que soit le type de site que vous cherchez à développer. N’importe quel visiteur désirant des informations, voulant vous prévenir d’un problème, ou souhaitant vous demander une précision, aura tendance à chercher un lien *Contactez-nous* dans la page sur laquelle il se trouve.

BONNE PRATIQUE **Ne pas présumer de la configuration d’un navigateur**

Il peut venir à l’idée d’utiliser, au lieu d’un formulaire, un lien de type `mailto://adresse@courriel.fr`. Une telle URL entraîne l’ouverture de l’éditeur de courrier associé au logiciel de messagerie de l’utilisateur, prérempli avec l’adresse fournie. Le problème de cette approche est que tous les postes clients ne disposent pas de cette fonctionnalité correctement configurée et il arrive alors qu’un clic sur ce type de liens ne provoque aucune action, ou pire, génère une erreur. Ce problème est encore plus vrai lorsqu’il s’agit de terminaux mobiles (téléphone, PDA, etc.). L’option du formulaire est donc à privilégier, pour être sûr de recevoir en toutes circonstances les messages envoyés par vos visiteurs.

Pour créer un tel formulaire, vous devez activer le module standard Contact de Drupal, qui se trouve dans la section *Core - facultatif*. Une fois cela fait, dirigez-vous vers *Construction du Site* puis *Formulaires de Contact*. La première chose à faire ici est d'ajouter une « catégorie », qui peut être considérée comme un type de formulaire ; pour cela, cliquez sur *Ajouter une catégorie*.

Remplissez d'abord le champ *Catégorie*, qui servira aussi de titre au formulaire. S'agissant d'un formulaire de contact général, nous prendrons en guise de titre « Nous contacter ». Vous devez ensuite préciser à qui les messages seront transmis, et pouvez saisir pour cela plusieurs adresses de courriel en les séparant par des virgules.

Vous pouvez ensuite saisir un message qui informera l'utilisateur à l'origine du message de contact que sa demande est bien en cours de traitement (champ *Réponse automatique*).

Catégorie : *

 Exemple : 'votre avis sur le site' ou 'information sur les produits'.

Destinataires : *

 Exemple : 'webmestre@votresite.com' ou 'commercial@votresite.com,support@votresite.com'. Pour entrer plusieurs destinataires, séparez chaque adresse e-mail par une virgule.

Réponse automatique :

 Réponse automatique optionnelle. Laissez vide si vous ne souhaitez pas envoyer à l'utilisateur un message de réponse automatique.

Figure 12-39
Création du formulaire de contact général

Enfin, à la fin du formulaire, vous trouverez le champ *Sélectionné*, qui permet de l'instaurer comme étant le formulaire par défaut ; puisque c'est bien le cas ici, nous allons donc sélectionner *Oui*. Nous pouvons maintenant enregistrer le formulaire. Le module nous amène alors sur l'onglet *Liste*, en confirmant l'enregistrement. Il est bien sûr encore possible de modifier ce formulaire ou bien d'en ajouter d'autres.

Le formulaire de contact général est à présent accessible via l'URL <http://www.cochique-conseil.fr/contact>.

Sachant que d'un point de vue ergonomique, un utilisateur s'attend à trouver le lien *Contactez-nous* soit en haut à droite de la page, soit en bas dans le pied de page, nous allons positionner cette nouvelle option en haut à droite de la page, en ajoutant pour cela un nouvel élément au menu des liens primaires.

BON À SAVOIR Un formulaire de contact pour vos utilisateurs

Le module Contact ajoute aussi un formulaire de contact par utilisateur. Chaque utilisateur peut ainsi activer son propre formulaire en allant dans *Mon Profil*, dans l'onglet *Modifier*, puis dans la section *Paramètres de contact*, en cochant *Formulaire de contact personnel*.

Figure 12-40
Ajout d'un nouveau lien primaire
pour Contactez-nous

Paramètres du menu

Chemin : *

contact

Le chemin de Drupal vers lequel pointe cet élément de menu. Cela peut être un chemin interne à Drupal comme `node/add` ou une url externe comme `http://drupal.org`. Entrez `<front>` pour pointer vers la page d'accueil.

Titre du lien dans le menu : *

Contactez nous

Texte correspond au nom du lien qui sera affiché dans le menu.

Description :

Cliquez ici pour nous laisser un message

La description affichée quand le pointeur survole cet élément de menu.

Figure 12-41
Le lien Contactez-nous est maintenant actif.



Pour compléter nos paramétrages, nous pouvons aussi éditer le bloc créé pour le pied de page et y ajouter un lien vers le formulaire de contact. Pour ce faire, nous pourrions insérer directement une balise de lien `` mais ce ne serait pas très pertinent. En effet, comme nous l'avons vu dans le chapitre sur la création d'un thème, il est plus judicieux, pour les liens sur des chemins internes au site, d'utiliser la fonction `l('Nous Contacter', 'contact')`. Nous devons cependant activer le format d'entrée PHP code, ce qui donne :

Figure 12-42
Changement du bloc Pied de page

Corps du bloc :

© Colchique Conseil - Tout droits réservés - `<?php print l('Contactez-nous', 'contact') ?>` - Ce site est propulsé par `Drupal`.

Le contenu du bloc vu par l'utilisateur.

Format d'entrée

☐ Filtered HTML

- Les adresses de pages web et de messagerie électronique sont transformées en liens automatiquement.
- Tags HTML autorisés : `<a>` `` `` `<cite>` `<code>` `` `` `` `<dl>` `<dt>` `<dd>`
- Les lignes et les paragraphes vont à la ligne automatiquement.

☐ Full HTML

- Les adresses de pages web et de messagerie électronique sont transformées en liens automatiquement.
- Les lignes et les paragraphes vont à la ligne automatiquement.

☒ PHP code

- Vous pouvez entrer du code PHP. Vous devrez inclure les balises `<?php ?>`.

Une fois le bloc enregistré, vous devez obtenir le pied de page suivant. Si ce n'est pas le cas, vérifiez si une erreur s'est glissée dans l'écriture de votre code PHP – ce qui vous montre par la même occasion les difficultés et les pièges d'utilisation de ce filtre (pour mémoire, il ne doit être utilisé que pour l'affichage de variables ou l'appel de fonctions).

© Colchique Conseil - Tous droits réservés - Contactez-nous - Ce site est propulsé par Drupal.

Figure 12-43
Le nouveau pied de page

Nous disposons à présent de deux liens vers le formulaire de contact. Il ne nous reste qu'à modifier les droits d'accès pour donner à l'utilisateur anonyme la permission *Accéder au formulaire de contact global*.

À présent, nous allons pouvoir tester le formulaire en nous déconnectant pour observer le dispositif en conditions réelles :

Contact

Vous pouvez nous laisser un message en utilisant le formulaire de contact ci-dessous.

Votre nom : *

Robert

Votre adresse de courriel : *

robert@une_societe.fr

Objet : *

Demande d'information

Message : *

Bonjour,
J'aurais aimé avoir plus de précisions sur vos expériences dans le domaine des...

Envoyer un courriel

Figure 12-44
Un visiteur nous laisse un message.

Une fois que le visiteur a cliqué sur le bouton *Envoyer un courriel*, il est prévenu de la bonne expédition du message par Drupal.

Votre message a été envoyé.

Figure 12-45
Message de confirmation d'envoi pour le visiteur

Il reçoit en outre un courriel de bonne prise en compte de sa demande. De votre côté, vous recevez le courriel dans votre messagerie à l'adresse indiquée lors de la définition du formulaire.

Ajouter un module d'impression et vos contenus en PDF

Il existe deux services très appréciés par les utilisateurs se trouvant face à un article un peu long : la possibilité d'imprimer l'article ou de télécharger sa version PDF pour le lire plus tard via un outil mobile (type PDA, téléphone, etc.).

URL Module Printer
► http://drupal.org/project/print
URL TCPDF
► http://www.tecnick.com/public/code/cp_dpage.php?aiocp_dp=tcpdf
URL Domtopdf
► http://www.digitaljunkies.ca/dompdf/

BON À SAVOIR Créer des PDF avec WebKit

Une troisième possibilité existe depuis peu et utilise cette fois-ci l'outil wkhtmltopdf. Cet outil n'est pas une bibliothèque PHP mais un petit utilitaire binaire qui fonctionne sous GNU/Linux et sous Windows. Il a l'immense avantage d'être très rapide et d'utiliser le même moteur de rendu que les navigateurs Safari, Google Chrome ou Konqueror (WebKit). Pour de plus amples informations sur l'installation de cet outil, consultez la page :
► <http://arnumeral.fr/node/72>

Figure 12-46
Activation du module Printer

Ces deux fonctionnalités sont prises en charge par le module Printer, qui permet de surcroît l'ajout sur chaque article d'un lien *Envoyer à un ami*.

Ce module nécessite une bibliothèque PHP pour générer des documents au format PDF. Nous avons à l'heure actuelle deux possibilités : l'ancien projet TCPDF qui fonctionne très bien mais souffre de l'absence de support pour les feuilles de style, et Domtopdf qui, lui, gère les fichiers CSS mais ne prend en revanche pas en compte les caractères accentués. De plus, Domtopdf semble être un projet très peu actif depuis deux ans. Pour ces raisons, nous allons préférer TCPDF, certes moins joli, mais plus fiable.

Pour que la prise en charge du PDF fonctionne, vous devez aller sur le site de TCPDF et télécharger l'archive de la dernière version du projet pour PHP 5, avant de la décompresser dans le dossier du module Printer de manière à y créer un sous-dossier `tcpdf`.

Une fois ceci fait, nous sommes prêt à activer les trois sous-modules de Printer, chacun correspondant à une fonctionnalité : génération PDF, impression et envoi par courriel.

▼ Printer, e-mail and PDF versions			
Activé	Nom	Version	Description
<input checked="" type="checkbox"/>	PDF version	6.x-1.7	Adds the capability to export pages as PDF. Dépend de : Printer-friendly pages (core) (disabled)
<input checked="" type="checkbox"/>	Printer-friendly pages (core)	6.x-1.7	Adds a printer-friendly version link to content and administrative pages. Requis par : Send by e-mail (disabled), PDF version (disabled)
<input checked="" type="checkbox"/>	Send by e-mail	6.x-1.7	Provides the capability to send the web page by e-mail Dépend de : Printer-friendly pages (core) (disabled)

Allons ensuite paramétrer ces fonctionnalités dans *Configuration du site* > *Version imprimante* > *Courrier électronique et PDF*.

Cliquez d'abord sur l'onglet *Page* qui permet de paramétrer l'apparence que prendra le lien affiché pour chaque contenu : lien standard, icône, position, etc. Les paramétrages par défaut correspondent aux usages en vigueur, vous pouvez donc les conserver sans aucun problème.

En revanche, deux options intéressantes se trouvent à la fin du formulaire. La première intime au module l'ordre d'ouvrir la version imprimable du contenu dans une nouvelle fenêtre, ce qui n'est pas une bonne idée, et la seconde l'ordre de déclencher automatiquement l'impression, ce qui est une moins bonne idée encore, car il est toujours préférable de laisser ces choix à l'utilisateur. Cela nous donne le paramétrage suivant que vous pouvez enregistrer :

- ☐ Affiche une version imprimable dans une nouvelle fenêtre
Activée, cette option ouvrira la version dans une nouvelle fenêtre/onglet.
- ☐ Envoyer à l'imprimante
Appelle automatiquement la fonction d'impression du navigateur lorsque la version imprimable est affichée.

Figure 12-47
Désactivation des options intrusives

Dans le second onglet, vous pouvez changer le paramétrage concernant les courriels. Comme pour l'onglet précédent, la première partie du formulaire correspond au type de lien, que nous pouvons laisser en l'état. À la fin du formulaire, vous avez la possibilité de choisir de n'envoyer que le résumé du contenu par défaut (le terme « aperçu » est ici assez mal choisi) et de laisser à l'utilisateur le choix d'en envoyer plus. Les deux cases étant cochées par défaut, nous pouvons les laisser ainsi.

Le troisième onglet nous permet d'activer la bibliothèque TCPDF que nous avons installée. Normalement, le module a automatiquement détecté sa présence et vous la propose donc comme *Outil de génération PDF*. Il vous suffit donc de cocher cette option, puis d'enregistrer la configuration.

Outil de génération PDF :


 <sites/all/modules/contrib/print/tcpdf/tcpdf.php>
Cette option choisit l'outil de génération de PDF employé par ce module pour créer la version PDF.

Figure 12-48
Sélection de la bibliothèque TCPDF

Passons sur le choix du type de lien pour aller à la fin du formulaire, où nous pouvons définir le traitement qui sera appliqué au PDF sur le navigateur client. Il peut ainsi être affiché dans la fenêtre courante du navigateur, sur une nouvelle fenêtre (comme pour l'impression), ou simplement proposé en téléchargement. Le dernier choix est celui par défaut, et c'est une bonne option car tous les navigateurs ne sont pas configurés pour afficher correctement le format PDF au sein même d'une page HTML. Vous pouvez laisser tel quel le reste du formulaire qui correspond à des options avancées, et enregistrer la configuration.

Le dernier onglet, *Paramètres*, nous permet de modifier des options communes aux trois modes, comme l'ajout des « URL des versions imprimables », intitulé peu clair signifiant simplement que vous désirez transformer tous les liens du contenu en notes de bas de page.

La seconde option vous permet d'inclure les commentaires au contenu généré (PDF ou impression), option qui est désactivée par défaut et que nous laisserons ainsi.

Vous pouvez explorer le reste des options, qui sont simples à comprendre, puis enregistrer la configuration.

Comme nous l'avons fait jusque là, l'étape suivante consiste à définir les droits d'accès sur ces nouvelles fonctions en ajoutant la permission *Accéder à l'imprimante* à l'utilisateur identifié.

Une fois les droits réglés, vous pouvez vérifier que tout fonctionne en voyant apparaître sous les contenus les nouveaux liens suivants :

» [Ajouter un commentaire](#) | [Envoyer à un ami](#) | [Version imprimable](#) | [version PDF](#)

Figure 12-49
Les liens ajoutés par Printer aux contenus

BON À SAVOIR

Limiter l'impression PDF pour les utilisateurs identifiés

Savoir à quel rôle vous donnez le droit de générer des documents PDF ou imprimables est une question de ciblage de votre site. Sachez cependant que la génération de PDF n'est pas une opération légère et qu'autoriser les anonymes à le faire inclura de facto des automates de recherche tel que Google. Il est donc préférable d'autoriser les utilisateurs anonymes à accéder seulement à la version imprimable du document et à l'envoi par courriel à un ami, et de réserver aux utilisateurs authentifiés l'accès à la version PDF.

Malheureusement, le module Printer a une manière très peu standardisée de gérer les droits sur ses fonctions. Il n'est ainsi pas possible de gérer finement les permissions. L'astuce consiste donc à donner la permission *Accéder à l'imprimante* aux utilisateurs anonymes, et dans l'onglet PDF, dans la section *Visibilité du lien*, de sélectionner l'option *Afficher si le code PHP suivant retourne TRUE* en saisissant le code PHP suivant :

```
<?php
  global $user;
  return $user->uid<0;
?>
```

Le principe est d'utiliser l'objet global `$user`, qui représente l'utilisateur pour qui la page est fabriquée. Si le champ `uid` de cet objet est égal à 0, il s'agit d'un utilisateur anonyme, auquel cas la fonction renvoie `FALSE`.

Ceci étant dit, cette astuce n'est pas parfaite et ne fait que cacher le lien ; le PDF reste accessible à l'utilisateur anonyme connaissant le chemin Drupal pour générer un PDF.

Par exemple, pour l'envoi du contenu par courriel à un ami, nous obtenons le formulaire suivant :

Accueil

Envoyer la page par courrier électronique

Votre adresse de courrier électronique :

Votre nom :

administrateur

Envoyer à :

Entrez plusieurs adresses séparées avec des virgules et/ou écrites sur différentes lignes.

Objet :

administrateur vous a envoyé un message à partir de Colchique

Page to be sent :

Mon premier article

Votre message :

☒ N'envoyer que l'aperçu

Envoyer un courriel

Effacer le formulaire

Annuler

Figure 12–50
Envoi d'un article à un ami

Ajouter la notation de vos contenus

Qu'est-ce qu'une notation de contenu ?

Le principe général d'un système de notation est de permettre au visiteur de donner un avis sur la qualité d'un contenu ou même d'un commentaire. La gestion des notations doit être étudiée avec soin, car elle peut être utilisée de nombreuses manières différentes et influe dans tous les cas sur l'ambiance générale de votre site. Une notation sur les commentaires cachant automatiquement les contributions aux notes négatives, par exemple, risquerait de froisser votre public, même si vous ne cherchiez qu'à améliorer la qualité de l'ensemble et surtout à réduire l'impact néfaste des trolls (voir chapitre 7).

Nous étudierons ici le système de notation à travers une vision plus consensuelle, celle des contenus que vous publiez, et nous verrons la production d'une liste des articles les mieux notés.

Voting API et Fivestar

Il existe de très nombreux systèmes de notation mais tous ou presque utilisent le module Voting API. Ce dernier ne produit rien de visuel, mais fournit le cadre de base indispensable à tous les systèmes de notation : gestion des compteurs, des sommes, des moyennes, etc. Pour notre exemple, nous allons confier la partie visuelle de la notation à un module relativement simple d'accès, Fivestar.

Une fois les deux modules téléchargés et installés, il suffit d'activer les options suivantes :

▼ Vote			
Activé	Nom	Version	Description
<input checked="" type="checkbox"/>	Fivestar	6.x-1.15	Un widget basic (5 étoiles) permettant de voter pour un nœud. Dépend de : Voting API (enabled) Requis par : Fivestar Comments (enabled)
<input type="checkbox"/>	Fivestar Comments	6.x-1.15	Évaluer les nœuds en laissant des commentaires. Dépend de : Voting API (enabled), Fivestar (enabled)
<input checked="" type="checkbox"/>	Voting API	6.x-2.0-rc2	Provides a shared voting API for other modules. Requis par : Fivestar (enabled), Fivestar Comments (enabled)

URL VotingAPI et Fivestar

- ▶ <http://drupal.org/project/votingapi>
- ▶ <http://drupal.org/project/fivestar>

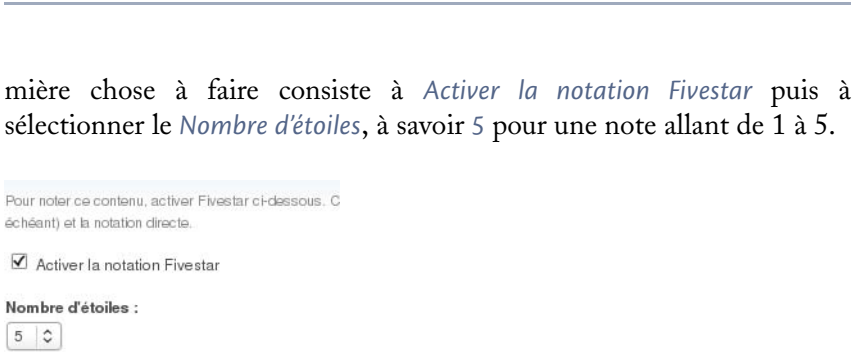
Figure 12-51

Activation des modules Voting API et Fivestar

Comme nous allons orienter notre notation sur les contenus, nous allons devoir choisir et paramétrer les types de contenu qui pourront être notés. Dans notre cas, il va s'agir du type de contenu Article.

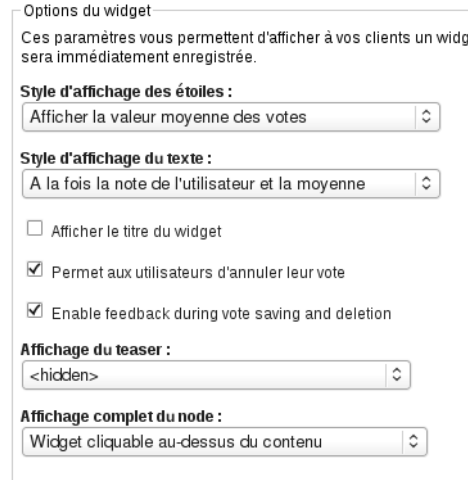
Pour cela, dirigeons-nous vers le panneau de configuration, et cliquons sur [Article](#). Sur ce formulaire que nous connaissons bien, vous constatarez l'apparition d'une nouvelle section : *Notations de Fivestar*. La pre-

Figure 12-52
Activation de la notation pour
les contenus de type Article



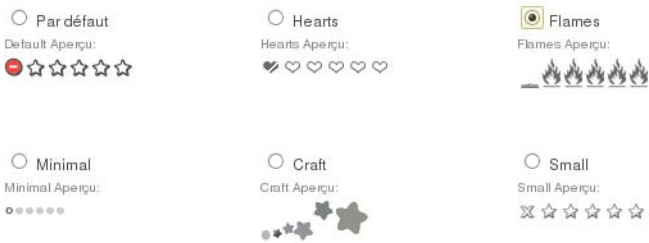
La section *Options du widget* permet de définir finement la manière dont le contrôle de gestion des votes est affiché dans l'article. Nous allons choisir d'afficher le contrôle au-dessus du contenu et de le maintenir caché dans la version résumée du contenu :

Figure 12-53
Paramétrage du contrôle prenant
en charge la notation



Nous pouvons maintenant enregistrer pour passer à la définition de l'apparence visuelle de notre notation. Pour cela, rendons-nous dans *Configuration du site > Fivestar*. Dans cette section, nous pouvons choisir différentes présentations pour le contrôle de notation.

Figure 12-54
Choix de l'apparence visuelle de la notation



Une fois notre choix effectué et enregistré, nous finirons comme toujours ce travail de paramétrage par un petit tour dans le panneau de gestion des droits d'accès, pour cette fois-ci ajouter la permission *rate content* au rôle Utilisateur anonyme.

Ceci étant fait, déconnectons-nous pour vérifier que l'ensemble fonctionne du point de vue du visiteur anonyme, et allons voir la version complète de l'article de bienvenue.

Bienvenue



Aucu vote pour l'instant.

Bienvenue sur le site de **Colchique Conseil**,

le cabinet de conseil en ressources humaine qui a l'*éthique à cœur*.

Vous trouverez sur notre site une présentation complète de la société, d'humeur.

Bonne visite

Figure 12-55

Notre article est prêt à être noté !

Le contrôle de notation apparaît correctement au-dessus du contenu. Toutes les flammes sont pour l'instant grisées, indiquant qu'aucun utilisateur n'a noté l'article. Il suffit de cliquer sur l'une des flammes pour donner sa note à l'article.

Bienvenue



Votre notation : 4 Moyenne : 4 (1 vote)

Figure 12-56

Et ce sera une très bonne note !

Votre vote est alors automatiquement enregistré et affiché, ainsi que la notation moyenne de l'article et le nombre total de votants. Si vous désirez supprimer votre vote, il suffit de cliquer sur le symbole juste avant la première flamme.

Les contenus les mieux notés

Une fois les votes mis en place, vous pourriez logiquement vous demander ce que l'on peut en faire. Une des utilisations les plus courantes consiste à créer une liste des articles les mieux notés, que l'on mettra par exemple dans un bloc.

Pour cela, utilisons de nouveau le module Views qui, outre les pages et les flux, est aussi capable de présenter une liste sous la forme d'un bloc.

Pour nous simplifier la tâche, nous allons dupliquer encore une fois la vue Articles et lui donner comme nom `les_articles_les_mieux_notes`, puis compléter avec la description et l'étiquette de votre choix avant d'enregistrer.

Nous voilà à nouveau dans le panneau de paramétrage de la vue. Commençons par ajouter une nouvelle relation entre le contenu et sa note.

Si vous vous souvenez bien, une relation dans le module Views correspond à la clause `JOIN` d'une requête SQL. Dit de manière plus triviale, une relation permet de prendre en compte des caractères liés à notre contenu, comme la note d'un article. Et la note d'un article, qui est prise en charge par le module Voting API, est bien en relation avec le nœud de l'article qui est, lui, géré par le module Node.

Pour ajouter cette nouvelle relation vers Voting API, cliquez sur le symbole d'ajout d'une relation, puis dans le groupe *Nœud*, sélectionnez *Nœud : Voting results*.

Groupes :

Noeud

☐ Nœud : Individual votes
Votes cast by users on nodes.

☒ Nœud : Voting results
The aggregate results of votes cast on nodes.

Figure 12-57
Ajout de la nouvelle relation

Une fois ce choix validé, le panneau de configuration de notre nouvelle relation s'affiche. Nous allons ici sélectionner *Aggregation function* avec pour valeur *Average* (la moyenne des notes). En effet, les fonctions d'agrégation permettent d'opérer une opération mathématique sur un ensemble de valeurs, ici la moyenne de toutes les notes du nœud.

Paramètres par défaut: Configure Relation Nœud : Voting results

Étiquette :

Voting results

Intitulé pour cette relation, qui ne sera affiché qu'à des fins d'administration.

☐ Exiger cette relation
Si la relation est exigée, les éléments qui ne présentent pas cette relation n'apparaîtront pas.

Data filters

For each piece of content, many pieces of voting data may be saved. I exactly which types should be available via this relationship. **Warning:** empty may result in multiple copies of each piece of content being displayed.

Type de valeur :

No filtering

Vote tag :

No filtering

Aggregation function :

Average

Figure 12-58
Paramétrage de la relation

Nous pouvons mettre à jour. Nous allons maintenant ajouter un nouveau critère de tri pour classer nos valeurs par ordre décroissant de note. Cliquez sur le symbole d'ajout de la zone des tris et sélectionnez le groupe *Voting API results*. Dans la liste qui apparaît, nous allons sélectionner

Voting API results : Valeur, qui correspond à la valeur moyenne de la note que nous avons définie dans notre relation.

Groupe :

Voting API results

- ☐ Voting API results : Fonction
The aggregate function used to calculate the result.
- ☐ Voting API results : Horodatage
The time the results were calculated.
- ☐ Voting API results : Type de valeur
The nature of the results in question (points, percentage, etc).
- ☒ Voting API results : Valeur
The value of an individual cast vote.
- ☐ Voting API results : Étiquette
An optional tag to group multi-criteria results.

Figure 12–59
Ajout d'un nouveau critère

Après la validation de ce choix, nous allons pouvoir indiquer le caractère descendant de notre tri (*Ordre de tri*).

Paramètres par défaut: Configurer critère de

Relation :

Voting results

Ordre de tri :

☐ Ascendant

☒ Descendant

Mise à jour Annuler Ôter

Figure 12–60
Paramétrage du tri

Nous pouvons mettre à jour. Avant de quitter les tris, nous allons supprimer l'ancien critère *Noeud : Date de mise à jour*, qui est un reste de notre vue dupliquée. Cliquez sur ce critère et appuyez sur le bouton *Ôter*.

Nous avons besoin d'afficher une liste de titres d'article avec, pour chacun, un lien vers l'article correspondant. Pour changer le style de ligne, cliquez sur la valeur du champ *Style de ligne* et sélectionnez *Champs*.

Paramètres par défaut: Comment les lignes de cette vue doivent être stylées

☒ Champs

☐ Noeud

Vous pouvez également adapter les paramètres pour le style de la ligne en cours en cliquant sur l'icône.

Mise à jour Annuler

Figure 12–61
Redéfinition du format des lignes

Une fois ce choix validé, validez à nouveau en laissant les paramètres tels quels. Vous constaterez alors que le bloc *Champs* est devenu actif. S'il contient des valeurs, sélectionnez chacune d'entre elles et cliquez sur *Ôter* de manière à obtenir un bloc vide.

Champs

Aucun défini

Figure 12–62
Suppression des champs existants

À présent, cliquez sur le symbole d'ajout et sélectionnez *Noeud : Titre* puis *Ajouter*. Dans le paramétrage du nouveau champ, supprimez le contenu du champ *Étiquette* et cochez *Lier ce champ à son noeud* puis validez.

Paramètres par défaut: Configure champ Noeud : Titre

Étiquette :

L'étiquette de ce champ, qui sera affichée aux utilisateurs finaux si le style l'exige.

☐ Exclure de l'affichage
Cochez cette case pour ne pas afficher ce champ mais le charger néanmoins dans la vue. Utilisez cette option pour ne pas montrer un champ de groupement dans chaque enregistrement ou lorsque vous travaillez sur des thèmes avancés.

☐ Rewrite the output of this field
If checked, you can alter the output of this field by specifying a string of text with replacement tokens that can use any existing field output.

☐ Output this field as a link
If checked, this field will be made into a link. The destination must be given below.

☐ Trim this field to a maximum length
If checked, this field be trimmed to a maximum length in characters.

☒ Lier ce champ à son noeud
This will override any other link you have set.

Mise à jour Annuler Ôter

Figure 12-63
Ajout d'un seul champ Titre, sans étiquette et lié au contenu

Pour terminer, cliquez dans les *Paramètres de base* sur la valeur du champ *Style* pour utiliser une liste HTML (*HTML List*) qui conviendra mieux que notre bloc, puis mettez à jour.

Paramètres par défaut: Style à appliquer à cette vue

☐ Grille

☒ HTML List

☐ Non mis en forme

☐ Tableau

Figure 12-64
Choix d'un format de type Liste

Voilà, notre vue est maintenant prête, comme nous le constatons sur l'aperçu en bas de page.

- Bienvenue
- Un nouvel article
- État du marché de l'emploi pour les ingénieurs informaticiens
- Etat du marché de l'emploi pour les Administrateurs réseaux

Figure 12-65
Notre bloc affiche maintenant les contenus triés par notation.

L'article « Bienvenue » est bien en tête de liste car, pour l'instant, il est le seul à avoir reçu une note.

Notre vue est maintenant correctement paramétrée. Avant de poursuivre, nous allons faire un peu de ménage dans les affichages de la vue dupliquée : sélectionnez l'onglet *Page*, et cliquez sur le bouton *Supprimer un affichage*. Renouvelez l'opération sur l'onglet *Flux* et enregistrez la vue. Les deux onglets inutiles devraient disparaître.

Enfin, nous allons créer un nouvel affichage de type Bloc. Pour cela, sélectionner *Bloc* dans la liste des affichages et cliquez sur le bouton *Nouvel affichage*. Ceci fait, cliquez sur l'onglet *Bloc* pour en changer le titre en « Les meilleurs notes ! », sans oublier de mettre ensuite à jour.

Nous pouvons maintenant enregistrer notre vue, avant de nous rendre dans le panneau de configuration des blocs pour ajouter le bloc *Les meilleurs notes* dans la région Barre de droite. Une fois la disposition des blocs enregistrée, le nouveau bloc devrait apparaître.

Les meilleurs notes

- État du marché de l'emploi pour les ingénieurs informaticiens
- Bienvenue
- État du marché de l'emploi pour les Administrateurs réseaux

Figure 12–66

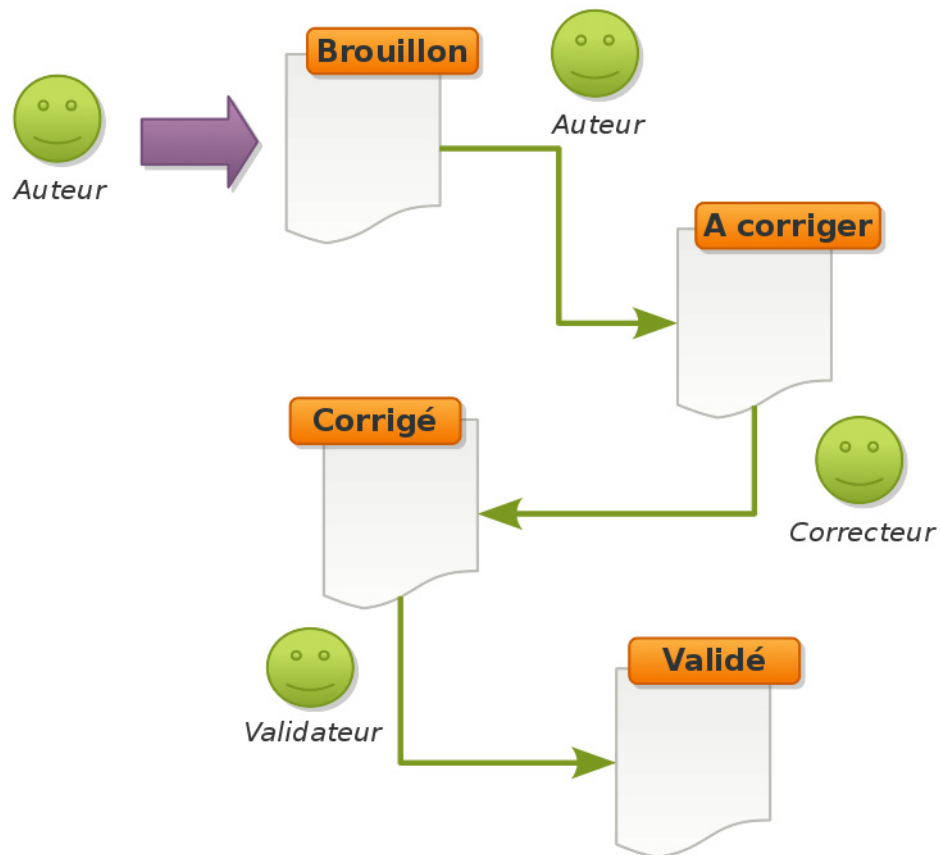
Le nouveau bloc apparaît dans la barre de droite.

Pour tester son fonctionnement, vous pouvez sélectionner le dernier contenu dans la liste, cliquer sur la note maximum, puis presser la touche *F5* pour rafraîchir notre bloc : ce contenu vient de passer en pole position.

En résumé

- Les flux RSS permettent à deux logiciels d'échanger du contenu. Il sont utilisés par des lecteurs de flux RSS pour se tenir au courant des nouveaux contenus (nœuds, commentaires) ou entre deux machines pour alimenter des planets.
- L'utilisation des flux RSS n'est pas encore suffisamment répandue pour se passer des notifications par courriel.
- OpenID est une technologie permettant de centraliser l'authentification et les informations personnelles sur un seul service. En cas d'utilisation d'un identifiant OpenID, Drupal utilise ce service pour créer le compte et authentifier l'utilisateur.
- Les Gravatars sont des portraits associés à des adresses de courriel et centralisés sur des serveurs que Drupal peut utiliser en lieu et place des portraits standards, notamment pour les utilisateurs anonymes.
- L'ensemble des modules permettant une notation de contenu utilise le module Voting API.

chapitre 13



Automatiser les tâches récurrentes

De nombreuses tâches répétitives reviennent à l'administrateur, comme la publication d'un article en attente de validation. Drupal propose une infrastructure de type événement/action pour automatiser un bon nombre de ces tâches, et libérer ainsi du temps pour des actions d'administration à plus forte valeur ajoutée. Nous découvrirons ici comment tout cela est possible grâce à la mise en œuvre d'un flux de travail (*workflow*), dans lequel plusieurs acteurs interviendront tout au long des étapes du cycle de vie d'un article, depuis sa création jusqu'à sa mise en ligne.

SOMMAIRE

- Créer des flux de travaux
- Automatiser vos actions en fonction de déclencheurs

Créer des flux de travaux (workflows)

Un flux de travail (en anglais, *workflow*) consiste dans notre contexte à définir les différents « états » d'un document, ainsi qu'une série de droits accordés à des rôles pour passer ce document d'un état à un autre.

Lors de notre phase de définition du besoin, nous avons décidé que l'auteur d'un article pouvait le faire passer de l'état Brouillon à l'état À corriger. Le correcteur, une fois son travail réalisé, lui attribuera l'état Corrigé. Enfin, le validateur – s'il considère que le document est d'un niveau de qualité suffisant – le fera passer à l'état Validé. À chacune de ces étapes, l'acteur en charge des travaux peut retourner un document à son état précédent. Le correcteur peut ainsi, avant validation, décider de revenir sur son travail, et repasser le statut de l'article en Brouillon afin qu'il soit revu en profondeur par l'auteur.

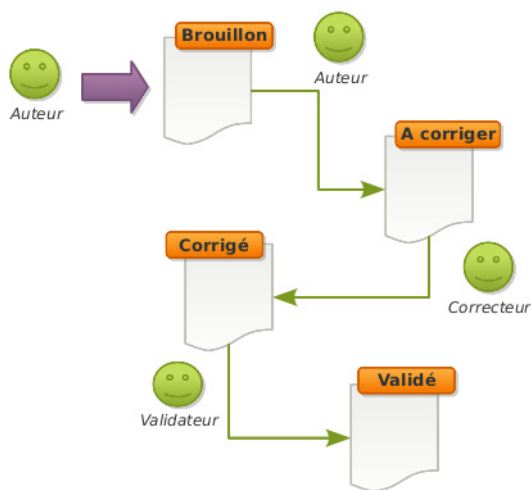


Figure 13–1
Le flux de travail sur un article,
tel que nous l'avons défini.

Il s'agit là d'un exemple très simple – voire simpliste – de flux de travail. Dans un cas réel, nous aurions dû prévoir un état À recorriger, que le validateur utiliserait pour qualifier un contenu présentant encore des fautes. Toutefois, cet exemple est largement suffisant pour comprendre comment mettre en œuvre ces flux de travaux avec Drupal.

Notre première tâche sera de créer deux nouveaux rôles, Validateur et Correcteur, puis d'affecter ces deux rôles à deux utilisateurs qui participeront aux tests, respectivement Karine et Philippe, que nous avons créés dans le chapitre sur la gestion des utilisateurs.

<input type="checkbox"/>	Nom d'utilisateur	Statut	Rôles	Membre depuis	Dernier accès	Opérations
<input type="checkbox"/>	karine	actif	<ul style="list-style-type: none">contributeurvalidateur	4 heures 51 min	il y a 4 heures 51 min	modifier
<input type="checkbox"/>	Philippe	actif	<ul style="list-style-type: none">contributeurcorrecteur	4 heures 52 min	il y a 2 min 42 sec	modifier

Figure 13–2
Karine et Philippe,
respectivement Validateur et Correcteur

Drupal ne gère pas en standard les flux de travaux, aussi allons-nous installer un nouveau module nommé Workflow pour effectuer ce travail. Une fois ce module en place, n’activez que le premier des sous-modules :

URL Module Workflow

► <http://drupal.org/project/workflow>

Workflow			
Activé	Nom	Version	Description
<input checked="" type="checkbox"/>	Workflow	6.x-1.1	Allows the creation and assignment of arbitrary workflows to node types. Requis par : Workflow access (enabled)
<input checked="" type="checkbox"/>	Workflow access	6.x-1.1	Content access control based on workflows and roles. Dépend de : Workflow (enabled)

Figure 13–3
Les deux sous-modules Workflow..

Workflow		
Activé	Nom	Version
<input checked="" type="checkbox"/>	Workflow	6.x-1.1
<input type="checkbox"/>	Workflow access	6.x-1.1

Figure 13–4
... et l’activation du premier sous-module

Suite à l’activation de ce sous-module, nous devons dans la foulée modifier quelques droits en ajoutant la permission *Access workflow summary views* aux contributeurs. De plus, nous ajouterons également la permission *Administrer les nœuds* aux deux nouveaux rôles de manière à ce qu’ils puissent effectuer leur travail sur un article qu’ils n’ont pas eux-mêmes écrit. Pour être franc, cette option est loin d’être parfaite puisque vous donnerez ainsi les droits de modification sur l’ensemble des types de contenu à ces rôles, mais il n’y a malheureusement pas de meilleure solution pour le moment.

Nous pouvons maintenant créer notre premier flux de travail. Pour cela, il faut se rendre dans *Construction du site > Workflow*, puis sur l’onglet *Ajouter un flux de travaux*. Donnez un nom au flux, puis appuyez sur le bouton *Ajouter un flux de travaux*.

Nom du flux de travaux :

Ajouter un flux de travaux

Figure 13–5
Création du nouveau flux de travail

Une fois ce nouveau flux créé, l'étape suivante consiste à définir les différents états qu'un contenu pourra prendre en suivant ce flux de travaux. Nous commencerons par saisir *Brouillon* puis nous cliquerons sur *Enregistrer*.

State name : *

Enter the name for a state in your wo

Poids :

In listings, the heavier states will sink

Figure 13–6
Création de la première étape du flux

Le module nous dirige ensuite sur une liste de flux où apparaît notre flux *Rédaction d'un article*, contenant l'état Brouillon que nous venons de créer. Cliquez sur le lien *Ajouter un état* pour saisir l'état *À corriger* ; enregistrez et renouvelez ensuite l'opération pour les états Corrigé puis Validé. Notre flux est à présent complet.

Workflow	Opérations			
Rédaction d'un article	Ajouter un état	Actions	Modifier	Supprimer
State	Opérations			
(création)				
À corriger	Modifier	Supprimer		
Brouillon	Modifier	Supprimer		
Corrigé	Modifier	Supprimer		
Validé	Modifier	Supprimer		

Figure 13–7
La liste finale des étapes du flux

Nous allons maintenant définir les rôles capables d'opérer les transitions d'un état à l'autre. En cliquant sur le lien *Modifier* se trouvant au même niveau que le libellé *Rédaction d'un article*, vous afficherez la matrice des transitions.

Ne vous laissez pas impressionner par cet imposant amas de cases à cocher qui se lit comme un tableau à double entrée : en ligne sont indiqués les états d'origines, et en colonne les états d'arrivée. À chaque intersection de ligne et de colonne figure la liste des rôles existants. Tout le jeu consiste ensuite à cocher le ou les rôles ayant le droit de faire passer un document de l'état d'origine à celui d'arrivée. La seule notion un peu déroutante est l'état (*création*) : il s'agit là d'un pseudo-état qui n'existe que pour indiquer quel statut doit prendre automatiquement le contenu à la création. Nous saisissons donc les transitions suivantes :

- de l'état (*création*) à l'état *Brouillon*, puis de l'état *Brouillon* à l'état *À corriger*, nous cochons *author* (l'auteur du contenu) ;
- de l'état *À corriger* à *Corrigé*, nous cochons *correcteur* ;
- enfin, de l'état *Corrigé* à *Validé*, nous cochons *validateur*.

De / À →	Brouillon	À corriger	Corrigé	Validé
(création)	<input checked="" type="checkbox"/> author	<input type="checkbox"/> author	<input type="checkbox"/> author	<input type="checkbox"/> author
	<input type="checkbox"/> anonymous user	<input type="checkbox"/> anonymous user	<input type="checkbox"/> anonymous user	<input type="checkbox"/> anonymous user
	<input type="checkbox"/> authenticated user	<input type="checkbox"/> authenticated user	<input type="checkbox"/> authenticated user	<input type="checkbox"/> authenticated user
	<input type="checkbox"/> contributeur	<input type="checkbox"/> contributeur	<input type="checkbox"/> contributeur	<input type="checkbox"/> contributeur
	<input type="checkbox"/> correcteur	<input type="checkbox"/> correcteur	<input type="checkbox"/> correcteur	<input type="checkbox"/> correcteur
	<input type="checkbox"/> valideur	<input type="checkbox"/> valideur	<input type="checkbox"/> valideur	<input type="checkbox"/> valideur
Brouillon		<input checked="" type="checkbox"/> author	<input type="checkbox"/> author	<input type="checkbox"/> author
		<input type="checkbox"/> anonymous user	<input type="checkbox"/> anonymous user	<input type="checkbox"/> anonymous user
		<input type="checkbox"/> authenticated user	<input type="checkbox"/> authenticated user	<input type="checkbox"/> authenticated user
		<input type="checkbox"/> contributeur	<input type="checkbox"/> contributeur	<input type="checkbox"/> contributeur
		<input type="checkbox"/> correcteur	<input type="checkbox"/> correcteur	<input type="checkbox"/> correcteur
		<input type="checkbox"/> valideur	<input type="checkbox"/> valideur	<input type="checkbox"/> valideur
À corriger	<input type="checkbox"/> author		<input type="checkbox"/> author	<input type="checkbox"/> author
	<input type="checkbox"/> anonymous user		<input type="checkbox"/> anonymous user	<input type="checkbox"/> anonymous user
	<input type="checkbox"/> authenticated user		<input type="checkbox"/> authenticated user	<input type="checkbox"/> authenticated user
	<input type="checkbox"/> contributeur		<input type="checkbox"/> contributeur	<input type="checkbox"/> contributeur
	<input type="checkbox"/> correcteur		<input checked="" type="checkbox"/> correcteur	<input type="checkbox"/> correcteur
	<input type="checkbox"/> valideur		<input type="checkbox"/> valideur	<input type="checkbox"/> valideur
Corrigé	<input type="checkbox"/> author	<input type="checkbox"/> author		<input type="checkbox"/> author
	<input type="checkbox"/> anonymous user	<input type="checkbox"/> anonymous user		<input type="checkbox"/> anonymous user
	<input type="checkbox"/> authenticated user	<input type="checkbox"/> authenticated user		<input type="checkbox"/> authenticated user
	<input type="checkbox"/> contributeur	<input type="checkbox"/> contributeur		<input type="checkbox"/> contributeur
	<input type="checkbox"/> correcteur	<input type="checkbox"/> correcteur		<input type="checkbox"/> correcteur
	<input type="checkbox"/> valideur	<input type="checkbox"/> valideur		<input checked="" type="checkbox"/> valideur
Validé	<input type="checkbox"/> author	<input type="checkbox"/> author	<input type="checkbox"/> author	
	<input type="checkbox"/> anonymous user	<input type="checkbox"/> anonymous user	<input type="checkbox"/> anonymous user	
	<input type="checkbox"/> authenticated user	<input type="checkbox"/> authenticated user	<input type="checkbox"/> authenticated user	
	<input type="checkbox"/> contributeur	<input type="checkbox"/> contributeur	<input type="checkbox"/> contributeur	
	<input type="checkbox"/> correcteur	<input type="checkbox"/> correcteur	<input type="checkbox"/> correcteur	
	<input type="checkbox"/> valideur	<input type="checkbox"/> valideur	<input type="checkbox"/> valideur	

Figure 13–8
L'impressionnante matrice de passage pour notre flux de travail

Notre flux de travail étant complet, nous pouvons l'enregistrer pour retourner sur la liste des flux. Nous allons cette fois nous intéresser à la seconde partie de la page qui nous permet d'associer à chacun des types de contenu un flux. Dans notre cas, nous allons donc choisir *Rédaction d'un article* pour le type de contenu Article et cliquer sur le bouton *Save workflow mapping*.

Content Type	Workflow	Display Workflow Form for:
Article	<div>Rédaction d'un article</div>	<input checked="" type="checkbox"/> Post <input type="checkbox"/> Commentaire
Billet	<div><Aucun></div>	<input checked="" type="checkbox"/> Post <input type="checkbox"/> Commentaire

Figure 13–9
Affectation du flux au type de contenu Article

Notre flux est maintenant opérationnel et prêt à être testé.

Pour cela, nous devons d'abord nous déconnecter, puis nous reconnecter en tant que Marine pour créer un type de contenu Article. Comme vous le constatez, le formulaire ne change en rien nos habitudes. Marine peut saisir son contenu et le valider comme à l'accoutumée.

Créer 'Article'

Titre : *

Domaines de Compétences : *

- Ressources Humaines
- Recrutement
- Bilans de compétence
- Systèmes d'information
- Systèmes et réseaux
- Assistante à maîtrise d'ouvrage
- Ingénierie logiciel

Corps :

B I U ABC [Liste à puces] [Liste numérotée] [Image] [Citation] [Smiley] [16/19]

... à publier...

Figure 13-10
Création d'un article

Si nous jetons un coup d'œil sur le menu de navigation *Marine*, nous constatons la présence d'un nouveau menu *Workflow summary*. Cliquez dessus pour faire apparaître une liste de contenus comme celle dont nous disposons dans *Gestion des contenus*, mais notez cette fois la présence d'une colonne indiquant l'état de chacun des documents. Dans la liste *Current State*, sélectionnez l'état *Brouillon* et cliquez sur le bouton *Appliquer*.

Workflow summary

Current State **Content Type**

Current state	Titre	Type
Rédaction d'un article: Brouillon	Un article dans le flux	Article

Figure 13-11
Le liste des contenus à l'état Brouillon

Nous retrouvons ici l'article que Marine vient de créer, qui se trouve en toute logique dans l'état Brouillon. Nous allons à présent passer cet article au correcteur et pour cela, cliquez sur le lien *Un article dans le flux*, puis sur l'onglet *Workflow*.

L'onglet *Workflow* a été ajouté par le module du même nom pour permettre à l'utilisateur de changer l'état d'un document, dans la limite des droits qui lui ont été attribués. En l'occurrence, Marine, en tant que Contributeur, a uniquement le droit de passer l'article à l'état *À corriger*, en ajoutant éventuellement un commentaire avant de cliquer sur le bouton *Soumettre*. Si Marine clique à nouveau sur l'onglet *Workflow*, elle constatera qu'aucune autre action ne lui est autorisée.

Un article dans le flux

Voir Modifier Workflow

Current state: **Brouillon**

Changer l'état de *Rédaction d'un article* :

☒ Brouillon

☐ À corriger

Commentaire :

Un commentaire à intégrer dans le journal du flux de travaux

Soumettre

Historique du flux de travaux

Date	Ancien état	Nouvel état	Par	Commentaire
ven, 06/26/2009 - 01:12 (création)		Brouillon	Marine	

Figure 13–12
Passage de l'article à l'état À corriger

Un article dans le flux

Voir Modifier Workflow

Current state: **À corriger**

Historique du flux de travaux

Date	Ancien état	Nouvel état	Par	Commentaire
ven, 06/26/2009 - 01:31	Brouillon	À corriger	Marine	
ven, 06/26/2009 - 01:31 (création)		Brouillon	Marine	

Figure 13–13
Historique des passages d'état

Déconnectons-nous et reconnectons-nous en tant que Philippe, qui a le rôle de Correcteur. La première action effectuée par Philippe sera d'aller lui aussi sur son lien *Workflow summary* pour collecter tous les articles à l'état À corriger.

Il clique lui aussi sur le lien *Un article dans le flux*, puis sur l'onglet *Modifier* pour effectuer ses corrections. Une fois cela fait, il clique sur l'onglet *Workflow*. Notez alors l'évolution de l'historique des flux de travaux.

Philippe peut maintenant sélectionner l'état *Corrigé* et cliquer sur le bouton *Soumettre*. Vient alors le tour de Karine, qui possède le rôle Valideur et effectuera les mêmes opérations pour cette fois-ci passer le document à l'état *Validé*.

Le flux étant à présent achevé, l'administrateur peut alors collecter les articles à l'état Validé et changer leur statut en *Publié* et *Promu en page d'accueil*. L'article est enfin visible aux yeux de tous.

Actions et déclencheurs

Principe de fonctionnement

La gestion des flux que vous venez d'expérimenter permet d'effectuer un travail collaboratif efficace, en fluidifiant le processus de traitement d'un document entre les différents intervenants.

Il reste cependant un aspect agaçant car peu optimal : pourquoi l'administrateur doit-il effectuer cette tâche répétitive et sans valeur ajoutée, consistant à publier un contenu qui vient d'être validé ? Fort heureusement, cette action peut être automatisée grâce à deux modules intégrés à Drupal : Action et Trigger. Ces derniers n'étant pas activés par défaut, vous devez le faire dès à présent.

- Le module Action vous fournit la possibilité d'effectuer un certain nombre d'actions, comme envoyer un courriel, changer l'état d'un contenu, publier un contenu, supprimer un commentaire, etc.
- Le module Trigger vous permet d'associer un déclencheur – c'est-à-dire un événement dans le système Drupal – à une action donnée. Par exemple, si nous désirons envoyer un courriel à l'auteur d'un contenu dès qu'un nouveau commentaire apparaît, nous allons associer le déclencheur « enregistrement d'un nouveau commentaire » à l'action « envoyer un courriel ». Le module Trigger fournit ainsi en standard une vingtaine de déclencheurs qu'il est possible d'associer à autant d'actions fournies par le module Action.

Plus intéressant encore, les autres modules peuvent à leur convenance fournir à ces deux modules de nouvelles actions et de nouveaux déclencheurs. C'est bien évidemment le cas du module Workflow, et pour illustrer notre propos, prenons ce problème d'automatisation du passage au statut Publié d'un contenu venant d'être validé.

Création d'une action simple

Pour créer ce comportement, nous devons trouver le bon déclencheur. Pour cela, allez dans *Construction du site*, puis *Déclencheurs* (voir figure 13-14).

Vous êtes ici dans le centre d'association des déclencheurs connus du module Trigger avec l'une des actions fournies par le module Action. Chaque onglet représente une catégorie de déclencheurs particuliers. Cliquez sur l'onglet *Workflow*.

Déclencheurs

Commentaires

Content

Cron

Simplenews

Taxonomie

UR-API

Utilisateurs

Workflow

Les déclencheurs sont des événements systèmes, comme par exemple quand un nouveau contenu est ajouté ou qu'un utilisateur s'identifie. Le module Trigger permet de combiner ces déclencheurs avec des actions (tâches qui exécutent des fonctions), comme par exemple 'retirer un contenu publié' ou 'envoyer un email à l'administrateur'. La page de réglage des actions liste les actions disponibles et permet de créer et de configurer des actions supplémentaires.

Vous pouvez ci-dessous associer des actions à des déclencheurs liés aux contenus. Vous pouvez par exemple envoyer un email à l'administrateur lorsqu'un contenu est créé ou mis à jour.

Déclencheur : Lors de l'enregistrement ou de la mise à jour d'un contenu

Choisir une action

Associer

Déclencheur : Après l'enregistrement d'un nouveau contenu

Choisir une action

Associer

Déclencheur : Après la mise à jour d'un contenu existant

Choisir une action

Associer

Déclencheur : Après la suppression d'un contenu

Choisir une action

Associer

Déclencheur : Lorsqu'un contenu est affiché par un utilisateur identifié

Choisir une action

Associer

Figure 13-14
La liste des déclencheurs disponibles

Déclencheurs

Commentaires

Content

Cron

Simplenews

Taxonomie

UR-API

Utilisateurs

Workflow

Use this page to set actions to happen when transitions occur. To configure actions, use the actions settings page.

Déclencheur : When story moves from Rédaction d'un article: (création) to Rédaction d'un article: Brouillon

Choisir une action

Associer

Déclencheur : When story moves from Rédaction d'un article: Brouillon to Rédaction d'un article: À corriger

Choisir une action

Associer

Déclencheur : When story moves from Rédaction d'un article: À corriger to Rédaction d'un article: Corrigé

Choisir une action

Associer

Déclencheur : When story moves from Rédaction d'un article: Corrigé to Rédaction d'un article: Validé

Choisir une action

Associer

Figure 13-15
Les déclencheurs spécifiques aux flux de travaux

Nous trouvons ici l'ensemble des changements d'état de notre flux de travail Rédiger un article, avec dans cette liste le déclencheur qui nous intéresse : *When story moves from Rédaction d'un article: Corrigé to Rédaction d'un article: Validé* (littéralement, « lorsqu'un article passe de l'état Rédaction d'un article : Corrigé à l'état Rédaction d'un article : Validé »).

Dans la liste *Choisir une action*, nous allons sélectionner, sous la catégorie *Node*, *Publier le contenu*, puis cliquer sur le bouton *Associer*.

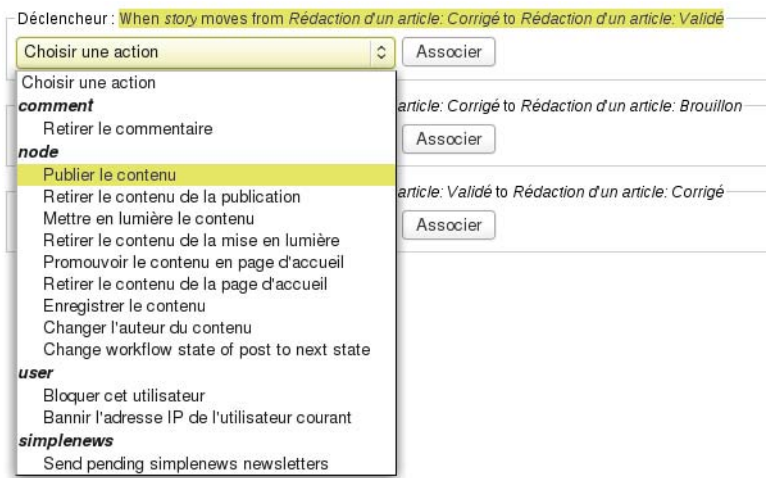


Figure 13-16
Ajout d'une action au déclencheur
Passage de l'état Corrigé à Validé

Renouvelez l'opération avec cette fois *Promouvoir le contenu en page d'accueil*.

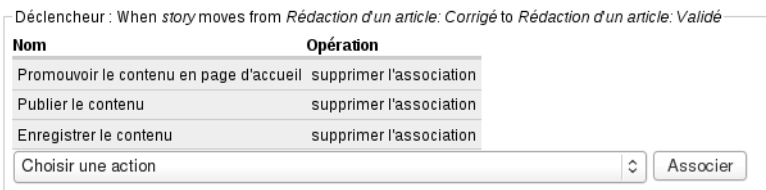


Figure 13-17
Il est possible d'ajouter plusieurs actions
à un même déclencheur.

Notre association est maintenant complète. Nous pouvons la tester en recréant un nouvel article puis en l'emmenant jusqu'à son état Publié pour constater que son statut aura effectivement changé. Sans aucune intervention de l'administrateur, l'article sera en page d'accueil.



Figure 13-18
Notre contenu a bien été automatiquement
publié et promu en page d'accueil.

Création d'une action avancée

Dans le même cas de figure, en supposant que Philippe est le seul correcteur de l'équipe, il peut être judicieux de le prévenir par courriel lorsqu'un nouveau contenu doit être traité. L'ennui est que vous ne trouverez pas d'action intitulée « Envoyer un courriel à Philippe ».

Il nous faut donc la créer nous-même, et pour cela nous rendre dans *Configuration du Site* > *Actions*. Le module Action nous donne sur cette page l'ensemble des « actions simples », c'est-à-dire ne nécessitant pas de paramétrage. Nous y retrouvons les deux actions que nous venons d'utiliser.

L'aspect le plus intéressant se situe à la fin de la page, où vous disposez de la possibilité de créer une « action avancée ».

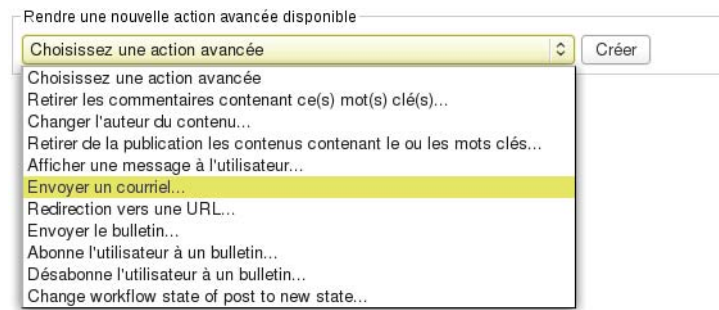


Figure 13-19
Sélection d'un action complexe
et donc paramétrable

Dans cette liste, nous trouvons *Envoyer un courriel*. Sélectionnez cette action et appuyez sur le bouton *Créer*. Cela nous mène à un formulaire dans lequel vous pouvez spécifier une description, un destinataire, l'objet du courriel et le corps du message. Saisissez les éléments suivants :

Description :

Description unique pour cette action avancée. Cette description sera affichée avec les actions : Déclencheur.

Destinataire :

L'adresse de la boîte aux lettres électronique à laquelle le message doit être envoyé OU entre original.

Objet :

Sujet du message.

Message :

Figure 13-20
Paramétrage de l'action Envoyer un courriel

Figure 13-21
Notre nouvelle action est maintenant
intégrée parmi les autres.

Rien d'extraordinaire dans ce message mis à part la variable `%node_url` qui sera automatiquement remplacée par l'URL pointant vers le contenu à corriger. Une liste de ces variables est visible sous la zone d'édition. Cliquez sur *Enregistrer*, et voilà votre nouvelle action disponible.

node	Change workflow state of post to next state
simplenews	Send pending simplenews newsletters
system	Notification à Philippe d'un nouveau contenu à corriger configurer supprimer
user	Bloquer cet utilisateur

Vous pouvez bien évidemment *supprimer* ou *modifier* la configuration de cette nouvelle action, mais ce n'est pas nécessaire dans le cas présent. Retournons alors sur la page d'association des déclencheurs dans notre onglet *Workflow* pour associer cette nouvelle action avec le changement d'état Brouillon vers À corriger.

Figure 13-22
Association du déclencheur Brouillon vers À
corriger avec notre action

Déclencheur : When story moves from Rédaction d'un article: Brouillon to Rédaction d'un article: À corriger

Notification à Philippe d'un nouveau contenu à corriger

Associer

Il suffit dès lors de cliquer sur *Associer* pour activer la nouvelle règle, et permettre à Philippe de recevoir son courriel sans avoir à écumer la page *Workflow summary*.

De la même manière, nous pouvons créer une action avancée « Envoyer un courriel à l'auteur d'un contenu lorsqu'un commentaire y est publié ». Le travail à effectuer est identique à celui que nous venons de réaliser, à la différence près que l'adresse de destination sera cette fois-ci `%author`.

Quant au déclencheur, vous le trouverez dans l'onglet *Commentaires* sous le nom de *Après l'enregistrement d'un nouveau commentaire*. Il ne vous reste plus qu'à créer l'association puis ajouter un commentaire sur un article afin de vérifier que Marine reçoit bien sa notification.

Figure 13-23
L'action est maintenant associée.

Déclencheur : Après l'enregistrement d'un nouveau commentaire

Nom	Opération
Envoyer un courriel à l'auteur d'un contenu lorsqu'un commentaire y est publié	supprimer l'association

Choisir une action

Associer

Ne vous étonnez cependant pas si Marine reçoit deux messages, car nous venons par cette association de mimer purement et simplement le fonctionnement du module Comment notify que nous avons activé lors du chapitre précédent...

MODULES MAGIQUES Créer des règles conditionnelles

Les modules Action et Trigger permettent de faire énormément de choses mais restent malgré tout limités par l'absence de conditions. Il n'est par exemple pas possible de créer une association entre le déclencheur « Un contenu a été modifié » et l'action « Envoyer un courriel à l'auteur du contenu » avec la condition « si le statut du contenu est passé à Modifié ».

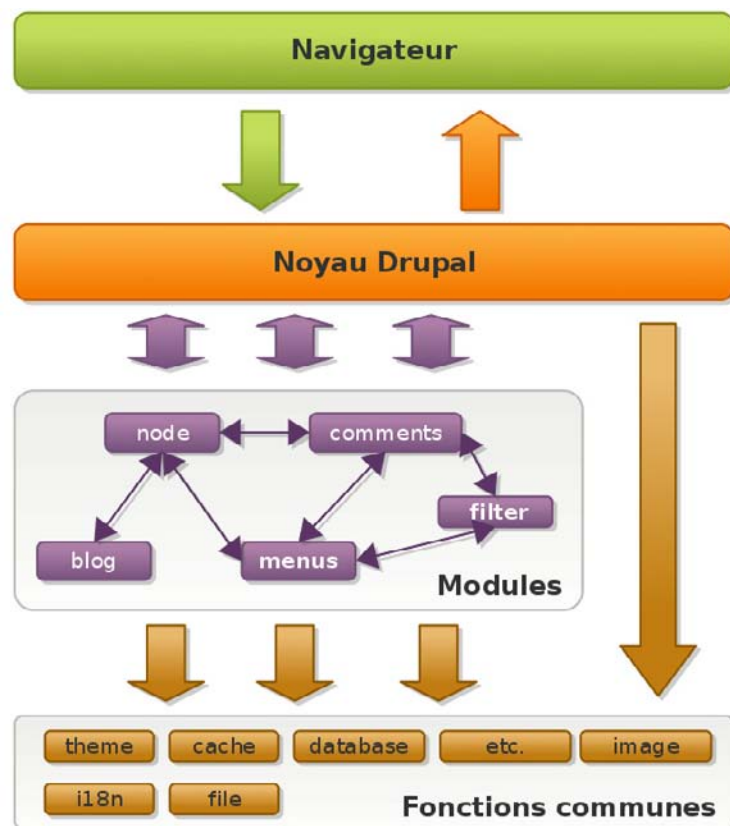
Si vous avez besoin de ce genre d'action, vous devez regarder de près le jeune module Rules qui a pour objectif de remplacer les modules Workflow, Action et Trigger par un système de règles évoluées.

► <http://drupal.org/project/rules>

En résumé

- Le module Workflow permet de créer des flux de travaux et de les associer à des types de contenu.
- Le module Action vous fournit des actions prêtes à l'emploi, ainsi que la possibilité de créer des actions complexes. De nombreux modules ajoutent de nouvelles actions.
- Le module Trigger a pour rôle de fournir une liste d'événements et de vous permettre d'associer une série d'actions à un événement donné.

chapitre 14



Écrire son premier module

Une des grandes forces de Drupal réside dans son architecture fondée sur des modules. Que ce soit pour la gestion des blogs ou celle d'un forum, chaque fonction fondamentale est en réalité un simple module interagissant avec le cœur de Drupal. Si les modules fournis en standard ne suffisent pas, nous l'avons à présent bien assimilé, des milliers d'autres sont disponibles et couvrent à peu près tous les usages. Malgré cette richesse, il arrive cependant que l'on ne trouve pas *le* module correspondant parfaitement à notre besoin. Pourquoi alors ne pas le fabriquer soi-même et découvrir ainsi comment Drupal s'adapte facilement à toutes les situations ?

SOMMAIRE

- Comprendre la structure d'un module
- Cerner le fonctionnement d'un module au sein de Drupal
- Créer et activer son premier module

Prérequis

Les seuls prérequis pour développer un module sont une bonne connaissance des concepts clés de Drupal (blocs, types de contenu, contenus, taxonomie, etc.) ainsi qu'une relative maîtrise de PHP et des bases de données.

Armés de tout cela, nous allons commencer en douceur avec la création d'un module dont le but sera de modifier à l'affichage tous nos nœuds pour y ajouter un petit message. Ce chapitre a pour but de vous montrer les possibilités qui vous sont offertes et de vous permettre de prendre conscience que la création de modules est une alternative simple à mettre en œuvre. Même si vous ne les développez pas vous-même, savoir comment ils fonctionnent et comment ils sont structurés permet d'éviter de commettre l'erreur classique d'un trop grand volume de code dans le thème alors qu'il aurait parfaitement sa place dans un module autonome et facile à manipuler.

Pourquoi faire son propre module ?

Contrairement à d'autres CMS du marché, Drupal est modulaire « à la racine », et ceci depuis ses débuts. Nous sommes loin d'un système qui s'apparenterait à des greffons mais plus proche d'une architecture de type microkernel. Comme les systèmes d'exploitation de ce type, Drupal charge lors de sa phase de démarrage l'ensemble des modules activés en mémoire et les laisse communiquer entre eux pour effectuer une tâche donnée.

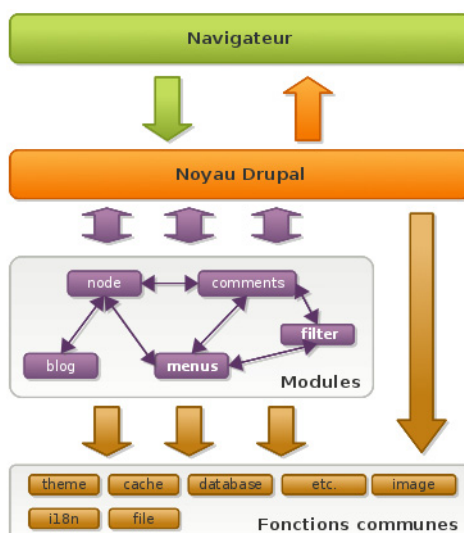


Figure 14–1
Fonctionnement interne
des modules de Drupal

Ainsi, comme nous l'avons vu, la grande majorité des fonctions CMS de Drupal sont à la charge des modules, y compris les fondamentaux comme la gestion des nœuds, des commentaires, des flux RSS, etc. Comme un système d'exploitation standard, le noyau et les modules disposent d'un ensemble de fonctions spécialisées et groupées par famille (cache, courriels, transactions HTTP, images, etc.), permettant d'accélérer le développement en évitant la duplication de code, et donc de diminuer le nombre d'erreurs.

Une fois le travail achevé, lorsque la page HTML est produite, le noyau Drupal va éteindre puis décharger chacun des modules actifs, pour finalement lui-même s'arrêter. À ce stade, la différence avec un système d'exploitation classique est que Drupal effectue ce cycle démarrage/chargement/traitement/déchargement/arrêt pour chaque page fabriquée. On comprend alors mieux l'importance de ne pas activer plus de modules que nécessaire car leur temps de chargement et déchargement s'additionne au temps de génération de toutes les pages affichées.

Dans ce type d'architecture, la nature distribuée des traitements entraîne un fort volume d'échanges entre modules. Ce point constitue généralement le goulot d'étranglement pour les performances des systèmes à base de micronoyaux. Avec Drupal, le problème est réglé simplement et efficacement par l'utilisation de fonctions PHP. Il n'y a donc plus aucun inconvénient particulier dans cette approche, et le fait que Drupal soit l'un des CMS les plus véloces de sa catégorie n'est en cela pas anodin.

Toujours est-il que cette approche « modulaire » depuis l'origine a eu pour résultat une extrême richesse des fonctions que les modules peuvent exploiter. Contrairement aux CMS qui utilisent une vision « greffons » et qui prévoient des points d'attache possibles, Drupal n'est au fond *que* points d'attache. Il est ainsi très rare de ne pas pouvoir modifier son comportement par un module correctement développé.

En résumé, les modules étant un des grands atouts de ce CMS, il serait dommage de se limiter aux créations des autres tant il est facile de fabriquer les siens, totalement adaptés à ses besoins. Pour aller plus loin encore, tous les modules de Drupal étant sous licence GPL, savoir comment fabriquer un module revient aussi à savoir comment le modifier pour récupérer par exemple la fonction qui vous intéresse, plutôt que d'embarquer l'usine à gaz au grand complet...

Des modules et des hooks

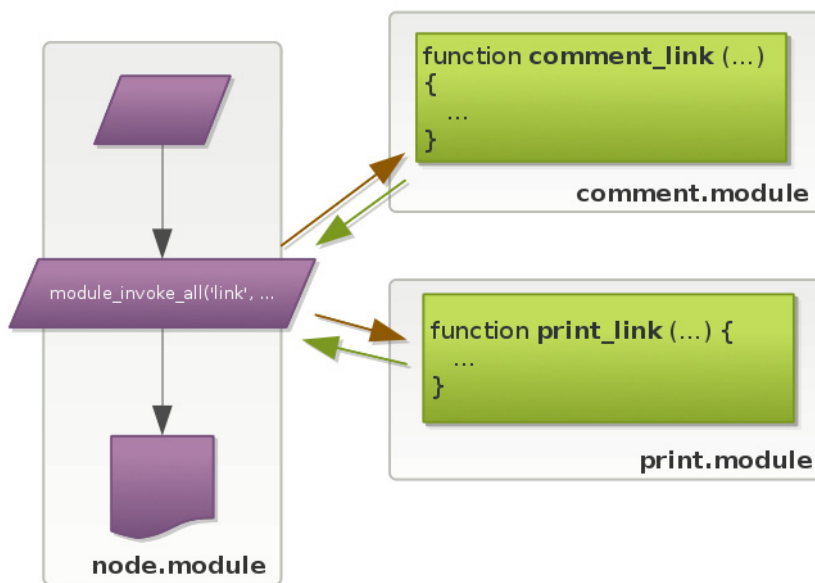


Figure 14–2
Fonctionnement des hooks de module

Un module Drupal n'est rien d'autre qu'un bout de code PHP que l'on peut activer ou désactiver à volonté et dont le seul rôle est d'interagir avec les autres modules de Drupal.

Les interactions entre modules se font sur un mode participatif. En effet, l'idée fondamentale est qu'au cours d'un de ses traitements, un module peut demander à d'autres modules une contribution.

Pour mieux comprendre, prenons un exemple. Comme nous le savons déjà, le module qui gère les contenus (nœuds) s'appelle Node. Un de ses rôles consiste à transformer un contenu stocké dans la base de données en un contenu affichable en HTML. Une des étapes de ce traitement consiste à ajouter les liens qui se trouvent sous le contenu (*répondre au commentaire*, *afficher la version imprimable*, etc.).

Le module Node, au moment de créer la liste des liens, demande à tous les modules si l'un d'eux a un lien à ajouter. Un des modules qui répondra présent est le module Comment ; il gère les commentaires et insérera donc le lien *Ajouter un commentaire*. De même, le module Print lui aussi répondra pour ajouter à son tour les liens *version PDF*, *version imprimable* et *envoyer à un ami*. Et ainsi de suite.

Pour réussir à mettre en œuvre cette collaboration, Drupal a introduit, comme pour les thèmes, le concept de *hook* (en français crochet). Chaque module peut ainsi déclarer un nombre arbitraire de hooks disponibles pour

les autres modules qui sont généralement décrits dans sa documentation. Dans notre exemple, le hook proposé par le module Node s'appelle `Link` et tous les modules qui veulent répondre à ce hook déclareront une fonction dont le nom sera composé du nom du module suivi du nom du hook, `link`, séparés par un caractère souligné. Ainsi, le module Comment et le module Print ont tous deux implémenté une fonction `comment_link` et `print_link`. Une telle implémentation ressemblera à ceci :

Un exemple d'implémentation de `hook_link`

```
function comment_link($type, $node = NULL, $teaser = FALSE) {
  $links=array();
  ...
  return $links;
}
```

Lorsque le module Node aura besoin de la participation des autres modules pour ajouter ses liens, il va simplement demander à Drupal de prendre chaque nom de module activé pour regarder s'il existe une fonction composée de ce nom suivi de `_link`. Ensuite, Drupal appellera ces fonctions les unes après les autres. Du côté du module Node, cet appel va donc ressembler à ceci :

Exemple d'invocation de toutes les implémentations de `hook_link`

```
$node->links = invoke_all('link', $node, $teaser);
```

`invoke_all` est la fonction de Drupal qui ira chercher les hooks de type `link` dans tous les modules activés puis appeler ces fonctions `link` – si elles existent – avec les paramètres qui sont donnés. À ce titre, cette fonction ressemble beaucoup à la fonction `theme()` que nous avons déjà abordée.

Chaque module fournira ainsi son tableau de liens que Drupal va agréger pour fournir au module Node un seul tableau contenant tous les liens.

Cependant, qui dit appel de fonction dit paramètres. Il ne suffit donc pas de connaître le nom du hook pour pouvoir l'implémenter, encore faut-il savoir quels paramètres le module l'appelant va nous transmettre. Pour cela, il faut lire la documentation de Drupal, ou pour aller plus vite, il est aussi possible de saisir dans votre moteur de recherche `hook_` suivi du nom du hook, par exemple `hook_block`. Faites cependant attention à bien sélectionner l'onglet correspondant à votre version de Drupal, car d'une version à l'autre, les paramètres et les hooks eux-mêmes peuvent changer.

BON À SAVOIR Google à la rescousse

Une astuce toute simple pour avoir la documentation d'un hook particulier est de saisir sa référence dans Google (ex : `hook_menu`) et d'utiliser le bouton *J'ai de la chance*. Cela vous mènera à tous les coups à la documentation de Drupal. Faites cependant attention à l'onglet sélectionné ; les API changent avec la version, sélectionnez donc bien celui correspondant à la version sur laquelle vous travaillez.

URL Documentation de Drupal

► <http://api.drupal.org/api/group/hooks>

Structure interne d'un module

Organisation générale

Un module est donc un fournisseur d'implémentation de hooks. Par conséquent, la majeure partie de son code s'insère dans les traitements d'un autre module, et donc au final, de Drupal, pour en étendre le comportement.

De manière plus prosaïque, un module est simplement un dossier, comme un thème, qui doit porter le nom du module, ici `mon_module`, et qui doit être placé dans le dossier `sites/all/modules/mes_modules`.

À l'intérieur de celui-ci, vous devez avoir au minimum deux fichiers portant le même nom de base que le dossier parent : `mon_module.info` et `mon_module.module`. Comme nous le verrons plus loin, le premier fichier contient des informations sur le module (comme pour le thème), le second son code PHP (un peu comme `template.php`).

Le fichier de description du module

Le fichier `mon_module.info` est un simple fichier texte contenant des informations sur le module comme son nom, sa description, la version de Drupal avec laquelle il est compatible, etc. Ce qui nous donne pour notre exemple, le contenu suivant :

Le fichier `mon_module.info`

```
name = "Mon Module"
description = "Un exemple simple de module."
package = "Mes Modules"
project = "mon_module"
version = "6.x-1.0"
core = 6.x
```

- `package` indique dans quelle catégorie le module doit être affiché sur la page d'administration des modules.
- `name` et `description` sont des informations textuelles qui seront affichées dans le panneau d'administration des modules. `name` reprend le nom du module (qui est aussi le nom du dossier).
- `version` indique la version du module. C'est d'ailleurs plus une norme qu'autre chose, qui se lit « Module version 1.0 pour Drupal 6.x ».
- `core` prévient Drupal que le module est seulement compatible avec sa version 6.x. Toute autre version de Drupal empêchera le module de s'activer.

Si nous avons eu besoin de définir une dépendance avec un autre module, nous aurions ajouté une clause de la forme `dependencies[] = autre_module`. Enfin, si vous désirez ajouter des commentaires, il suffit d'utiliser le symbole `;` suivi du texte du commentaire.

Le code du module

Le second fichier est le code PHP du module. C'est lui qui hébergera les hooks que nous souhaitons implémenter. Pour l'instant, nous allons faire simple et n'en mettre aucun. Notre fichier contiendra donc une simple balise de démarrage de code PHP :

Le fichier `mon_module.module` pour l'instant réduit au minimum

```
<?php
```

Nous avons maintenant un module complet et certes inutile, mais que Drupal est cependant capable de voir et d'activer. Pour s'en convaincre, il suffit d'aller faire un tour dans l'administration des modules pour le voir apparaître dans la liste.

▼ Mes Modules			
Activé	Nom	Version	Description
<input type="checkbox"/>	Mon Module	6.x-1.0	Un exemple simple de module.

Figure 14-3
Activation de notre module

Vous constaterez que les informations affichées sont les champs `name` et `description` de notre fichier `mon_module.info`.

À ce stade, nous pourrions activer notre module mais ne le faites pas tout de suite : nous avons encore quelque chose à ajouter.

Implémentation d'un hook

Ce module a pour seul but de nous permettre de comprendre comment tout cela fonctionne, aussi l'idée n'est pas de le doter d'un comportement extraordinaire. Nous allons simplement utiliser le hook `Link` du module `Node` dont nous parlions plus haut de manière à ajouter à chaque nœud un nouveau lien `modifier` qui permettra d'éditer le contenu du nœud courant.

Comme nous l'avons vu, le hook `Link` est défini de la manière suivante dans la documentation de Drupal (http://api.drupal.org/api/function/hook_link) :

La structure d'appel de `hook_link`

```
hook_link($type, $object, $teaser = FALSE)
```

BON À SAVOIR Les normes de codage

Vous aurez sûrement remarqué que la balise PHP n'est pas fermée ici. Tout d'abord sachez que la fermeture de cette balise est optionnelle (<http://fr3.php.net/basic-syntax.instruction-separation>). Mais elle est surtout déconseillée pour la rédaction de code Drupal, car son utilisation peut induire des lignes blanches non voulues qui seront insérées dans flux de sortie. Comme le code des modules est inséré avant toute autre exécution, cela aurait pour effet d'empêcher Drupal d'écrire la définition des en-têtes HTTP et provoquerait une erreur. Pour plus d'informations sur les normes de codage utilisées au sein du projet Drupal :

► <http://drupal.org/coding-standards/>

Ce qui veut dire que nous allons devoir ajouter à notre fichier `mon_module.module` une fonction définie comme ceci :

Notre implémentation de `hook_link`

```
function mon_module_link($type, $object, $teaser = FALSE) {
  ...
}
```

Notre implémentation du hook étant nommée `<nom du module>_{nom du hook}`, Drupal la reconnaîtra lorsque le module Node aura besoin de l'appeler.

Même si ce n'est pas le principal sujet de ce chapitre, signalons que ce hook est très pratique pour ajouter de nouveaux liens sous les contenus, mais aussi sous les commentaires. En effet, le paramètre `$type` permet de faire la distinction entre les types de lien demandés : `$type=='comment'` ou `$type=='node'`. Le second paramètre `$object` sera donc soit un nœud, soit un commentaire selon la valeur de `$type`. Ici, c'est le type `'node'` qui nous intéresse.

Le dernier paramètre permet quant à lui de décider si l'on souhaite placer le lien sous le contenu dans sa forme résumée (`$teaser==TRUE`) – par exemple sur la page de garde du site –, ou dans sa forme complète. Nous avons à présent tout ce qu'il nous faut pour ajouter un lien à tous les contenus.

Le contenu de `mon_module.module`

```
<?php
function mon_module_link($type, $object, $teaser = FALSE) {
  // Seulement si l'utilisateur a les droits d'administration
  if (!user_access("administer site configuration")) {
    return;
  }

  $links = array ();

  // Si l'objet est un node
  if ($type == 'node') {
    // On ajoute un nouveau lien
    $links['editer-node'] = array (
      'title' => t('Edit'),
      'href' => "node/" . $object->nid . "/edit"
    );
  }

  // On renvoie les liens au module Node
  return $links;
}
```

La première chose que nous faisons ici est de vérifier que l'utilisateur a bien le droit de modifier le contenu. Si nous ne l'avions pas fait, le lien aurait été visible y compris pour les visiteurs anonymes. Même si ces derniers aboutiraient sur une erreur « accès interdit », cela ne serait pas très élégant.

Ensuite, après nous être assuré que la demande vise bien un nœud (`$type=='node'`), il suffit de créer un tableau associatif dont le contenu est une série de tableaux représentant chacun un lien. Ici, nous n'en avons qu'un mais rien n'empêche d'en mettre plusieurs. Dans tous les cas, c'est Drupal qui se charge de transformer ce tableau en la balise ``.

Notez que l'on ne met pas de *slash* au début de l'attribut `href`. Ceci permet à Drupal de choisir lui-même la meilleure représentation possible pour l'URL et d'opter pour un éventuel alias.

Notez enfin l'attribut `title` qui, comme son nom l'indique, contient le titre du lien. Vous y retrouvez l'utilisation de la fonction `t(...)` qui va traduire la chaîne de caractères « Edit » dans la langue active de l'interface.

MODULES MAGIQUES Des modules qui aident à développer des modules

Il existe un module appelé Coder qui permet de tester la validité de votre code et vous donne ainsi une meilleure assurance pour construire des modules compatibles avec les règles de développement de Drupal.

► <http://drupal.org/project/coder>

Il existe aussi un module appelé Schema qui vous apportera une grande aide pour développer des modules qui discutent avec la base de données.

► <http://drupal.org/project/schema>

Enfin, il est difficile de fonctionner sans l'indispensable module Devel que nous avons déjà croisé plusieurs fois. Ce dernier dispose notamment d'un bloc spécialement conçu pour les développeurs permettant de simplifier toutes sortes d'opérations courantes.

► <http://drupal.org/project/devel>

Activation du module

Notre premier module est maintenant complet, il ne reste plus qu'à l'activer pour en observer le résultat.




Figure 14–4
Notre module en action,
notez le nouveau lien modifier

Comme vous le voyez, le lien a effectivement été ajouté à chacun des contenus, vous venez donc de réaliser avec succès votre premier module.

En résumé

- Pour qui possède quelques connaissances sur PHP, la création de modules est beaucoup trop simple pour s'en priver.
- Les modules permettent d'agir au cœur de Drupal et de l'adapter précisément à vos besoins.
- Si vous avez besoin d'aide, n'hésitez pas à poser une question sur le forum drupalfr.org, dans la section *Développement*.

chapitre 15

**Colchique**
Car chaque brin d'herbe est unique...

Contactez-nous

Chercher dans ce site :

Les Publications Les Billets FAQ

Bienvenue

Soumis par Mainine le 8 juillet 2009 - 11:11am


Bienvenue sur le site **Colchique Conseil**,
le cabinet de conseil en ressources humaines qui à l'éthique au fond du coeur.

[2 commentaires](#) [En savoir plus](#) [2 fichiers attachés](#)

Recrutement d'une équipe Exploitation pour une grande banque Allemande

Prestation Réalisée:
La mission a consisté en la constitution en un temps record d'une équipe dédiées à l'exploitation du système informatique d'une grande banque allemande.

Colchique est une société de conseil en management spécialisée dans les ressources humaines dans le domaine des technologies de l'information...



Rejoignez-nous...

Nom d'utilisateur : *

Mot de passe : *

- [Créer un nouveau compte](#)
- [Demander un nouveau mot de passe](#)

© Colchique Conseil - Tous droits réservés - Ce site est propulsé par [Drupal](#).

Mettre son site en production

C'est le grand moment. Votre site est enfin prêt et toutes les fonctionnalités sont correctement implémentées. La tentation de le mettre en ligne est grande, mais encore un peu de patience. Voyons d'abord ensemble toutes les étapes à ne pas négliger pour effectuer une mise en production parfaite.

SOMMAIRE

- ▶ Mettre à jour son installation Drupal
- ▶ Déployer son site sur le serveur de production
- ▶ Faire des réglages pour obtenir de meilleures performances
- ▶ Créer une installation multisite

Mettre à jour son installation Drupal

Pourquoi parler de cela maintenant ?

Drupal est un écosystème bien vivant qui a sans aucun doute encore évolué pendant que nous travaillions d'arrache-pied pour construire notre site. Il y a fort à parier que dans l'intervalle de temps, certains modules, et Drupal lui-même, ont changé de version, corrigeant ainsi anomalies et failles de sécurité.

C'est donc le moment idéal, maintenant que nous maîtrisons Drupal et notre site dans les moindres détails, de procéder à une rapide mise à jour nous assurant que notre mise en production sera la plus parfaite possible.

Doit-on toujours tout mettre à jour ?

Décider de ce que vous devez ou non mettre à jour doit être fait en fonction de ce que la nouvelle version de chaque élément contient concrètement.

Soit il s'agit d'une mise à jour concernant des anomalies corrigées ou des problèmes de sécurité et dans ce cas, il y a toujours urgence. S'agissant de failles de sécurité, le seul fait qu'un correctif soit publié implique que votre site est potentiellement en danger. En effet, cette situation signifie que des faiblesses ont été détectées, et que toute personne malintentionnée désireuse d'attaquer votre site en utilisant ces failles dispose d'emblée de toutes les informations nécessaires sur le Web pour mener son œuvre à bien.

D'une manière générale, les mises à jour de sécurité du cœur de Drupal sont en permanence visibles sur la page d'accueil de drupal.org. Nous vous recommandons vivement de vous abonner aux flux RSS vous prévenant en temps réel des nouvelles mises à jour pour Drupal, et pour tous les modules contribution. Pour cela, ajoutez à votre client RSS le flux officiel des annonces de sécurité de drupal.org, <http://drupal.org/security/rss.xml>.

S'il s'agit d'une mise à jour dite « majeure », c'est-à-dire impliquant un changement du premier chiffre du numéro de version (par exemple, de Drupal 6.12 vers Drupal 7.0), la mise à jour ne doit jamais être effectuée à la légère, que ce soit pour Drupal ou l'un des modules contribution. Il faut en effet être certain que la nouvelle version fonctionne correctement avec votre site, ce qui demande généralement du temps.

Enfin, évitez de mettre à jour un module stable en utilisant une version plus récente mais encore à l'état de développement ou en bêta. Ces versions sont de manière générale déconseillées pour un site en production.

Le tableau de bord des mises à jour

Un autre moyen pour se tenir au courant des mises à jour consiste à exploiter le module Update status que nous avons activé au début de notre projet. Ce module a pour rôle, souvenez-vous, d’aller régulièrement sur le site drupal.org afin de vérifier que les versions de Drupal et de tous les modules activés sont à jour. En cas de découverte d’une mise à jour importante (typiquement la correction d’une faille de sécurité), il vous prévient sur chaque page de l’administration avec le message suivant :

• Une mise à jour de sécurité est disponible pour votre version de Drupal. Afin de garantir la sécurité de votre serveur, effectuez immédiatement la mise à jour ! Voir la page des **mises à jour disponibles** pour plus d'informations.

Figure 15–1
Drupal nous avertit que de nouvelles mises à jour sont disponibles.

Si vous cliquez sur le lien *Mises à jour disponibles*, vous êtes alors redirigé sur une liste de versions pour l’ensemble des modules activés. Vous pouvez aussi à tout moment afficher cette page en allant dans *Administration > Rapports* puis *Mises à jour disponibles*.

BONNE PRATIQUE Être tenu au courant des mises à jour

Si vous cliquez sur l’onglet *Paramètres*, vous avez la possibilité d’indiquer une ou plusieurs adresses de courriel pour être prévenu automatiquement dès lors qu’un module, ou Drupal lui-même, a besoin d’une mise à jour. En effet, le module Update status ne se contente pas d’afficher simplement la page des mises à jour disponibles, mais vérifiera régulièrement par l’intermédiaire du cron leur mise à disposition.

Les différents réglages de la page vous permettent de déterminer à partir de quel niveau de criticité vous désirez être notifié.

En tête de la liste se trouvent les informations sur le cœur de Drupal, suivies par les informations sur chacun des modules actifs.


Drupal 6.9		Mise à jour de sécurité requise ! 	
Version recommandée :	6.12 (2009-mai-13)	Téléchargement	Notes de version
Mise à jour de sécurité :	6.12 (2009-mai-13)	Téléchargement	Notes de version
Mise à jour de sécurité :	6.11 (2009-avr.-30)	Téléchargement	Notes de version
Mise à jour de sécurité :	6.10 (2009-févr.-25)	Téléchargement	Notes de version
Contient : Block, Book, Color, Comment, Contact, Content translation, Filter, Help, Locale, Menu, Node, OpenID, PHP filter, Path, Ping, Poll, Profile, Search, Syslog, System, Taxonomy, Throttle, Tracker, Update status, Upload, User, amumeral, amumeral-mobile			

Figure 15–2 Détail des informations sur la mise à jour critique de Drupal

Ici, les informations sur le cœur de Drupal sont pour le moins explicites... Cette version n’est plus du tout à jour et contient des failles de sécurité à combler.

D’une manière générale, chaque ligne dispose d’un code couleur vous permettant d’un coup d’œil de savoir où vous en êtes. Lorsqu’une mise à jour critique à appliquer rapidement est disponible, le fond est rouge comme nous venons de le voir.

Figure 15–3
Détail des informations sur la mise à jour importante du module CCK

Content Construction Kit (CCK) 6.x-2.2

Mise à jour disponible

Version recommandée : 6.x-2.3 (2009-juin-05)

Téléchargement | Notes de version

Contient : Content, Content Copy, Fieldgroup, Node Reference, Number, Option Widgets, Text, User Reference

Lorsqu’il s’agit d’une mise à jour critique touchant à la sécurité du système, vous êtes en plus averti par un message supplémentaire : *Mise à jour de sécurité requise !*

Figure 15–4
Détail des informations sur la mise à jour critique du module Views

Views 6.x-2.5

Mise à jour de sécurité requise !

Mise à jour de sécurité : 6.x-2.6 (2009-juin-10)

Téléchargement | Notes de version

Egalement disponible : 6.x-3.x-dev (2009-juin-26)

Téléchargement | Notes de version

Contient : Views, Views UI

Lorsqu’il s’agit d’une mise à jour peu critique voire non critique, le fond jaune est utilisé.

Figure 15–5
Détail des informations sur une mise à jour non critique, ici pour le module Voting API

Voting API 6.x-2.0-rc2

Mise à jour disponible

Version recommandée : 6.x-2.0 (2009-juin-24)

Téléchargement | Notes de version

Contient : Voting API

Si le module Update status ne peut trouver d’informations de version, ce qui est typiquement le cas pour le module que nous avons créé nous-même, le fond est gris.

Figure 15–6
Exemple d’information pour un module non connu par drupal.org

mon_module 6.x-1.0

Aucune version disponible

Contient : Mon Module

Enfin, si aucune mise à jour n’est disponible, le fond est simplement vert.

Figure 15–7
Tout est au vert, aucune mise à jour n’est à programmer.

Comment RSS 6.x-2.0

À jour

Contient : Comment RSS

En cliquant sur le lien *Notes de révision*, par exemple pour le module Views concerné par une mise à jour critique, nous obtenons plus d'informations sur ce qui a été corrigé.

views 6.x-2.6

View

Revisions

merlinofchaos · June 10, 2009 - 22:09

6.x · Security update · Bug fixes · New features

Official release from CVS tag: DRUPAL-6--2-6

First released: June 10, 2009 - 22:09

Download: [views-6.x-2.6.tar.gz](#)

Size: 1.46 MB

md5_file hash: a39827a5430373ab8508f2773b0bc6ea

Last updated: June 10, 2009 - 22:11

[View usage statistics for this release](#)

Bugs fixed

- [#419226](#) by kelvincool: Respect admin theme during views ajax operations.
- [#419270](#) by voxPELLi: set_where_group() could fail if no previous groups.
- [#422054](#) by ybahn: Protect exports from improperly indenting multiline strings.
- [#437070](#) by mikeryan: Base table formerly limited to 32 characters.
- [#317271](#) by quicksketch: ensure_my_table() limited to one relationship per table.
- [#441520](#) by jonathan1055: Allow the "time ago" date format to work on future dates as well.

Figure 15-8 La liste des corrections effectuées par la mise à jour critique du module Views

Mise à jour du code source

Mise à jour de Drupal

Pour rester simple dans notre exposé, nous partirons du principe, pour toutes les mises à jour qui vont suivre, que vous n'avez jamais modifié le moindre fichier fourni en standard ni même ajouté quoi que ce soit dans les dossiers, que ce soit pour un module ou pour Drupal lui-même. Il s'agira donc purement et simplement de supprimer les anciennes versions pour décompresser à la place les nouvelles.

Pour mettre à jour le cœur de Drupal, nous commencerons par télécharger la nouvelle archive en utilisant le lien *Téléchargement* qui se trouve sur la première ligne des versions de retard qui nous sont indiquées sur la page des versions disponibles.

Dans notre cas, il s'agit de Drupal 6.12. Notez qu'il n'est pas nécessaire d'appliquer toutes les versions de retard. Pour Drupal comme pour les modules, c'est la dernière version qui importe.

L'objectif est maintenant de remplacer les sources obsolètes par la nouvelle version. La seule précaution que nous devons prendre concerne le dossier `sites` qui contient tout ce que nous avons modifié : le paramétrage, les modules ainsi que les thèmes que nous avons installés ou créés.

BON À SAVOIR Technique avancée et ultrarapide pour mettre à jour Drupal

Pour les utilisateurs avancés et rompus à la ligne de commande, pour mettre à jour, de la manière la plus sûre possible, une installation Drupal qui a été lourdement modifiée, le moyen efficace consiste à utiliser les commandes `diff` et `patch`. Pour cela, il faut télécharger sur drupal.org les deux archives correspondant l'une, à la version de votre installation et l'autre, à la version cible. Vous décompresserez ensuite ces deux archives dans un même dossier et utiliserez la commande `diff` pour créer un patch de mise à jour

```
diff -Naur drupal-6.9 drupal-6.12 > drupal-6.9-6.12.patch
```

Ensuite, il ne vous reste qu'à appliquer le patch à la racine de votre Drupal, d'abord en mode simulation :

```
patch -p1 --dry-run < ../patches/drupal-6.9-6.12.patch
```

Si vous constatez des erreurs, il s'agit sûrement d'un fichier que vous avez modifié, et vous aviez sûrement une très bonne raison pour faire une chose pareille.

Comparez le fichier problématique avec sa version d'origine dans le dossier `drupal-6.9` avec un outil comme WinMerge (Windows) ou Meld (Gnome). Corrigez ce qui pose problème ou, le cas échéant, sauvegardez votre modification à part pour la réappliquer plus tard.

Lorsque le patch passe enfin correctement, vous pouvez relancer la commande, cette fois en mode définitif.

```
patch -p1 < ../patches/drupal-6.9-6.12.patch
```

Cette approche permet de tenir son Drupal à jour malgré un cœur modifié. C'est aussi une bonne occasion de vous demander à chaque fois si telle ou telle modification qui bloque la mise à jour pourrait ou non être remplacée par un module à part entière.

Pour cela, nous décompressons la nouvelle archive au même niveau que le dossier de notre Drupal, obtenant ainsi un dossier `drupal-6.12`.

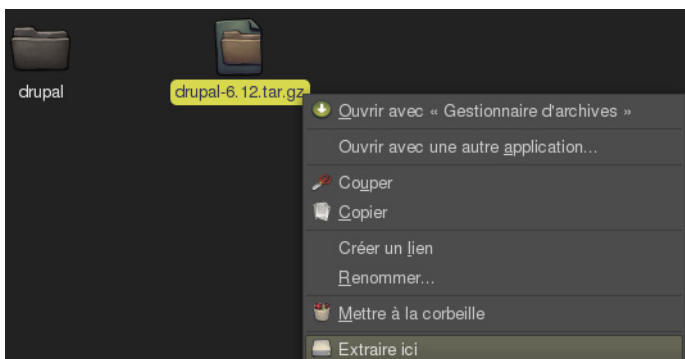


Figure 15–9
Décompression de la nouvelle
archive sous Gnome

Seul le dossier `sites` est à sauvegarder lors d'une mise à jour, aussi le déplacerons-nous du dossier `drupal` au dossier `drupal-6.12`. Ensuite, il ne reste qu'à déplacer le dossier `drupal` dans un lieu sûr à des fins d'archivage, et à renommer le dossier `drupal-6.12` en `drupal`. La mise à jour est terminée.

Mise à jour d'un module

La mise à jour d'un module est beaucoup plus simple que celle de Drupal : la suppression de l'ancien dossier du module, puis la décompression du nouveau à sa place sont les deux seules actions à effectuer.

Mise à jour d'un thème

La mise à jour d'un thème se passe strictement de la même manière que celle d'un module.

Mise à jour de la base de données

Sauvegarde de la base de données

La majeure partie de la configuration de Drupal se trouve dans la base de données. Lorsqu'un module, ou Drupal, change de version, il faut donc mettre à jour cette base pour qu'elle s'accorde avec ces nouveautés.

En premier lieu et avant toute action, nous devons faire une sauvegarde de cette base de données pour nous prémunir de tout incident pouvant survenir lors de la mise à jour. Si un problème inattendu apparaissait au cours de l'opération, vous conserverez ainsi la possibilité de revenir en arrière et de restaurer la version de départ de la base de données.

Pour sauvegarder votre base, nous allons utiliser phpMyAdmin pour effectuer son « vidage » (en anglais *dump*). Ce vidage se présentera sous la forme d'un fichier SQL qui contiendra l'ensemble des données de la base et que l'on pourra utiliser en cas de problème.

Pour ce faire, allez sur l'URL de phpMyAdmin (<http://localhost/phpmyadmin>) et sélectionnez la base de données `drupal` que nous avons créée en début d'ouvrage. Cliquez ensuite sur l'onglet *Exporter*.

Dans la zone *Exporter*, cliquez sur le lien *Tout sélectionner* de manière à ce que toutes les tables soient prises dans le vidage.



Figure 15–10
Sélection de toutes les tables pour le vidage

Figure 15-11
Définition du mode de transmission

BON À SAVOIR Faites vos sauvegardes en ligne de commande

Pour créer des vidages de manière rapide et efficace, il n'y a rien de plus efficace que la ligne de commande.

Pour créer un vidage pour une base mySQL, la syntaxe est :

```
mysqldump drupal -p qui apparaît/sauvegardes/drupal_1.0.sql
```

Et pour une base postgreSQL :

```
pg_dump -U postgres drupal > /chemin/vers/sauvegardes/drupal_1.0.sql
```

BON À SAVOIR Pouvoir mettre à jour même si vous n'êtes pas administrateur

Si vous avez besoin d'effectuer une mise à jour mais que vous n'êtes pas connecté en tant qu'administrateur, il est possible de changer dans le fichier `settings.php` la variable suivante :

```
$update_free_access = TRUE;
```

Prenez cependant grand soin de remettre cette valeur à `FALSE` dès que votre mise à jour est terminée car, dans le cas contraire, n'importe qui pourrait mettre à jour votre site.

Figure 15-12
La page de mise à jour de Drupal

Juste en dessous, vérifiez que le bouton radio `SQL` est bien sélectionné pour produire un vidage en langage SQL. En bas de la page, dans la zone `Transmettre`, activez la case à cocher à gauche du libellé, puis saisissez le nom du fichier final et cochez la compression `aucune`.

Ceci fait, cliquez sur le bouton `Exécuter` ; une boîte d'enregistrement doit alors apparaître pour vous permettre de sauvegarder le vidage sur votre disque local. Placez ce fichier dans un dossier à part, dédié aux sauvegardes.

Le script update.php

Vous pouvez à présent procéder à la mise à jour de la base de données Drupal, et pour cela, il vous faut faire appel à une page spéciale : `update.php`. Assurez-vous au préalable que vous vous êtes bien connecté en tant qu'administrateur, puis saisissez l'adresse `http://www.colchique-conseil.fr/update.php` pour afficher la page de mise à jour.

Drupal database update

Use this utility to update your database whenever a new release of Drupal or a module is installed.

For more detailed information, see the [Installation and upgrading handbook](#). If you are unsure what these terms mean you should probably contact your hosting provider.

1. **Back up your database.** This process will change your database values and in case of emergency you may need to revert to a backup.
2. **Back up your code.** Hint: when backing up module code, do not leave that backup in the 'modules' or 'sites/modules' directories as this may confuse Drupal's auto-discovery mechanism.
3. Put your site into [maintenance mode](#).
4. Install your new files in the appropriate location, as described in the handbook.

When you have performed the steps above, you may proceed.

Continue

Drupal vous indique ici qu'il est sage de faire au préalable une copie de sauvegarde de votre base de données et du code du site, ce que nous avons déjà fait. Vous pouvez donc appuyer sur le bouton `Continue`.

Sur la page suivante, en déroulant la section `Select version`, vous obtenez la liste des modules activés suivi du dernier numéro de version que Drupal a détecté pour chacun d'entre eux.

system module :
6050

content module :
6010

Figure 15-13
Derniers numéros de version des modules

Attention, ces numéros de version ne correspondent en rien aux numéros de version des modules eux-mêmes, même si généralement leurs auteurs suivent la logique qui veut que 6202 signifie « mise à jour n° 2 de la version 6.2 du module ».

Si vous déroulez la liste, vous pouvez voir que cette version est éventuellement suivie d'une série d'autres versions allant jusqu'à l'entrée *No updates available*. Pour illustrer cette liste qui n'est pas très claire, prenons un exemple.

Si, pour un module, nous avons les versions 6202, 6203 et 6204, cela signifie que lorsque Drupal fera sa mise à jour, il commencera par appliquer la mise à jour n° 6203 (la 6202 étant la version courante) puis poursuivra sur la 6204, avant de passer au module suivant. Si vous souhaitez ne mettre à jour qu'un seul module, vous pouvez le faire en sélectionnant manuellement *No Update Available* pour tous les autres modules.

Lorsque vous êtes prêt, appuyez sur le bouton *Update*. Une barre de progression apparaît pour indiquer l'avancement des mises à jour, module par module. Le processus s'achève par une page indiquant les opérations réalisées et les éventuelles erreurs.

Drupal database update

Updates were attempted. If you see no failures below, you may proceed happily to the [administration pages](#). Otherwise, you may need to update your database manually. All errors have been [logged](#).

☐ Main page
☐ Administration pages

The following queries were executed

system module
Update #6050

☐ No queries

content module
Update #6010

☐ No queries

Figure 15-14
Les mises à jour se sont correctement effectuées.

Notre base de données est à présent totalement à jour et nous pouvons commencer à tester pour vérifier qu'aucune régression n'est apparue sur notre site.

BON À SAVOIR

Le fonctionnement des mises à jour

Pour ceux qui craindraient que la mise à jour soit trop longue et dépasse le temps alloué à PHP pour l'exécution d'un script, sachez que cette procédure ne se fait pas en une seule passe. En réalité, la barre de progression est rafraîchie par des appels successifs à la procédure de mise à jour, chaque appel étant calibré pour ne jamais dépasser 30 secondes.

MODULES MAGIQUES **Drupal entièrement en ligne de commande**

Les amoureux de la ligne de commande en rêvaient, le module Drush l'a fait. Avec cet outil, vous pouvez contrôler entièrement Drupal dans une console. Quelques exemples ?

Activer un module :

```
./drush enabled mon_module
```

Mettre à jour la base de données :

```
./drush updatedb
```

Obtenir l'état de tous les modules :

```
./drush statusmodules
```

Vider les caches :

```
./drush cache clear
```

Fantastique, non ?

► <http://drupal.org/projets/drush>

Si tout va mal...

La procédure de mise à jour de la base de données est un moment assez critique dans la vie de votre site. Normalement, pour les modules les plus utilisés, cette étape se passe très bien.

Toutefois, si des erreurs graves apparaissent à la fin de la procédure de mise à jour, lisez chacune d'elles avec soin et vérifiez leur cause dans la base de données. Si vous ne comprenez pas ce qui a pu se passer, stoppez tout et allez poser la question sur le forum de drupalfr.org, dans la section [Support général](#). Prenez soin de donner la version du Drupal que vous utilisez, et de recopier les erreurs qu'il vous a renvoyées.

Une fois la cause identifiée et corrigée, vous devez relancer la procédure de mise à jour, et donc restaurer préalablement la sauvegarde des données que vous avez effectuée.

Pour cela, utilisez phpMyAdmin et cliquez cette fois sur l'onglet [Supprimer](#). Un message d'avertissement apparaît afin de confirmer l'opération.



Figure 15-15
Confirmation de la suppression
de la base de données Drupal

Confirmez votre choix. Une fois la base détruite, nous allons la recréer de la même manière que lors de l'installation.



Figure 15-16
Recréation de la base de données Drupal

Notre base étant créée, nous allons maintenant restaurer ses données grâce à notre sauvegarde en cliquant sur l'onglet [Importer](#).

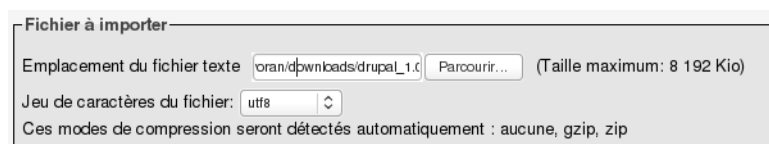


Figure 15-17
Restauration de la sauvegarde

Dans la zone [Fichier à importer](#), cliquez sur [Parcourir](#) de manière à sélectionner le fichier `drupal_1.0.sql` que nous avons créé auparavant. Ceci

fait, cliquez sur *Exécuter*. Lorsque l'importation est terminée, php-MyAdmin vous indique que tout s'est bien passé :

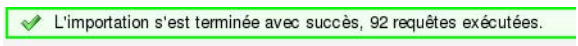


Figure 15-18
Le message de bonne fin de l'importation de phpMyAdmin

Vous pouvez maintenant relancer votre procédure de mise à jour.

Avez vous bien testé ?

Votre installation Drupal est maintenant à jour est vous êtes de plus en plus tenté de mettre votre œuvre en production. Mais il est important de vérifier auparavant que tout fonctionne parfaitement. Voyons ci-après les points à passer en revue pour avoir l'esprit tranquille.

Les tests fonctionnels

Avez-vous testé toutes les fonctions de votre site ? Pour cela, le mieux est de se construire une liste à partir de votre définition des besoins. À chaque fonctionnalité correspond une case avec une coche indiquant si le test s'est passé sans problème. Ce type de document est appelé *cahier de recette*.

Fonctionnalité	Utilisateur	Résultat	Validé ?
Ajouter de nouveaux articles	Marine	Article ajouté mais non visible en page d'accueil	OUI
Corriger un article	Philippe	Article corrigé et à l'état « corrigé »	NON
...			

Figure 15-19
Exemple d'un cahier de recette simplifié

Les tests techniques

Le premier point technique à vérifier est la compatibilité de votre site avec l'ensemble des navigateurs cibles. Lors de la définition de notre projet, nous avons identifié un besoin de compatibilité avec Mozilla Firefox 3.x et Microsoft Internet Explorer 3.x. Il est donc important de vérifier que tout fonctionne correctement sur ces deux plates-formes. Ne vous contentez pas de tester avec un seul système d'exploitation. Si votre site passe parfaitement sous Mozilla Firefox 3.5 pour Linux, rien ne dit que ce sera le cas avec le même navigateur sur Mac OS ou Microsoft Windows.

De même, pensez à la résolution d'écran qui a été définie comme minimum (ici 1280 × 1024). Bien souvent, les développeurs travaillent sur des écrans immenses et oublient de vérifier que leurs réalisations s'affichent correctement sur des écrans de moindre résolution.

BON À SAVOIR Restauration rapide de la base de données en ligne de commande

Comme pour la création de la sauvegarde, il est possible en ligne de commande de restaurer une base de manière très rapide.

Pour MySQL d'abord :

```
mysql -p
mysql> drop database drupal
mysql> create database drupal
mysql> <CTRL-C>
cat /chemin/vers/ma/sauvegarde/
drupal_1.0.sql | mysql -p
```

Puis pour PostgreSQL :

```
dropdb -U postgres drupal
createdb -U postgres drupal
cat /chemin/vers/ma/sauvegarde/
drupal_1.0.sql | psql -U postgres
```

Si vous pouvez disposer d'un serveur de test qui reproduirait à l'identique (système d'exploitation, configuration matérielle) le serveur de production, effectuez vos tests techniques sur cette machine.

Validez !

Le consortium W3C définit les normes qui régissent l'Internet. XHTML ou CSS font partie de ces normes et il convient d'être certain que votre production est parfaitement compatible.

Pour valider le code XHTML de votre site, vous avez la possibilité d'utiliser l'outil en ligne fourni par le W3C (<http://validator.w3.org/>).

Si votre site est accessible sur l'Internet, saisissez son adresse dans le premier onglet du validateur.



Figure 15-20
Validation d'une URL
sur le validateur XHTML du W3C

Dans le cas contraire, utilisez le dernier onglet et copiez-y le code source à valider.

BON À SAVOIR Microsoft Internet Explorer est-il vraiment si mauvais ?

Microsoft Internet Explorer est réputé pour sa très mauvaise prise en charge des normes CSS. C'est en partie faux. En réalité, ce sont les anciennes versions de ce navigateur qui souffraient de ce défaut ; les nouvelles – et particulièrement les versions 7 et 8 – sont quant à elles bien meilleures. De nombreuses entreprises ayant utilisé Microsoft Internet Explorer comme outil principal de navigation sur leur intranet, Microsoft ne peut donc pas casser cette compatibilité avec ses défauts de jeunesse pour d'évidentes raisons commerciales.

Il est cependant possible de créer un site parfaitement compatible avec le W3C et Microsoft Internet Explorer dans sa version 8, en insérant dans l'en-tête de vos pages la balise suivante :

```
<meta http-equiv="X-UA-Compatible" content="IE=8" />
```

De cette manière, Microsoft Internet Explorer 8 basculera automatiquement en mode « compatible », vous évitant ainsi de développer une feuille de style spécifique pour ce navigateur. Dans le cas bien sûr où votre besoin de compatibilité n'englobe pas les versions précédentes.

Vous pouvez valider vos feuilles de style en utilisant un autre outil du W3C disponible à l'adresse <http://jigsaw.w3.org/css-validator/> et qui fonctionne exactement de la même manière.

Enfin, vous avez aussi la possibilité de valider vos flux RSS en allant à l'adresse <http://validator.w3.org/feed/>.

De l'intégration à la production

Choisir son hébergement

Nous avons travaillé jusqu'à présent en local sur une machine de développement. Pour que notre site soit rendu public, il est nécessaire de transférer son code et sa base de données sur une machine visible depuis l'Internet. C'est ce que l'on appelle communément un *hébergement*.

Pour de gros sites, cet hébergement est un service payant effectué par des *hébergeurs*. Il convient de choisir une telle société en étudiant avec soin les points suivants.

- **Le type d'hébergement** : il peut s'agir d'une machine dédiée, mutualisée ou virtuelle. La machine dédiée est une machine physique qui ne travaille que pour vous. La machine mutualisée est aussi une machine physique, mais qui héberge d'autres sites en plus du vôtre. Enfin, la machine virtuelle peut être vue comme une ou plusieurs « tranches » de machine physique, une tranche vous étant totalement dédiée. Le choix d'une formule dépend principalement du trafic prévisionnel que vous avez estimé, et bien évidemment de votre budget.
- **Le débit montant et descendant de l'accès à l'Internet** : plus le trafic attendu est important, plus ce débit doit être élevé.
- **La capacité de stockage du serveur** : cet aspect est important si vous comptez héberger des contenus lourds (par exemple, des vidéos).
- **Les applications disponibles** : dans un environnement mutualisé, ou sur demande pour un dédié ou un virtuel, vous devez disposer de tout l'environnement logiciel pour faire tourner votre site Drupal : Apache, PHP, MySQL/PostgreSQL, phpMyAdmin et un *panel*, qui est un outil simplifié d'administration du serveur.
- **Le contrat de service** : aussi appelé SLA (*Service Level Agreement*), il précise le délai maximal d'attente avant intervention lors d'une panne, et le délai maximal d'attente avant remise en service d'un serveur. Plus ces temps sont courts, plus le service est cher. Ce contrat de service est donc à moduler en fonction de vos besoins. Notez qu'il peut aussi inclure la prise en charge de prestations liées aux logiciels installés sur les serveurs (paramétrage, mise à jour, etc.).

BON À SAVOIR

Un hébergement libre pour le libre

Un hébergement sans publicité, de qualité et gratuit, c'est possible grâce au *libre*.

L'association à but non lucratif TuxFamily met en effet à la disposition de tout projet en rapport avec la philosophie du libre un hébergement gratuit, de niveau professionnel et de type « mutualisé ». Cet hébergement comprend tout ce qui est nécessaire à la mise en production d'un site sous Drupal, y compris une base de données PostgreSQL (base de données concurrente de MySQL, mais plus rare à trouver chez un hébergeur).

Notez cependant que même si ce service est gratuit, il repose sur l'entraide et le partage. Aussi, toute donation de votre part en temps (pour administrer les serveurs ou aider les autres utilisateurs) et en argent (à la mesure de vos moyens), permettra à ce projet de continuer à démocratiser l'accès à l'Internet.

► <http://www.tuxfamily.org>

Si votre projet est un site personnel, vous pouvez aussi vous tourner vers des hébergements gratuits. La majorité des fournisseurs d'accès à l'Internet (FAI) mettent à la disposition de leurs abonnés des espaces personnels compatibles avec Drupal. Eux aussi disposent d'une interface d'administration simplifiée (panel).

Acheter son nom de domaine

Pour disposer d'un site visible sur l'Internet, vous devez disposer d'un nom de domaine. Dans notre cas, il s'agira de colchique-conseil.fr.

Un nom de domaine ne s'achète pas, il se loue pour une période de temps donnée. La société vers laquelle vous vous tournerez pour effectuer cette location est un bureau d'enregistrement des noms de domaine. Il existe plusieurs sociétés de ce genre, qu'il convient d'étudier pour déterminer le meilleur rapport service/prix.

Une fois la période de location écoulée, le bureau d'enregistrement qui vous a loué ce nom vous demandera si vous souhaitez renouveler cette location. Si vous n'y pensez pas (ou que la société ne vous prévient pas), quelqu'un d'autre pourra alors acheter ce nom de domaine à votre place. Soyez donc vigilant et pensez à planifier vos renouvellements.

ATTENTION Restez maître de votre nom de domaine

Ne laissez jamais une société tierce acheter un nom de domaine à votre place. Il faut impérativement que ce nom de domaine soit à votre nom ou à celui de votre société.

Par le passé, il y a déjà eu des cas où certaines sociétés peu scrupuleuses incluaient dans leur offre de réalisation d'un site l'achat du domaine et le service d'hébergement. Lorsqu'un de ces sites connut le succès, la société refusa que l'hébergement passât chez un autre prestataire et prit le nom de domaine en otage car elle l'avait acheté à son nom.

De manière générale, diversifiez vos fournisseurs en en prenant un pour le nom de domaine, un autre pour l'hébergement et un éventuel troisième pour la réalisation.

Une fois le nom de domaine acheté, le bureau d'enregistrement a pour tâche de prévenir la planète de son existence. C'est le rôle des serveurs DNS (*Domain Name Server*). Les serveurs DNS à travers le monde discutent en permanence entre eux pour s'échanger les noms de domaine créés ou modifiés.

Mais le rôle principal d'un serveur DNS est de permettre à l'internaute, lorsqu'il saisira l'adresse www.colchique-conseil.fr dans son navigateur, d'aboutir sur le serveur de votre hébergeur. Il convient donc de paramétrer votre nom de domaine acheté avec l'adresse IP du serveur de votre hébergeur. Tous les bureaux d'enregistrement de noms de domaine disposent à cet effet d'une interface spécifique permettant de créer cette association.

Le premier déploiement

Le grand moment est enfin arrivé : votre site est terminé, votre Drupal est à jour et vous avez testé tout ce qu'il vous était possible de tester. Vous allez enfin pouvoir déployer votre œuvre sur votre serveur de production.

Dans la plupart des cas, l'hébergeur que vous avez choisi vous propose – en plus des indispensables Apache, PHP et MySQL (ou PostgreSQL) – une interface d'administration simplifiée vous permettant de déployer votre site sans posséder de connaissances particulières (c'est le panel évoqué précédemment).

Il existe de nombreux panels et il y a de fortes chances que votre hébergeur ait développé un exemplaire qui lui est totalement spécifique. Il est donc impossible de donner une procédure applicable pour l'ensemble des panels existants. Nous en présenterons un ci-après qui a l'avantage d'être comme Drupal, un projet libre : AlternC.

Nous avons déjà paramétré le nom de domaine pour qu'il pointe sur l'adresse de notre hébergeur. Il nous faut à présent déclarer ce même nom de domaine en tant que service web, grâce au panel de notre hébergeur.

URL AlternC

► <http://www.alternnc.org/>

Figure 15-21

Déclaration du nom de domaine chez l'hébergeur

Dans la mesure où le serveur DNS est géré par notre bureau des enregistrements de noms de domaine, nous allons prendre soin de décocher *Héberger le DNS sur le serveur*. Ceci fait, cliquez sur *Héberger ce nouveau domaine*.

Figure 15-22

Le domaine est maintenant créé.

Copier les sources du site

Le plus simple pour envoyer notre code source sur le serveur est d'utiliser un compte FTP.

Figure 15-23
Création d'un compte FTP sur AlternC

Ici, nous utilisons l'identifiant `admin` pour notre compte et un mot de passe difficile à casser (long, contenant des lettres et des chiffres et pas de mots identifiables dans un dictionnaire). La création effective du compte est réalisée en appuyant sur le bouton *Créer le compte FTP*.

Figure 15-24
Le compte FTP est maintenant créé.

Figure 15-25
Détail de l'arborescence créée par AlternC via
Gnome Nautilus

```

▼ ftp en tant que admin sur demo.alternc.org
  ▼ www
    ▼ colchique-conseil.fr
  
```

Vous constatez alors que le panel nous a automatiquement créé un dossier `www/colchique-conseil.fr`. Ce sera la racine de notre site Drupal.

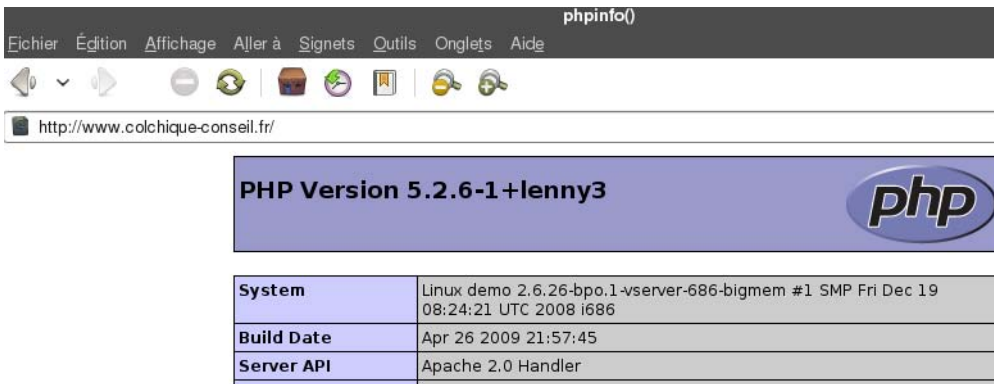
Mais pour vérifier que tout fonctionne, nous allons au préalable créer dans le dossier `colchique-conseil.fr` un nouveau fichier `index.php` dans lequel nous placerons le code suivant :

Code PHP de test du serveur

```
<?php phpinfo() ?>
```

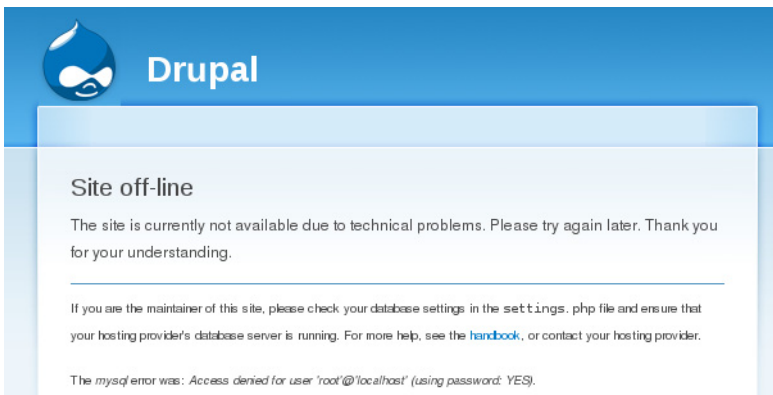
Une fois le fichier sauvegardé dans le dossier FTP `www/colchique-conseil.fr`, nous pouvons vérifier que tout fonctionne correctement à l'aide d'un navigateur :

Tout fonctionne correctement, ce qui nous permet de passer à la dernière étape : recopier le contenu de notre dossier `drupal` dans le dossier `www/colchique-conseil.fr`. Auparavant, pensez à détruire le script de test `index.php`.

**Figure 15–26**

Cet écran nous indique que le serveur fonctionne et que PHP y est opérationnel.

Cette opération réalisée (ce qui peut prendre un certain temps !), nous pouvons vérifier que Drupal fonctionne en rafraîchissant la page de notre navigateur :

**Figure 15–27**

Notre site est en ligne, mais encore « off-line » pour cause d'absence de base de données.

Cela est de bonne augure, puisque effectivement notre site est hors-ligne en l'absence d'une base de données.

Mettre en place la base de données

Cette étape est très proche de ce que nous avons appris à faire un peu plus haut pour une sauvegarde/restauration de base de données. La première étape consiste donc à effectuer une sauvegarde de nos données. Cependant, n'oubliez pas que votre installation contient pour l'instant beaucoup de données de tests. Veillez à faire un grand ménage avant de créer la sauvegarde ; il serait dommage que ces données apparaissent en ligne et vous oblige au dernier moment à procéder au nettoyage directement sur le serveur de production.

Une fois la sauvegarde achevée, nous pouvons créer sur le serveur la base de données et, par la même occasion, définir le mot de passe permettant d'y accéder. Vous savez déjà le faire avec phpMyAdmin, mais cela nécessite que votre hébergeur vous ait donné les droits de création sur `mySQL`. Voyons donc comment la même opération est réalisable avec AlternC en allant dans la section *MySQL*, et en cliquant sur le lien *Utilisateurs MySQL*.

Figure 15-28
Création de la base de données MySQL

Une fois que vous avez choisi un mot de passe solide, cliquez sur le bouton *Créer ma base de données principale*. Attention, le nom de la base de données nous est imposé par AlternC, ici c'est `admin`.

Une fois la base créée, nous utiliserons notre phpMyAdmin local pour créer une sauvegarde de notre base de données. Dans la section *Transmettre*, cochez cette fois le bouton radio *gzippé*. Nous obtiendrons ainsi une sauvegarde prenant beaucoup moins de place, et donc plus rapide à transférer sur le serveur de l'hébergeur.

À l'aide de votre connexion FTP, transférez la sauvegarde à la racine. Attention, ne placez jamais vos sauvegardes dans le dossier `www/colchique-conseil.fr` car elles seraient automatiquement téléchargeables par des tiers, rendant ainsi visibles les informations confidentielles qu'elles peuvent contenir (comptes utilisateur, mots de passe, etc.).

Une fois la sauvegarde téléchargée à la racine, nous retournons sur le panel AlternC pour cliquer, toujours dans la section *MySQL*, sur le lien *Bases de données*. Apparaît alors une liste contenant notre seule base de données `admin` et deux liens : *Sauvegarde* et *Restauration*. Cliquons sur *Restauration*.

Figure 15-29
Restauration de nos données locales
sur le `mySQL` distant

Saisissez le nom du fichier de sauvegarde que nous venons de transférer par FTP et cliquez sur *Restaurer la base de données*. AlternC nous confirme alors que les données ont été correctement réintégrées.

Pour que Drupal puisse reconnaître la base de données, il nous faut modifier son fichier de paramétrage `defaults/settings.php`. Comme précédemment avec le fichier `index.php`, votre outil FTP vous permet d'éditer directement ce fichier. Nous allons changer la référence à la base de données correspondant au MySQL local pour indiquer à présent celle correspondant au MySQL de l'hébergeur, en prenant soin de saisir ici l'identifiant (`admin`) et le mot de passe définis plus haut.

```
$db_url = 'mysql://admin:mot_de_passe@localhost/drupal';
```

Une fois le fichier sauvegardé, rafraîchissons la page de notre navigateur pour juger du résultat.

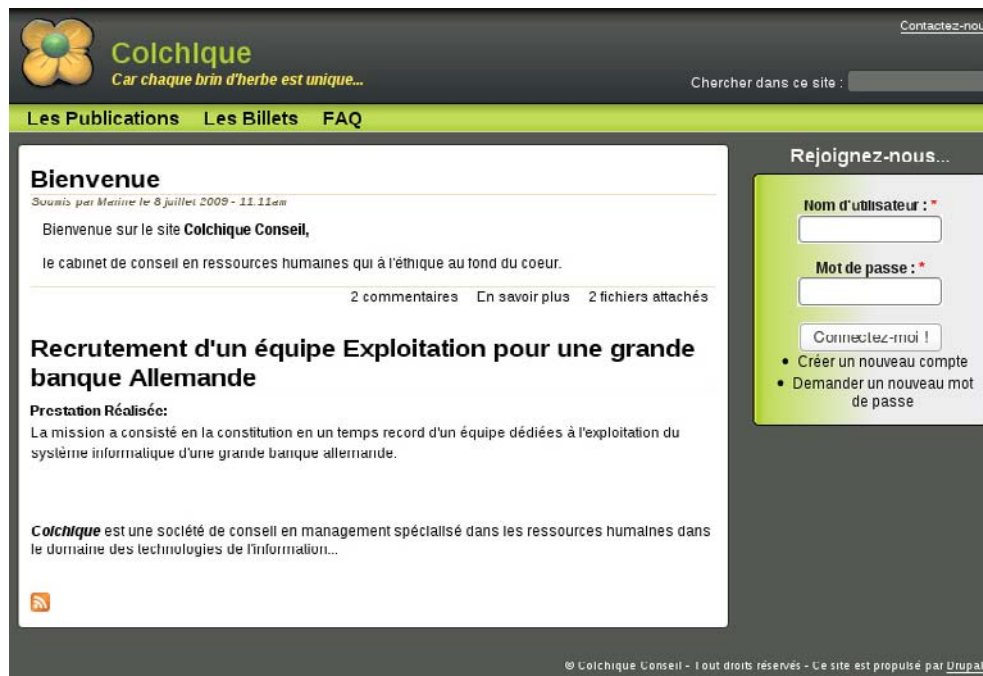


Figure 15-30
Notre site, en ligne, pour de vrai !

Vous pouvez faire le tour du propriétaire et vérifier ainsi que tout est en place. Vous détenez maintenant les clés d'un site public visible par tous.

Les déploiements suivants

Le temps a passé, et vous avez certainement eu de nouvelles idées pour améliorer et étoffer votre site. Comme nous avons pris la bonne habitude de ne jamais travailler sur le serveur de production, toutes ces modifications ont été faites sur votre machine de développement. Il vient un moment où ces nouvelles fonctionnalités sont prêtes à être déployées sur le serveur de production pour devenir accessibles à votre public.

MODULES MAGIQUES Déployer vos réglages en toute simplicité

Il existe un module très prometteur appelé Deploy dont l'objectif est de combler cette lacune de déploiement automatisé. Il est pour l'instant à l'état de projet expérimental mais fonctionne déjà très bien avec une base de données MySQL. Grâce à lui, il sera possible de déployer automatiquement les mises à jour sur le serveur distant après les avoir testées en local, sans avoir à refaire toute l'administration à la main.

► <http://drupal.org/project/deploy>

Pour ce qui est du code (modification sur le thème, nouveaux modules, etc.), la manipulation est très simple : il suffit de recopier les modifications par FTP.

Pour la base de données, les choses se corsent un peu. Généralement, vos modifications s'accompagnent d'un certain nombre d'actions d'administration, par exemple activer un module, paramétrer un type de contenu, etc. Tout ceci est, comme nous l'avons vu, stocké dans la base de données. Or, cette base de données dans l'environnement de développement ne peut plus être intégrée telle quelle sur le serveur de production, car vous écraseriez toutes les informations qui ont été ajoutées entre temps (nouveaux contenus, nouveaux commentaires, etc.) par vos utilisateurs et visiteurs.

À l'heure actuelle, il n'existe malheureusement pas de solution fiable à ce problème, qui est somme toute le seul dont souffre Drupal, comme beaucoup d'autres CMS. Les modules Views et CCK disposent de fonctions d'exportation qu'il est possible d'utiliser pour importer les nouvelles vues ou les nouveaux contenus sur le serveur. Mais c'est là tout ce qu'il est possible d'automatiser. Vous êtes donc contraint de refaire l'ensemble de vos paramétrages à la main sur le serveur. C'est sans aucun doute fastidieux, mais ne cédez pas pour autant à la facilité de travailler directement sur le serveur, en y ajoutant de nouveaux modules ou thèmes. Vous prendriez ainsi le risque de commettre une erreur qui est sans conséquence sur votre machine de développement, mais qui ne le sera pas sur la machine de production.

Avant de procéder à ces modifications de configuration en ligne, et pour ne pas perturber les visiteurs, vous devez basculer votre site en mode Maintenance. Ainsi, tous les visiteurs seront informés qu'une opération de maintenance est en cours, et ne pourront aller plus loin dans leur navigation.

Pour activer le mode Maintenance de Drupal, il faut aller dans *Configuration du site*, puis *Maintenance du site*. Là, vous cochez l'option *Hors ligne*. Vous pouvez éventuellement modifier le message aux visiteurs :

Etat du site :

- ☐ En ligne
☒ Hors ligne

Si vous entrez "En ligne", tous les visiteurs pourront naviguer sur votre site normalement. Si vous choisissez "Hors ligne", seuls les utilisateurs avec le droit "administrer la configuration du site" pourront avoir accès à votre site pour réaliser la maintenance, tous les autres visiteurs verront le message du site hors ligne configuré ci-dessous. Les utilisateurs autorisés peuvent se connecter pendant le mode "Hors ligne" directement via la page de [connexion](#).

Message du site hors ligne :

Colchique est en cours de maintenance. Nous serons de retour très bientôt.
 Merci de votre patience.

Figure 15-31
 Activation du mode
 « maintenance » de Drupal

Une fois que vous avez cliqué sur *Enregistrer*, vous êtes maintenant le seul à pouvoir accéder au site car, pour le reste du monde, le message suivant s’affichera :



Figure 15-32
L’écran de maintenance pour les visiteurs

Enfin, pour vous rappeler que le site est en mode Maintenance, Drupal affiche sur chaque page de l’administration le message suivant :

Fonctionne en mode hors-ligne.

Figure 15-33
Le message de rappel du mode Maintenance activé pour l’administrateur

Vous pouvez maintenant opérer les modifications, ou encore mettre à jour la base de données. Lorsque le site est opérationnel, il vous suffit de retourner dans *Configuration du Site > Maintenance du site* pour rebasculer en mode En ligne.

Performances

Vision générale des performances

Les bonnes pratiques des métiers de l’informatique recommandent de prévoir les performances d’un système informatique en amont, dès la conception, et de faire des ajustements tout au long du développement. Lorsque le projet est terminé, il est souvent trop tard – dans le sens de « plus assez de temps » mais aussi de « plus rien à faire » – pour se préoccuper de ces aspects. Il est illusoire de croire que des optimisations de dernière minute pourront récupérer un site mal pensé.

L’informatique est l’art de manipuler des données. De manière très schématique, les données doivent être lues quelque part, puis traitées et envoyées autre part. La durée de tels cycles est donc logiquement conditionnée par la nature des intervenants « physiques ». Ainsi, écrire sur un

BON À SAVOIR Revenir du mode maintenance grâce à SQL

Attention à ne pas fermer votre navigateur lorsque votre site est hors ligne sous peine d’être vous aussi dans l’impossibilité d’y accéder à nouveau ! Si d’aventure vous vous retrouviez dans une telle situation, pas de panique. Connectez-vous à la base de données via phpMyAdmin et lancez les requêtes suivantes :

```
update      variable      set
value='s:1:"0";'          where
name='site_offline';
delete from cache
```

BON À SAVOIR Changer l’apparence de la page de maintenance

Il est possible de modifier l’apparence de cette page pour qu’elle corresponde complètement à la charte graphique de votre site. Pour cela, il suffit simplement de recopier le modèle qui se trouve dans le dossier `modules/system/maintenance-page.tpl.php` dans votre thème pour pouvoir le modifier à votre convenance.

réseau n'a pas le même impact qu'écrire en mémoire ou encore sur un disque dur, comme le résume le graphique suivant :

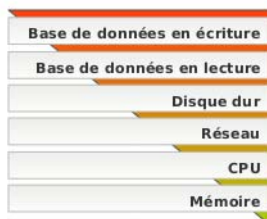


Figure 15-34
Les points critiques pour les performances,
par ordre de priorité

Il est donc important de minimiser ce qui est le plus lent (en rouge, en haut) et d'optimiser l'utilisation de ce qui est le plus rapide (en vert, en bas). Notez que le rang accordé au CPU ne tient pas compte d'une situation où un algorithme mal pensé rendrait le traitement plus lent que sur la pire des bases de données.

Certains éléments sont composites. Une base de données en local sera plus rapide, même en écriture, qu'une base de données sur un réseau très lointain, mais peut être plus lente qu'une base de données sur un réseau bénéficiant d'une baie de disques ultrarapides, de beaucoup de mémoire et de plusieurs CPU. De nombreux facteurs conditionnent la performance finale, aussi tout ceci ne donne-t-il que des ordres d'idée. La seule règle qui prévaut pour s'assurer de bonnes performances est « prévoir et vérifier ».

C'est encore plus vrai pour les applications web qui, par le nombre d'intermédiaires intervenant entre le navigateur de votre visiteur et votre code, demandent une attention toute particulière.

Les performances de Drupal

Drupal est lui-même un système complexe dont l'architecture modulaire vous permet d'étendre à l'infini ses possibilités de base. Mais cette extraordinaire souplesse peut aussi avoir des impacts catastrophiques sur les performances. L'exemple typique est donné par les requêtes SQL que génère Drupal pour fabriquer une page.

Pour en juger par vous-même, utilisez le module Devel que nous avons installé lors de notre travail sur les thèmes, et activez le sous-module Performance Logging. Cela nous permettra d'obtenir un rapport sur les requêtes SQL générées, mais aussi sur l'occupation mémoire et le temps de création de la page.

Pour paramétrer ces fonctionnalités, allez dans [Configuration du site](#) > [Devel Settings](#). Dans la section [Query Log](#), cochez [Collect query info](#).

Devel settings

Query log

☒ Collect query info

Collect query info. If disabled, no query log functionality

☐ Afficher la liste des requêtes.

Display a log of the database queries needed to generate the view. The queries are summed in the # column, and printed in red in the view.

Sort query log :

☒ by source☐ by duration**Figure 15–35**

Le paramétrage des requêtes du module Devel

Dans la section suivante, cochez *Afficher le chronomètre de page* et *Display memory usage*. Vous pouvez désormais enregistrer et retourner sur la page d'accueil.

En bas de la page, vous devriez voir apparaître une ligne de statistiques générales.

```
Page execution time was 866.67 ms. Executed 154 queries in 273.23 milliseconds.
```

Memory usage:

```
Memory used at: devel_init()=0.32 MB, devel_shutdown()=6.31 MB.
```

Figure 15–36

Les performances de notre système

Nous voyons ici que le temps de fabrication de la page est de 866 microsecondes et que cette fabrication a nécessité l'exécution de 154 requêtes qui ont occupé la machine pendant un peu plus d'un tiers de ce temps de création (273 microsecondes). Les temps peuvent vous sembler importants, mais n'y prêtez pas trop attention, car nous ne sommes pas sur un serveur mais une machine de développement. Les chiffres qui comptent vraiment sont le nombre de requêtes et le rapport entre le temps pris par celles-ci et le temps total passé (dans notre cas, un tiers du temps total).

Allons maintenant dans le panneau de configuration des modules, et désactivons *workflow*, *node access*, *five star* et *votingapi*. Revenons ensuite sur la page d'accueil et observons les chiffres :

```
Page execution time was 961.83 ms. Executed 153 queries in 238.49 milliseconds.
```

Memory usage:

```
Memory used at: devel_init()=0.31 MB, devel_shutdown()=6.07 MB.
```

Figure 15–37

Les performances de notre système avec quelques modules en moins

Le nombre de requêtes n'a pas beaucoup changé (juste une de moins) ; en revanche, le temps pris par les requêtes a nettement baissé (un quart au lieu d'un tiers).

L'explication tient justement de la nature modulaire de Drupal. En effet, Workflow et Node access n'ajoutent pas beaucoup de requêtes, mais modifient les requêtes faites par d'autres modules, notamment celles du module Node. Par exemple, Node access ajoute une série de jointures pour restreindre éventuellement l'affichage du contenu en fonction des droits de la personne qui affiche la page.

La conséquence est que le nombre de requêtes ne change pas, mais que celles-ci deviennent de plus en plus lourdes. Et ceci est encore plus marqué avec des modules prenant en charge les réseaux sociaux comme User relationship.

En résumé, le danger des modules réside avant tout dans leur simplicité d'installation, et donc dans leur multiplication, alors que le besoin n'est pas toujours réel. Combien de sites ont activé des modules comme Node access ou Workflow uniquement pour les tester, mais ne les ont jamais désactivé par la suite ? Les bonnes performances de Drupal sont donc assurées en partie par une bonne hygiène dans votre gestion des modules : n'activez que ce dont vous avez besoin.

Pensez également à vérifier grâce à Devel le nombre de requêtes générées et la proportion temps SQL/temps de création. Dès que vous détectez un ratio excessivement élevé, retournez dans la configuration *Devel Settings* et activez l'option *Afficher la liste des requêtes* et *Sort query log by duration*. Vous obtiendrez ainsi sur chaque page la liste des requêtes générées, avec en tête celles qui sont les plus longues.

Figure 15-38
Rapport des requêtes produit
par le module Devel

Executed 163 queries in 243.36 milliseconds. Queries taking longer than 5 ms and queries executed more than once, are highlighted . Page execution time was 880.72 ms.			
ms	#	where	query
21.3	1	cache_set	UPDATE cache_filter SET data = '<p>Les administrateurs réseaux connaissent t-ils la crise ?</p>W012', created = 1246028603, expire = 1246115003, headers = '', serialized = 0 WHERE cid = '1:2cd477728ae3bb46973e74a939d25d99'
9.66	1	module_list	SELECT name, filename, throttle FROM system WHERE type = 'module' AND status = 1 AND bootstrap = 1 ORDER BY weight ASC, filename ASC

Améliorer les performances du serveur

Gestion des caches

Ceci étant dit, si vous désirez construire un site avec flux de travaux, droits d'accès par utilisateur, réseaux sociaux et blogs, il est bien évident que les requêtes seront nombreuses et denses, sans que vous puissiez faire grand chose pour les réduire. Notez seulement que vous avez aussi la possibilité

de développer ou de faire développer des modules taillés au plus près de vos besoins, et donc plus optimisés pour un site complexe à fort trafic.

Drupal dispose néanmoins de plusieurs parades efficaces lui permettant de rester très performant malgré le nombre de requêtes imposées par les modules. La première de ces techniques est son système de cache, qui est un élément crucial pour améliorer les performances.

Son principe est très simple : lorsque Drupal ou un module doit faire une opération qui risque d'être coûteuse en temps à cause d'un nombre important de requêtes SQL, il vérifie d'abord si ce travail n'a pas déjà été réalisé et placé alors en cache. Si c'est le cas, il saute l'étape en question pour utiliser directement la valeur présente dans le cache. Si ce n'est pas le cas, il effectue le calcul et y stocke le résultat.

De nombreux caches sont ainsi utilisés par Drupal :

- le cache de page, qui permet de renvoyer quasi instantanément une page complète qui a déjà été générée ;
- le cache des formats d'entrée, dans lequel les contenus formatés par les formats d'entrée sont stockés ;
- le cache des blocs, qui permet de ne pas recalculer systématiquement chaque bloc d'une page ;
- le cache des menus, utilisé par Drupal pour ne pas avoir à systématiquement passer en revue les menus déclarés par les modules ;
- le cache des contenus, utilisé par CCK ;
- le cache des vues, utilisé par Views.

La majeure partie de ces caches fonctionne sans que vous n'ayez rien à faire. Le cache de page constitue la seule exception, mais il est aussi le plus efficace de tous. Il permet en effet de stocker l'ensemble d'une page de manière à ce qu'elle engendre un temps de création à peu près nul les fois suivantes. C'est bien simple : lorsque ce cache de page est activé, il faut moins de 10 requêtes SQL pour générer une page, et ceci avec un nombre très limité de modules chargés en mémoire.

Ce système de cache de page permet d'obtenir des performances équivalentes à un site en XHTML statique, avec une seule contrainte, cependant : il ne fonctionne que pour les visiteurs anonymes.

En effet, il peut y avoir beaucoup trop de changements entre une page d'un utilisateur identifié et celle d'un autre (son menu de navigation, ses blocs, son thème, etc.) pour qu'on puisse utiliser le cache de page sans risque. Ceci étant dit, dans la majorité des cas, vos visiteurs seront anonymes et le cache de page libérera ainsi autant de temps machine pour ceux qui sont identifiés.

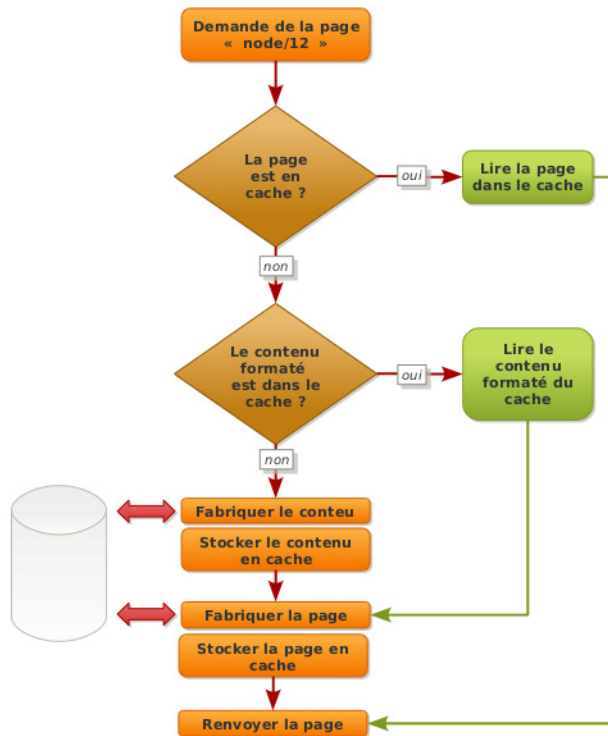


Figure 15–39
Fonctionnement du cache de page

Paramétrage du cache

Par défaut, le cache de page est désactivé et, pour changer cela, il faut aller dans *Configuration du site > Performances*. C’est d’ailleurs à cet endroit que se règle la majorité des aspects liés aux performances que nous allons aborder.

La première étape est donc d’activer ce cache de page en choisissant l’un de ses trois modes de fonctionnement :

Mode de cache :

- ☐ Désactivé
- ☒ Normal (recommandé en production, pas d’effets secondaires)
- ☐ Agressif (experts seulement, effets secondaires possibles)

Figure 15–40
Les trois modes de fonctionnement
du cache de page

La différence entre les modes *Normal* et *Agressif* vient de la manière dont sont pris en charge les modules lorsqu’une page est déjà en cache. En mode Normal, un certain nombre de modules qui en ont fait la demande sont démarrés. En mode Agressif, plus un seul module n’est utilisé.

Les modules que Drupal indique comme étant incompatibles avec le mode Agressif ont généralement besoin de commencer à travailler très tôt, par exemple pour changer le thème en fonction de l'adresse ou du type de navigateur (ex : Mobile tools pour faire un Drupal optimisé pour les mobiles). Ceci étant dit, il ne s'agit que d'un *risque* d'incompatibilité ; le mieux est donc de tester et de vérifier préalablement.

Juste après la sélection du mode de cache arrive la durée de vie du cache. De manière automatique, le cache Page est vidé à chaque modification de contenu ou lors de l'ajout d'un commentaire. Il serait donc logique d'indiquer ici la plus haute des valeurs, à savoir **1 jour**, mais ce serait oublier que certains blocs, par exemple la liste des utilisateurs en ligne, ne provoquent pas de leur propre chef – et fort heureusement – de vidange du cache Page. Finalement, cette valeur doit être réglée au temps qui vous semble raisonnable pour que de tels blocs soient à jour, par exemple, 30 minutes est une valeur raisonnable pour un site à moyen trafic.

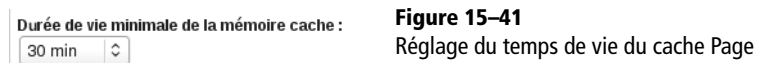


Figure 15-41
Réglage du temps de vie du cache Page

Pour conclure sur le sujet des caches, vous avez aussi la possibilité d'activer ici le **Cache des blocs**. C'est en général une bonne idée, car cela améliorera sensiblement les performances, surtout si vous utilisez beaucoup de blocs et que ces derniers induisent de nombreuses requêtes.

Cache des blocs :

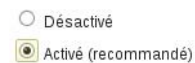


Figure 15-42
Activation du cache des blocs

Accélérateurs de code PHP

Une autre manière d'améliorer très sensiblement les performances de Drupal, ou d'une application PHP en général, est d'utiliser un accélérateur PHP. Le principe est très proche de celui du cache. Au lieu d'interpréter à chaque page le même code PHP, l'accélérateur crée la première fois une version plus compacte et plus rapide qu'il stockera en mémoire. Les fois suivantes, le code « optimisé » est directement exécuté en mémoire et ne demande ainsi plus d'accès au disque. Les gains de performance sont très importants, surtout pour Drupal, avec ses milliers de lignes de code PHP.

Ce type d'outil s'installe en général comme un module pour le service Apache et ne demande absolument aucune administration dans Drupal. Il existe plusieurs outils de cette sorte comme eAccelerator (<http://eaccelerator.net/>), XCache (<http://xcache.lighttpd.net/>) ou encore APC

BON À SAVOIR

Booster les performances d'APC

APC vérifie systématiquement si les fichiers PHP ont été modifiés pour éventuellement les recompiler et mettre ainsi à jour son cache. Cette approche permet de prendre en compte toute modification du code PHP au détriment des performances, car cela implique un accès disque pour chaque fichier, et comme Drupal en compte des milliers...

La solution à appliquer en serveur de production est de modifier le paramétrage d'APC et de désactiver cette fonctionnalité (paramètre `apc.stat = 0`). Vous gagnerez ainsi en performance, mais en contrepartie vous devrez utiliser l'interface d'administration d'APC pour mettre à jour vous-même le cache lorsque vous modifiez le code de Drupal sur le serveur.

URL Module Cache Router

► <http://drupal.org/project/cacherouter>

BON À SAVOIR

Utilisez la mémoire de vos serveurs

Comme vous l'aurez remarqué sur la page de projet du module Cache Router, ce dernier permet aussi d'utiliser les caches File et Memcache. Le premier mode consiste simplement à stocker le cache en tant que fichier. C'est moins rapide que la mémoire mais plus rapide que la base de données, à utiliser si votre serveur n'a pas assez de mémoire disponible. L'autre mode permet d'utiliser un serveur de mémoire appelé Memcached (<http://www.danga.com/memcached/>). Ce serveur permet d'utiliser la mémoire d'une machine qui se trouve sur le réseau à partir d'une ou plusieurs autres machines. C'est un outil de prédilection pour les sites à très haut trafic hébergés sur des grappes de serveurs.

(<http://us.php.net/apc>). Il est communément admis qu'APC est un système qui a prouvé sa robustesse et ses performances.

Le cache de Drupal en mémoire

L'autre avantage des accélérateurs de code PHP est qu'ils fournissent aux applications écrites en PHP la possibilité de stocker des données arbitraires dans une mémoire partagée, ce qui peut là aussi énormément booster les performances de notre Drupal.

En effet, nous avons maintenant bien compris comment le cache de Drupal fonctionne, mais nous n'avons pas encore pris connaissance de l'endroit où sont stockées ces données. La réponse – qui peut sembler un peu paradoxale – est : en base de données. Il devient par conséquent évident que notre site serait beaucoup plus rapide si ces données étaient stockées en mémoire (en utilisant par exemple APC), et nous aurions tort de nous en priver tant ceci est facile à mettre en place.

Il nous suffit pour cela d'installer le module Cache Router. Ce module permet de remplacer le stockage par défaut du cache en base de données par un stockage en mémoire via APC, eAccelerator ou XCache.

Une fois le module installé (il n'est pas nécessaire de l'activer), éditez le fichier de configuration `settings.php` et ajoutez-y les lignes suivantes :

Activation de Cache Router dans settings.php

```
$conf['cache_inc'] = './sites/all/modules/contrib/cacherouter/cacherouter.inc';
$conf['cacherouter'] = array(
  'default' => array(
    'engine' => 'apc',
    'server' => array(),
    'shared' => TRUE,
    'prefix' => 'colchique',
  ),
);
```

La première ligne permet à Drupal d'utiliser le module Cache Router pour sa gestion du cache. Veillez à bien vérifier que le chemin est le bon, sinon votre site ne fonctionnera plus ! Le bloc suivant indique que les données du cache de Drupal doivent utiliser APC (`'engine'=>'apc'`).

Il suffit de sauvegarder le fichier (et de s'assurer bien évidemment qu'APC est installé et activé) pour que le nouveau stockage du cache soit automatiquement pris en compte. Dans la mesure où beaucoup de données seront ainsi stockées en mémoire, vous pouvez aussi augmenter l'espace de RAM allouée à APC. Pour cela, vous devez modifier le paramètre `apc.shm_size`, par exemple `apc.shm_size=256` pour allouer 256 Mo au cache APC, et donc à Drupal.

Accélérer le journal

Comme nous l'avons mentionné au tout début de cet ouvrage, Drupal écrit dans un journal (appelé *Watchdog*) de très nombreuses informations sur son fonctionnement. Ces informations sont disponibles en allant dans *Administrer* > *Rapports* > *Tableau de bord*.

Le problème de performance induit par ce système réside dans le fait que ces informations sont écrites en base de données, ce qui prend obligatoirement du temps. Ce stockage est pris en charge par le module standard de Drupal Database logging.

Sur les systèmes Unix, il est conseillé d'utiliser le module Syslog fourni en standard, et de désactiver le module Database logging. Ainsi, votre système n'écrit plus rien en base de données, mais utilisera le démon Unix Syslog local à la machine, et qui est conçu dès l'origine pour la performance. L'inconvénient est que les erreurs ne seront plus accessibles par l'interface, mais seulement par visualisation des journaux du système Unix.

Améliorer les performances du navigateur

Compression des pages

Une fois les performances du serveur optimisées, il est possible de passer encore un peu de temps à améliorer celles du navigateur. Cela correspond avant tout à réduire au maximum le volume de données qu'il échange avec le serveur.

La compression des pages, le dernier avantage de l'optimisation, est le fait qu'un navigateur web qui respecte les standards ne dépassera pas quatre téléchargements à la fois. Si nous avons douze feuilles de style, le navigateur lancera donc un premier groupe de quatre téléchargements, attendra qu'ils s'achèvent et recommencera ainsi deux fois de suite en perdant à chaque fois le temps d'initialisation de la connexion.

Notez, que l'optimisateur de feuilles de style ne fonctionnera que si vous avez correctement déclaré vos fichiers CSS dans votre fichier `mon_theme.info`.

Optimisation des feuilles de style

Les scripts et les feuilles de style représentent de nos jours la majeure partie de ce qu'un navigateur a besoin de télécharger pour obtenir une page complète. Chaque module et thème pouvant potentiellement déclarer ses propres styles, ce sont très rapidement des dizaines de ces feuilles qui sont ainsi liées à une page. Pour s'en convaincre, nous pouvons utiliser l'onglet *Réseau* de Firebug :

Figure 15-43
L'onglet Réseau du module Firebug

⊕ GET www.colchique-conseil.fr	200 OK	colchique-conseil.fr	31 KB
⊕ GET admin_menu.css?g	200 OK	colchique-conseil.fr	4 KB
⊕ GET node.css?g	200 OK	colchique-conseil.fr	740 B
⊕ GET defaults.css?g	200 OK	colchique-conseil.fr	757 B
⊕ GET system.css?g	200 OK	colchique-conseil.fr	10 KB
⊕ GET system-menus.css?g	200 OK	colchique-conseil.fr	935 B
⊕ GET user.css?g	200 OK	colchique-conseil.fr	2 KB
⊕ GET content-module.css?g	200 OK	colchique-conseil.fr	3 KB
⊕ GET date.css?g	200 OK	colchique-conseil.fr	4 KB
⊕ GET datepicker.css?g	200 OK	colchique-conseil.fr	4 KB
⊕ GET timeentry.css?g	200 OK	colchique-conseil.fr	509 B
⊕ GET simplenews.css?g	200 OK	colchique-conseil.fr	610 B
⊕ GET fieldgroup.css?g	200 OK	colchique-conseil.fr	166 B
⊕ GET style.css?g	200 OK	colchique-conseil.fr	6 KB
⊕ GET elements-html.css?g	200 OK	colchique-conseil.fr	758 B

Comme vous le voyez, avec près de 4 Mo cumulés, le volume de ces feuilles de style n'est pas, loin de là, négligeable et rend ridicule les quelques 31 Ko de la page XHTML.

C'est pour cela que Drupal intègre un optimisateur de feuilles de style, dont le principe est de fabriquer une « super feuille de style » dans laquelle est ajouté le contenu de chacune des feuilles présentes, puis de la compresser, c'est-à-dire de la débarrasser de toutes les choses inutiles (commentaires, sauts de ligne, etc.). Pour activer cette fonction, toujours dans le panneau *Performances*, il suffit de changer la valeur de *Optimise les fichiers CSS* à *Activé*.

Figure 15-44
Activation de l'optimisation
des feuilles de style

Optimise les fichiers CSS :

☐ Désactivé

☒ Activé

Ceci fait, enregistrez la configuration et retournez sur la page d'accueil pour juger du résultat avec Firebug :

Figure 15-45
Le résultat est pour le moins
convaincant, n'est-ce pas ?

⊕ GET ad5577f642fd49b1b4c47e3d	200 OK	colchique-conseil.fr	22 KB
--------------------------------	--------	----------------------	-------

BONNE PRATIQUE L'optimisateur de code CSS

Il existe un outil particulièrement pratique appelé CSSTidy dont le but est de réduire la taille des fichiers CSS par une réorganisation interne. Selon le paramétrage de l'outil, l'optimisation peut aller du simple nettoyage, permettant d'obtenir un code plus lisible et maintenable, à une compression proche de ce que permet Drupal. Il est donc plus intéressant de l'utiliser en mode nettoyage.

► <http://csstidy.sourceforge.net/>

Outre la diminution du temps de chargement induite par cette réduction de taille, l'optimisation des feuilles de style vous permet de commenter proprement le code CSS sans risque d'impact sur les performances.

Optimisation des scripts

Drupal propose là aussi un système d'optimisation des scripts reposant sur le même principe que celui des feuilles de style. Ceci est d'autant plus intéressant qu'un navigateur comme Mozilla Firefox ne charge pas les scripts quatre par quatre, mais juste un à la fois.

GET jquery.js?C	200 OK	colchique-conseil.fr	31 KB
GET drupal.js?C	200 OK	colchique-conseil.fr	10 KB
GET fr_993fe8c085319526e887cbx	200 OK	colchique-conseil.fr	3 KB
GET admin_menu.js?C	200 OK	colchique-conseil.fr	2 KB
GET jquery.curvycorners.packed.	200 OK	colchique-conseil.fr	13 KB
GET ui.js?C	200 OK	colchique-conseil.fr	208 B

Figure 15–46

Les scripts chargés les uns après les autres par Mozilla Firefox

Comme pour les feuilles de style, l'optimisation des scripts supprimera les commentaires, les espaces en trop, et fabriquera un script unique qui se téléchargera donc plus rapidement.

Pour activer l'optimisation des scripts, activez comme pour les styles la zone correspondante du panneau de gestion des performances :

Optimise les fichiers Javascript :

- ☐ Désactivé
☒ Activé

Figure 15–47

Activation de l'optimisation des scripts

Vous pouvez ensuite juger du résultat avec Firebug :

GET e5328ced26f0ae258c954802e	200 OK	colchique-conseil.fr	57 KB
-------------------------------	--------	----------------------	-------

Figure 15–48 Il n'y a plus qu'un seul fichier JavaScript de taille réduite.

L'impact est moins flagrant que pour les styles, mais reste appréciable pour minimiser le nombre de connections.

Vidange des caches

L'activation de toutes ces optimisations n'est pas une bonne chose sur une machine de développement, car elles ajoutent une couche de comportement qui rend délicat les tests et autres débogages.

De même, les caches et autres techniques d'optimisation peuvent également nous gêner sur une machine de production pour valider certains paramètres. Il est donc nécessaire de pouvoir en faire la purge quand le besoin s'impose.

Pour vider les caches du côté serveur, vous avez sur la page de gestion des performances en fin de formulaire un bouton prévu à cet effet :

Supprimer les données du cache

Utiliser la fonction de cache augmente les performances, mais si des informations sont mises en cache. Pour éviter le mauvais fonctionnement des sites à fort trafic peut ralentir le site.

Supprimer les données du cache

Figure 15–49

Bouton de vidange des caches de Drupal

MODULES MAGIQUES Comprimez vos JavaScript

Si vous comptez beaucoup de scripts, une autre possibilité est d'utiliser le module JavaScript Aggregator. Ce dernier fait la même chose que le système natif de Drupal en ajoutant en plus l'utilisation de JSMIn, un outil évolué de compression de script. À vous de tester cette approche si vous considérez le script produit par Drupal encore trop gros. La procédure d'installation est simple et lisible sur la page de projet du site. Ce module est aussi la seule option de compression de scripts que vous avez, si vous devez encore travailler avec Drupal 5.

- http://drupal.org/project/javascript_aggregator
- <http://code.google.com/p/jsmin-php/>

BON À SAVOIR Vidage de cache en ligne de commande

Le module Drush, que nous avons vu un peu plus haut, permet d'opérer en ligne de commande une vidange complète des caches, ce qui se révèle être plus que pratique en phase de test où l'opération doit être sans cesse renouvelée :

```
./drush cache c'lean
```

MODULES MAGIQUES Le cron du pauvre humain

Si vous ne pouvez ou ne voulez pas utiliser le planificateur de tâche de Windows ou le cron Unix, il vous reste l'option du module Poormanscron. Ce module lance le cron Drupal selon une fréquence paramétrable en profitant de la requête d'un de vos visiteurs.

Ce système ne ralentit pas l'utilisateur, car son exécution se fait après avoir envoyé la page au navigateur client. En revanche, si vous avez un très faible trafic, il se peut que la fréquence ne soit pas respectée.

► <http://drupal.org/project/poormanscron>

Il suffit d'appuyer sur ce bouton pour vidanger l'ensemble des caches, y compris les scripts et feuilles de style optimisées. Attention cependant à ne pas abuser de ce bouton sur un site qui reçoit beaucoup de monde, car vous allez obligatoirement créer un ralentissement le temps que les caches se reforment.

Cron

Jusqu'à maintenant, nous gérons le lancement du cron de Drupal à la main en utilisant `cron.php` dans le navigateur. En production, il va nous falloir trouver un moyen plus automatique pour lancer ce script vital à toute installation Drupal.

Sous Unix, qui est le système d'exploitation majoritairement utilisé pour un site web public, la solution idéale pour activer le cron de Drupal est d'utiliser le démon `crond`.

Cela consiste à ajouter dans le dossier `/etc/crontab`, une nouvelle table cron semblable à celle-ci :

```
#!/bin/bash
# /etc/cron.d/php: crontab pour Drupal

# Look for and purge old sessions every 30 minutes
*/15 * * * * root /usr/bin/lynx -source http://www.colchique-
conseil.fr/cron.php > /dev/null 2>&1
```

Le principe est ici d'utiliser l'utilitaire `Lynx`, un navigateur web en ligne de commande présent sur toutes les distributions, pour aller chercher la page `cron.php` et ainsi déclencher le cron de Drupal. Notez que la même chose est réalisable avec `wget`.

Sous Microsoft Windows, l'affaire se complique un peu. En effet, d'une version à l'autre, les choses ne fonctionnent pas de la même manière, du moins pour l'aspect graphique. Heureusement, il est possible de programmer le planificateur de tâches en ligne de commande. Le principe est ici d'utiliser Mozilla Firefox pour aller chercher la page `cron.php`.

```
C:\> schtasks /create /tn "Drupal Cron Job" /tr "c:\Program
Files\Mozilla Firefox\firefox.exe http://www.colchique-
conseil.fr/cron.php" /sc hourly
```

La commande vous demandera alors le nom et l'identifiant de l'utilisateur qui lancera le processus :


```
La tâche va être créée sous le nom de l'utilisateur local
actuellement connecté
("A-283AE68F6F5F4\Gaston").
Entrez le mot de passe pour A-283AE68F6F5F4\Yoran dans
"Exécuter en tant que" :
*****
```

Une fois la commande correctement lancée, le planificateur de tâche exécutera le cron de Drupal toutes les heures.

BON À SAVOIR Comment être certain que le cron est opérationnel ?

Dans un premier temps, pour vérifier que le cron fonctionne, vous pouvez créer une action « Envoyer un courriel à » en l'associant au déclencheur Lancement de cron. Vous serez ainsi notifié par courriel à chaque lancement du cron. Lorsqu'il est établi que le cron fonctionne, vous pouvez supprimer cette règle.

Gérer le multisite

Nous avons créé notre premier site Drupal, et cela peut vraisemblablement donner envie d'en créer d'autres. La question de faire coexister plusieurs sites Drupal sur une seule et même machine peut alors se poser.

Comme vous l'avez vu, toute la configuration pour Drupal se trouve dans la base de données. Il peut donc sembler légitime d'imaginer un même Drupal avec plusieurs bases de données, chacune pour un site différent, et c'est très exactement ainsi que cela se fonctionne.

Toute la magie tient dans le seul dossier `sites` dont nous n'avons jusqu'à maintenant utilisé que deux sous-dossiers : `default` qui contient notre configuration `settings.php`, et `all` qui contient nos thèmes et nos modules. Imaginons à présent que nous voulions ouvrir un second site <http://www.colchique-conseil.fr/blogs/marine>, que Marine utilisera comme son blog personnel.

Il faut alors que Drupal puisse trouver une nouvelle configuration `settings.php`, lui permettant d'utiliser la seconde base de données. Pour cela, il recherchera comme le fait PHP Engine dans le dossier `sites`, parmi une liste de noms de dossiers éligibles.

- `www.colchique-conseil.fr.blogs.marine` : nous avons ici l'adresse complète du site en remplaçant juste les slashes par des points.
- `colchique-conseil.fr.blogs.marine` : la même chose, mais cette fois en ôtant le sous-domaine `www`.
- `fr.blogs.marine` : ici, Drupal enlève le domaine pour ne laisser que le domaine de premier niveau (*tld*, pour *top-level-domain*) et le chemin.

Si aucun dossier éligible n'existe dans le dossier `sites`, Drupal recommence, mais cette fois en enlevant le dernier dossier, ce qui nous donne :

- `www.colchique-conseil.fr.blogs` ;
- `colchique-conseil.fr.bloc` ;
- `fr.blogs`.

Vous l'avez maintenant compris : si Drupal ne trouve toujours pas de dossier, il recommence, mais cette fois sans aucun sous-dossier dans l'URL :

- `www.colchique-conseil.fr` ;
- `colchique-conseil.fr` ;
- `fr`.

Enfin, si rien n'a marché, il se rabat sur le sous-dossier `default`. C'est pour cela que nous pouvions jusqu'à maintenant l'utiliser.

Pour que nos deux sites fonctionnent, il faut donc commencer par recopier `default` vers un dossier `colchique-conseil.fr`. Ainsi nous aurons le même comportement qu'avant. Notez au passage que ne pas spécifier le `www` permettra à notre site de fonctionner correctement lorsque nous disposerons d'une version « mobile » de la forme `http://mobile.colchique-conseil.fr/`.

Créons ensuite un nouveau dossier `colchique-conseil.fr.blogs.marine`, et allons y recopier le fichier `settings.php` provenant du dossier `default`.

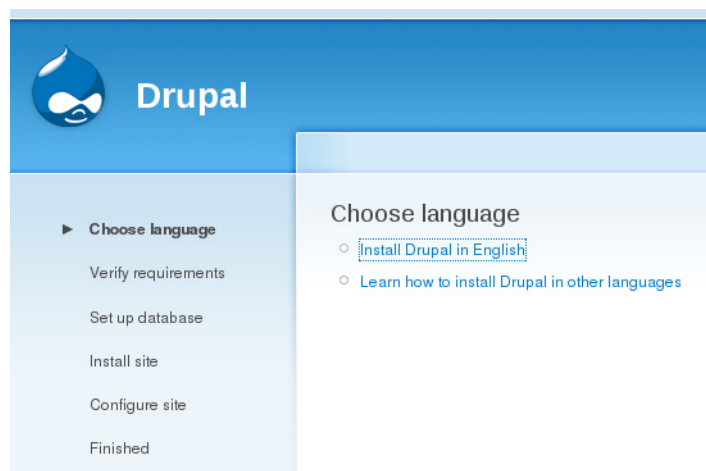


Figure 15-50
Le blog de Marine

Nous voici avec un site vierge que nous pouvons installer et paramétrer comme le premier, et qui n'entre pas en conflit avec lui. Pour preuve : en allant cette fois sur `http://www.colchique-conseil.fr/`, nous voilà de nouveau sur le site que nous connaissons bien :



Figure 15-51
Le site Colchique classique

Il est possible d'aller plus loin encore, en disposant de modules et de thèmes spécifiques par site. Pour cela, il suffit de créer dans le dossier de chaque site une arborescence `modules` et `themes` et d'y décompresser les archives, comme dans le dossier `a11`. Tout ce qui se trouvera dans le dossier d'un site ne sera vu que par lui.

Ceci nous permet de comprendre l'utilité du dossier `a11` que nous utilisons jusqu'alors, car tout ce qui s'y trouve est visible pour l'ensemble des sites.

En résumé


- Pensez à consulter régulièrement les mises à jour disponibles (ou encore mieux, faites-vous prévenir), il en va de la sécurité de votre site.
- Il ne suffit pas de mettre à jour le code, il faut aussi mettre à jour la base de données.
- Testez, testez et testez encore sur votre machine de développement, ne jouez *jamaïs* à l'apprenti sorcier sur un serveur en production.
- Les performances sont à optimiser sur le serveur de production : rien ne sert d'appliquer des réglages de performance sur votre machine de développement, vous ne feriez que vous compliquer la tâche.
- N'oubliez pas d'activer le lancement automatique du cron de Drupal.

MODULES MAGIQUES Partagez vos contenus entre plusieurs sites

Une autre manière d'utiliser la même installation Drupal pour plusieurs sites est d'utiliser le module Domain Access qui permet également de partager la même base de données. Le domaine d'utilisation n'a plus rien à voir ici et présente l'avantage de permettre le partage de contenus entre plusieurs sites.

► <http://drupal.org/project/domain>

chapitre 16

**Drupal 7**

[Add a main menu link](#)

[My account](#)

[Log out](#)

[Home](#) > [Add new content](#)

Management

[Add new content](#)

[Article](#)

[Page](#)

[Administer](#)

Create Article

Title: *

Tags:

Enter a comma-separated list of words.

Body

Summary:

Leave blank to use trimmed value of full text as the summary.

Full text:

Text format: Filtered HTML ▾ [More information about text formats](#) ⓘ

☐ Web page addresses and e-mail addresses turn into links automatically.

☐ Allowed HTML tags: <a> <cite> <blockquote> <code> <dl> <dd>

☐ Lines and paragraphs break automatically.

Menu settings

Not in menu

[Revision information](#)

No revision

[URL path settings](#)

Menu link title:

The link text corresponding to this item that should appear in the menu. Leave blank if you do not wish to add this post to the menu.

Un regard sur Drupal 7

Drupal 7 n'est pas encore sorti mais les contours de cette nouvelle mouture commencent à se préciser. Rassurez-vous, aucun de vos acquis ne sera remis en cause par cette version qui se place dans la continuité de la philosophie de Drupal. Elle apportera trois évolutions majeures et plusieurs autres améliorations, avec pour objectif commun de rendre Drupal plus simple à utiliser tout en améliorant ses performances et en l'ouvrant à d'autres systèmes de bases de données.

SOMMAIRE

- ▶ Découvrez les prochaines améliorations ergonomiques
- ▶ CCK directement dans le cœur de Drupal ?
- ▶ Un nouveau moteur de base de données
- ▶ Et de nombreuses autres nouveautés...

MODULES MAGIQUES Trop impatient pour attendre Drupal 7 ?

Eh bien, réjouissez-vous car il existe un module Vertical Tabs qui permet de doter Drupal 6 de cette merveilleuse amélioration ergonomique pour l'édition des contenus.

► http://drupal.org/project/vertical_tabs

Figure 16–1
Le nouveau mode de
présentation des réglages
par onglets verticaux

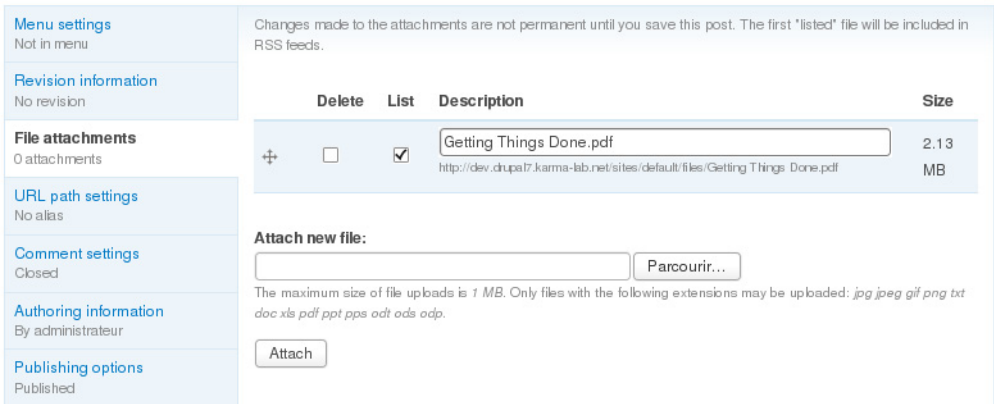
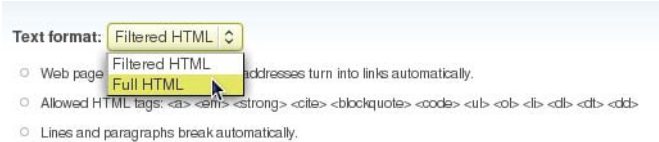


Figure 16–2
Sélection des formats de texte



Amélioration de l'ergonomie d'utilisation

La première chose que vous noterez en découvrant Drupal 7 sera très probablement la grande amélioration de l'ergonomie d'ensemble. Ce sujet est devenu un enjeu à part entière, au point de lui dédier pour cette nouvelle version une équipe de travail exclusive (<http://groups.drupal.org/usability>).

Pour prendre un exemple, si les interminables séries de sections déplaçables – typiquement sur la page d'édition des contenus – vous rebutaient, vous vous réjouirez sans nul doute du nouveau mode de présentation par onglets verticaux. Plus agréable et surtout beaucoup plus compact, ce nouveau mode de présentation vous permettra de sélectionner les différents groupes de réglage en quelques clics.

Toujours dans l'optique d'améliorer l'ergonomie de l'édition des contenus, la sélection des formats d'entrée a été grandement simplifiée. Les boutons radio ont disparu au profit d'une liste déroulante qui, lorsque vous sélectionnez un format, affiche dynamiquement l'aide associée.

De même, le terme de « format d'entrée » a été remplacé par « Text format » (format du texte), ce qui est tout de même plus cohérent.

Dans le même esprit, un système d'aide avancée a été intégré. Il est ainsi possible de disposer d'informations détaillées sur un élément de l'interface en cliquant sur l'un des points d'interrogation sur fond bleu.

<input type="checkbox"/>	Book	7.x-dev	Allows users to create and organize related content in an outline.	
<input checked="" type="checkbox"/>	Color	7.x-dev	Allows administrators to change the color scheme of compatible themes.	? More help
<input checked="" type="checkbox"/>	Comment	7.x-dev	Allows users to comment on and discuss published content. Required by: Forum (disabled), Tracker (disabled)	? More help

Figure 16–3
Présentation
de l’aide avancée

De nombreux petits détails ont également évolué, et continueront de le faire jusqu’à la version finale, afin de toujours améliorer l’utilisation :

- le menu d’administration contient davantage de sous-catégories, comme la *Gestion des langues* ;
- le choix du thème d’administration a été placé dans l’administration des thèmes ;
- une explication est placée à côté des permissions pour en expliquer l’usage ;
- notre astuce du rôle Administrateur pour déléguer l’administration du site est devenu une fonctionnalité officielle.

Intégration de CCK

Vous l’avez certainement compris, CCK est un module indispensable à Drupal et c’est pourquoi Drupal 7 l’intégrera, du moins en partie.

Le principe choisi est de garder le module CCK pour l’aspect graphique de l’édition des types de contenu, et d’intégrer intimement la gestion interne des champs au cœur même de Drupal.

Cette nouvelle fonctionnalité s’appellera FieldAPI. Elle fonctionnera donc comme l’actuel CCK en y ajoutant de nouvelles fonctionnalités, comme pouvoir changer dynamiquement le type d’un champ.

Mais FieldAPI va plus loin encore. Étant directement intégré à Drupal, il distribuera ses bienfaits sur l’ensemble du système entraînant la disparition de la frontière entre les champs natifs d’un contenu et les champs ajoutés. Tous les champs d’un contenu (titre, taxonomie, menu, etc.) pourront dès lors être manipulés comme s’ils étaient des champs CCK.

Article		
	Edit	Manage fields Display fields
Add fields to the Content type, and arrange them on content display and input for You can add a field to a group by dragging it below and to the right of the group.		
Label	Name	Field
Title	Node module element.	
Taxonomy	Taxonomy module element.	
Menu settings	Menu module element.	
File attachments	Upload module element.	

Figure 16–4
Tous les champs natifs deviennent des champs CCK.

Enfin, FieldAPI sera aussi utilisé pour toutes les données du système qui n'étaient pas encore vues comme de réels contenus. C'est notamment le cas des commentaires, et du profil des utilisateurs.

User settings

EditManage fieldsDisplay fields

Add fields to the user, and arrange them on user display and input forms.
You can add a field to a group by dragging it below and to the right of the group.

Label	Name	Field	Widget	Operations
+	User name and password	User module form element		
+	History	User module view element.		
+	Timezone	User module form element.		

Add

+

New field

Label

field_

Field name (a-z, 0-9, _)

- Select a field type -

Type of data to store.

- Select a widget -

Form element to edit the data.

+

New group

Label

group_

Group name (a-z, 0-9, _)

Save

Figure 16-5 FieldAPI permettra d'ajouter des champs au profil des utilisateurs, comme cela était possible pour les types de contenu.

Un nouveau système de base de données

Depuis ses débuts, Drupal évolue pour s'affranchir peu à peu de la base de données sous-jacente mais il est au final toujours resté bloqué sur MySQL et PostgreSQL, même si des projets permettent de rendre Drupal 6 compatible notamment avec Oracle et SQLServer.

Avec Drupal 7, les choses changent radicalement grâce à l'adoption de PDO (*PHP Data Objects*). PDO est une extension de PHP qui peut se comparer dans une certaine mesure à JDBC (*Java Database Connectivity*) ou encore à ODBC (*Open Database Connectivity*). Il s'agit dans les trois cas d'une abstraction d'accès à la base de données.

Les avantages de PDO sont nombreux. Tout d'abord, l'audience de PDO est plus importante que celle de Drupal, ce qui implique un support plus large et plus poussé pour davantage de bases de données. Il existe déjà aujourd'hui des pilotes PDO pour MySQL, MSSQL, SQLite, Firebird,

URL **Projet OraDrup**

► <http://www.oracle.com/technology/pub/articles/pedros-drupal.html>

URL **Drupal SQL Server compatibility**

► <http://drupal.org/node/1394>

URL **Pilote PDO pour Oracle**

► http://www.oracle.com/technology/pub/articles/php_experts/otn_pdo_oracle5.html

PostgreSQL, ODBC, Oracle, etc. Cela signifie que Drupal, avec la version 7, s'ouvre potentiellement à toutes ces bases de données.

Ensuite, PDO est écrit en langage compilé, on peut donc légitimement s'attendre à une amélioration des performances d'accès aux données.

Enfin, PDO rend générique des fonctionnements SGBD qui manquaient cruellement à Drupal, comme la gestion des transactions ou la prise en charge des architectures redondantes.

Autres évolutions notables

Amélioration des performances

Au regard de Drupal 5, Drupal 6 constitue déjà un formidable bond en avant en termes de performances. La raison de ce gain a été l'implémentation d'une technique permettant de ne charger que le minimum de fichiers PHP. Avec Drupal 6, chaque module a été séparé en deux lots, les fichiers PHP importants et ceux d'usage plus épisodique (typiquement l'administration). Ainsi, seuls les fichiers importants sont chargés lorsqu'une page est construite.

Drupal 7 continue dans cette voie de manière encore plus radicale en mettant en place un registre du code PHP (*code registry*). Le principe est que Drupal 7 aura connaissance de chacune des fonctions déclarées par chacun des modules, et ne chargera en mémoire que le code strictement nécessaire à la fabrication de la page.

Cette nouvelle fonctionnalité permet d'obtenir de meilleures performances que Drupal 6, en réduisant sensiblement le temps d'initialisation de Drupal.

L'ouverture du moteur de recherche

Le moteur de recherche actuel de Drupal étant difficile à étendre, une nouvelle architecture de recherche est donc introduite avec Drupal 7, avec pour objectif de faciliter l'intégration d'autres moteurs de recherche performants, tels que Xapian, Apache Solr, etc.

Une refonte des droits sur les contenus

Tous les modules de gestion des droits associés aux contenus que nous avons abordés dans cet ouvrage souffrent du même problème : leur potentielle incompatibilité les uns avec les autres. Drupal 7 propose une mise à plat de la gestion des droits, qui rendra possible la coexistence simultanée de plusieurs modèles.

En résumé

- Drupal 7 ne changera rien à vos habitudes.
- L'ergonomie générale sera considérablement améliorée.
- L'intégration de PDO permet d'imaginer une plus grande ouverture vers un plus grand nombre de bases de données.
- Les performances devraient être encore meilleures que celles de Drupal 6.

Index

403 (erreur) 257
404 (erreur) 257
404 Blocks (module) 258

A

accélérateur PHP 383
 APC 383
 eAccelerator 383
 xcache 383
Acquia Marina (thème) 254
action 340
Action (module) 340
administrateur 245
administration 66
 compte utilisateur 107
 panel 371
 thème d'administration 259
Advanced forum (module) 74
Aggregator (module) 74
agrégateur 299
alias 132
AlternC 371
Apache 39, 45
APC 383
API 19
apparence : thème 244
architecture 18
article 126
 aperçu 131
 attacher un fichier 129
 format d'entrée 142
 publication 134
 résumé 128
Atom (format) 302
Atom (module) 302
attacher un fichier 129
avatar 102

B

base de données 23, 39, 396
 création 53
 dump 363
 mise à jour 363
 mySQL 364
 postgresql 364

 problème 62
 sauvegarde 363
 vidage 363
Basic (thème) 263
Bibliothèque jQuery 268
bloc 204
 ajouter 220
 bloc de menu 220
 gérer 209
 personnalisé 207
 visibilité 209
Block (module) 71
Block Visibility Page (module) 214
block.tpl.php 261
Blocks 404 (module) 258
Blog (module) 74
Blog API (module) 74
Bluemarine (thème) 248
boîte à outils image 83
Book (module) 74, 160
 Book Navigation (bloc) 206
 Book page 160
bulletin d'informations 307

C

cache 24, 380
 de page 381
 des blocs 381
 des contenus 381
 des formats d'entrée 381
 des menus 381
 des vues 381
 paramétrage 382
 vidage 387
Cache Router (module) 384
calendrier 187
CCK (module) 180, 395
chaîne de transformations 21
champ
 ajout 183
 de recherche 249
 multiple 185
Checkout (module) 131
chemin
 gestion 217
 interne 132

 préfixe 230
classification
 contenu 167
 terme 168
 vocabulaire 169
CMS
 contenu 9
 histoire 9
 présentation 10
Coder (module) 355
Color (module) 74
commande
 diff 362
 patch 362
Comment (module) 74
 permission 94
Comment mover (module) 139
Comment RSS (module) 303
Comment Notify (module) 305
comment.tpl.php 261
commentaire 13, 123, 137
 flux 303
 format d'entrée 147
 notification 305
compression de page 385
compte
 création 106
 FTP 371
 utilisateur 98
 administrer 107
configuration du site 56
contact (formulaire) 318
Contact (module) 74, 319
 permission 96
Content Management System (CMS) 9
Content translation (module) 74, 237
contenu 9, 13, 20
 aperçu 131
 article 126
 attacher un fichier 129
 classement 167
 dynamique 11
 flux 300
 format d'entrée 142
 gestion 67
 historisation 13

- liste de contenus 192
- livre 160
- nœud (node) 71, 85
- notation 325
- résumé 128
- traduction 228, 237
- type 10, 31, 120, 178

- core 71
- courrier de confirmation 62
- cron 69, 312, 388
- CSS 265, 368
- CSSTidy 386

D

- Database logging (module) 74, 385
- date (calendrier) 187
- Date (module) 186
- Datestamp 186
- DateTime 186
- déclencheur 340
- décompression 50
- default.settings.php 52
- déploiement 371
 - hébergement 369
 - maintenance 375
 - nom de domaine 370
- Deploy (module) 376
- désinstaller un module 81
- Devel (module) 283, 355, 378
- développeur 245
- diff (commande) 362
- Diff (module) 167
- dimensionnabilité 24
- disposition d'un bloc 204
- DNS (Domain Name Server) 370
- document
 - état 334
- Domain Access (module) 391
- Domtopdf 322
- dossier
 - sites 52
 - structure 51
- droit 397
 - d'accès 199
 - gestion 23
 - problème 60
- Drupal
 - architecture 18
 - Drupal 7 393
 - historique 18
 - installation 51

- licence 25
- mise à jour 58, 358
- paramétrage 52, 83
- performances 378
- téléchargement 49
- traduction 50
- Drush (module) 365

E

- eAccelerator 383
- éditeur visuel 150
- élément de menu 218
- entrée (format) 142
- environnement
 - LAMP 44
 - LAPP 40
 - MAMP 40
 - WAMP 40, 41
- ergonomie 28, 394
- erreur
 - 403 257
 - 404 257
 - message d'erreur 60
 - page d'erreur 257
- état 13, 334
- étiquetage 22
- étiquette (nuage) 177
- extension 260, 263
- External links filter (module) 148

F

- Faq (module) 165
- favicon 250
- feuille de style 265
 - optimisation 385
- fichier (attacher) 129
- FieldAPI 395
- Filter (module) 71
- Filter by node type (module) 150
- Filtered HTML 144
- filtre 21, 145
 - ajout 148
 - PHP 147
- Firebug 265
- Fivestar (module) 325
- flux
 - Atom 302
 - de travail 31, 334
 - personnalisé 303
 - RSS 298
 - annonce de sécurité 358
- fonction invoke_all() 351

- format
 - d'entrée 21, 142, 394
 - commentaire 147
 - Filtered HTML 144
 - PDF 321
 - PO 236
 - XHTML 21
- formulaire de contact 318
- Forum (module) 74
- FTP 371

G

- Garland (thème) 204, 247, 251
- gestion
 - bloc 209
 - chemin 217
 - contenu 67
 - menu 216
 - module 70
 - révision 165
 - thème 247
 - utilisateur 67
- GNU/Linux 40, 45
- GPL 25
- Gravatar 316
- Gravatar Integration (module) 317

H

- hébergement 369
- hook
 - de module 350
 - de thème 278
- hook_link 351
- HTTP service 38

I

- I18N (module) 239, 240
- impression 321
- installation
 - Drupal 51
 - message d'erreur 60
 - mise à jour 358
 - module 80
- interface (traduction) 228
- invoke_all() 351

J

- JavaScript Aggregator (module) 387
- journal 385
- jQuery 268
- jQuery Corner 269
- JSMIn 387

- L**
- LAMP 44
 - langage
 - PHP 39
 - SQL 40
 - langue 51, 227
 - liste 228
 - négociation 229
 - LAPP 40
 - lettre d'informations 307
 - licence 25
 - GPL 25
 - lien
 - primaire 250
 - secondaire 250
 - Liens primaires (menu) 216
 - Liens secondaires (menu) 216
 - Linux (GNU) 40, 45
 - liste
 - de contenus 192
 - des langues 228
 - livre 160
 - Page de livre 160
 - Locale (module) 74
 - Localization Client (module) 234, 236
 - logo 249
- M**
- Mac OS 40
 - maintenance 375
 - MAMP 40
 - Markdown (module) 149
 - Masquerade (module) 110
 - Meld 362
 - menu
 - élément de menu 218
 - gérer 216
 - Liens primaires 216
 - Liens secondaires 216
 - Navigation 216
 - Menu (module) 75
 - message d'erreur 60
 - MimeMail (module) 311
 - mise à jour 58, 358
 - base de données 363
 - Drupal 361
 - module 363
 - restaurer 366
 - tableau de bord 359
 - thème 363
 - update.php 364
 - Mobile Tools (module) 254
 - modèle 259, 270
 - `$base_path` (variable) 272
 - `$block` 274
 - `$block->content` (variable) 274
 - `$block->delta` (variable) 274
 - `$block->module` (variable) 274
 - `$block->subject` (variable) 274
 - `$body_classes` (variable) 272
 - `$breadcrumb` (variable) 272
 - `$closure` (variable) 272
 - `$comment` (variable) 277
 - `$content` (variable) 272, 276, 277
 - `$footer` (variable) 273
 - `$front_page` (variable) 272
 - `$head` (variable) 272
 - `$head_title` (variable) 272
 - `$header` (variable) 273
 - `$help` (variable) 272
 - `$language` (variable) 271
 - `$left` (variable) 273
 - `$link` (variable) 276
 - `$logo` (variable) 272
 - `$messages` (variable) 272
 - `$mission` (variable) 272
 - `$new` (variable) 277
 - `$node` (variable) 276, 277
 - `$page` (variable) 276
 - `$picture` (variable) 275, 277
 - `$primary_link` (variable) 272
 - `$right` (variable) 273
 - `$script` (variable) 272
 - `$search_box` (variable) 272
 - `$secondary_link` (variable) 272
 - `$signature` (variable) 277
 - `$site_name` (variable) 272
 - `$site_slogan` (variable) 272
 - `$status` (variable) 275, 277
 - `$sticky` (variable) 275
 - `$styles` (variable) 272
 - `$submitted` (variable) 276, 277
 - `$tabs` (variable) 272
 - `$taxonomy` (variable) 276
 - `$teaser` (variable) 275
 - `$title` (variable) 272, 277
 - bloc 274
 - bloc.tpl.php 274
 - block.tpl.php 261
 - comment.tpl.php 261, 276
 - contenu 275
 - enrichissement 289
 - node.tpl.php 261, 275
 - nom de modèle éligible 280
 - page.tpl.php 261, 270
 - path_to_theme() 272
 - phptemplate_preprocess() 292
 - preprocess 292
 - theme() 278
 - module 76
 - 404 Blocks 258
 - Action 340
 - Advanced forum 74
 - Aggregator 74
 - anomalie 82
 - Atom 302
 - Block 71
 - Block Visibility Page 214
 - Blocks 404 258
 - Blog 74
 - Blog API 74
 - Book 74, 160
 - Cache Router 384
 - CCK 180, 395
 - Checkout 131
 - choisir 77
 - code 353
 - Coder 355
 - Color 74
 - Comment 74, 94
 - Comment mover 139
 - Comment RSS 303
 - Comment Notify 305
 - Contact 74, 96, 319
 - Content profil 116
 - Content translation 74, 237
 - core 71
 - créer 348
 - Database logging 74, 385
 - Date 186
 - dépendances 76
 - Deploy 376
 - désinstaller 81
 - Devel 355, 378
 - Diff 167
 - Domain Access 391
 - Drush 365
 - External links filter 148
 - Faq 165
 - fichier de description 352
 - Filter 71
 - Filter by node type 150
 - Fivestar 325

Forum 74
 gestion 70
 Gravatar Integration 317
 hook 350
 I18N 239
 installer 80
 JavaScript Aggregator 387
 Locale 74
 Localization Client 234, 236
 Markdown 149
 Masquerade 110
 Menu 75
 MimeMail 311
 mise à jour 363
 Mobile Tools 254
 Node 71, 95
 Node Access 199
 nommage 78
 Notification 307
 OpenID 75
 OpenID URL 314
 Organic Group 201
 Path 75, 96
 Path Redirect 137
 Pathauto 134
 PHP filter 75, 147
 Ping 75
 Poll 75
 Poormanscron 388
 Printer 322
 Profile 75
 Quick Tabs 209
 RootCandy 259
 Rules 345
 Schema 355
 Search 75, 97, 249
 Search 404 137
 Simple Access 201
 Simplenews 307
 Statistics 76
 structure interne 352
 Subscriptions 307
 Syslog 76, 385
 Tagadelic 177
 Taxonomy Access Control 201
 Taxonomy navigator 177
 téléchargement 76
 Theme Developer 283
 Throttle 76
 Tracker 76
 Trigger 76, 340

Update status 76, 80, 359
 Upload 76, 87, 97, 129
 Upload Path 130
 User 71, 97
 Vertical Tabs 394
 Views 192, 240, 303, 327
 Voting API 325
 Workflow 335
 Wysiwyg 151
 mot-clé 10
 étiquetage 22
 moteur
 de recherche 397
 de rendu 34
 de thème 22, 244
 PHPTemplate 245
 multilingue 240
 multisite 389
 MySQL 47, 364

N

Navigation (menu) 216
 négociation de la langue 229
 newsletter 307
 Node (module) 71
 permission 95
 node (nœud) 71, 85, 132
 Node Access (module) 199
 node.tpl.php 261
 nœud 132
 nom
 de domaine 370
 du site 249
 nommage (module) 78
 norme
 CSS 368
 XHTML 368
 notation 325
 notification 305
 Notifications (module) 307
 nuage d'étiquettes 177

O

OpenID 313
 OpenID (module) 75
 OpenID URL (module) 314
 optimisation
 feuille de style 385
 script 386
 Organic Group (module) 201
 outil
 Akregator 298

AlternC 371
 APC 383
 CSSTidy 386
 décompression 50
 Domtopdf 322
 eAccelerator 383
 FeedReader 298
 Firebug 265
 Gimp 265
 Inkscape 265
 JSMIn 387
 Liferea 299
 Meld 362
 Photoshop 265
 serveur OpenID 313
 TCPDF 322
 WebKit 322
 WinMerge 362
 XCache 383

P

package 352
 page
 bloc 204
 compression 385
 d'erreur 257
 de garde 240
 de livre 160
 introuvable 257
 page.tpl.php 261, 270
 panel 369
 AlternC 371
 paramétrage 83
 cache 382
 compte utilisateur 98
 des blocs 253
 Drupal 52
 patch (commande) 362
 Path (module) 75
 permission 96
 Path Redirect (module) 137
 path_to_theme() 284
 Pathauto (module) 134
 PDF 321
 Domtopdf 322
 TCPDF 322
 WebKit 322
 PDO (PHP Data Objects) 396
 performance 24, 377
 cache 24
 Drupal 7 397

permalien 133
 permalink 133
 permission 12, 23, 92
 PHP 39
 accélérateur 383
 PDO (PHP Data Objects) 396
 PHP filter (module) 75
 phpMyAdmin 42, 48
 PHPTemplate (moteur de thème) 245, 259
 Ping (module) 75
 planet 299
 plate-forme
 LAMP 44
 LAPP 40
 MAMP 40
 WAMP 40, 41
 PO (format) 236
 Poll (module) 75
 Poormanscron (module) 388
 portrait 102
 PostgreSQL 190, 364
 préfixe (langue) 230
 preprocess_page 291
 présentation 10, 204
 Printer (module) 322
 problème
 base de données 62
 courrier de confirmation 62
 droits 60
 SMTP 62
 production
 hébergement 369
 maintenance 375
 nom de domaine 370
 profil 113
 Profile (module) 75
 publication d'un article 134

Q

Quick Tabs (module) 209

R

rapport 68
 recherche
 champ 249
 moteur 397
 région 204
 réglage
 alertes 86
 boîte à outils image 83
 commentaire 123

courriel 101
 date et heure 84
 informations 85
 journalisation 86
 mode privé 89
 mode public 89
 portrait 102
 rapports d'erreur 86
 règle d'accès 111
 signature 102
 système de fichiers 88
 transfert de fichiers 87
 restaurer 366
 résumé 128
 révision 165
 rôle 12, 23, 31, 92
 RootCandy (module) 259
 RSS 298
 annonce de sécurité 358
 lecteur de flux 298
 Rules (module) 345

S

sauvegarde
 base de données 363
 scalability 24
 Schema (module) 355
 script 268
 optimisation 386
 update.php 364
 script.js 268
 Search (module) 75, 249
 permission 97
 Search 404 (module) 137
 serveur 38
 DNS (Domain Name Server) 370
 OpenID 313
 service 38
 HTTP
 Apache 39
 SMTP 42
 settings.default.php 59
 settings.php 52, 56, 59, 60, 68, 364, 375, 384
 SGBD 40
 signature 102
 Simple Access (module) 201
 Simplenews (module) 307
 site
 configuration 56, 66
 construction 66

 dossier 52
 multisite 389
 slogan 249
 SMTP 42
 SQL 40
 Statistics (module) 76
 storyboard 28
 structure d'un dossier 51
 style 196, 263
 de ligne 196
 Subscriptions (module) 307
 syndication 298
 agrégateur 299
 Akgator 298
 annonce de sécurité 358
 Atom 302
 FeedReader 298
 flux personnalisé 303
 Liferea 299
 planet 299
 RSS 298
 Syslog (module) 76, 385
 système d'exploitation 40
 GNU/Linux 40, 45
 Mac OS 40
 Windows 40
 système de gestion des contenus (SGC) 9

T

tableau de bord 68
 mise à jour 359
 Tagadelic (module) 177
 taxon 10
 taxonomie 10, 22, 167
 étiquetage 22
 flux 302
 terme 168
 vocabulaire 169
 Taxonomy (module) 76
 Taxonomy Access Control (module) 201
 Taxonomy navigator (module) 177
 TCPDF 322
 téléchargement de Drupal 49
 template 259, 288
 terme 168
 test
 fonctionnel 367
 technique 367
 thème 244, 288
 Acquia Marina 254
 Basic 263

- Bluemarine 248
- création 259
- d'administration 259
- Garland 204, 247, 251
- gestion 247
- hook 278
- mise à jour 363
- moteur 22, 244
- nouveau 254
- paramétrage des blocs 253
- starter 263
- theme() 278, 284
- theme_username() 286
- Zen 263
- Theme Developer (module) 283
- thèmeur 245
- Throttle (module) 76
- TinyMCE 151
- Tracker (module) 76
- traduction 50
 - ajout 234
 - contenu 228, 237
 - import/export 236
 - interface 228
- Trigger (module) 76, 340
- troll 141

type de contenu 31, 120, 178

U

- Update status (module) 76, 80, 359
- update.php 364
- Upload (module) 76, 87, 129
 - permission 97
- Upload Path (module) 130
- URL simplifiée 58, 61
- User (module) 71
 - permission 97
- utilisateur 92
 - compte 98, 106, 107
 - gestion 67, 104
 - Mon compte (page) 112
 - permission 12, 92
 - profil 113
 - règle d'accès 111
 - rôle 12, 31, 92

V

- variables des modèles
 - \$language 271
 - \$language->dir 271
- Vertical Tab (module) 394
- Views (module) 192, 240, 303, 327
- visibilité d'un bloc 209

- vocabulaire 169
- Voting API (module) 325
- vue 192
 - duplication 198

W

- W3C 368
- WAMP 40, 41
- WampServer 41
- Watchdog 385
- Web (histoire) 8
- WebKit 322
- widget 184
- Windows 40
- WinMerge 362
- workflow 32, 334
- Workflow (module) 335
- Wysiwyg (module) 151

X

- XCache 383
- XHTML 21, 368

Z

- Zen (thème) 263

Pour que **l'informatique**
soit **un outil**
et non *un ennemi* !

Concevoir et déployer ses sites web avec **Drupal**

La puissance et la modularité pour des sites web plus riches

L'outil libre de publication web Drupal dans sa version 6 et 7 à venir est exemplaire par son architecture modulaire. Il continue de s'imposer aux créateurs de contenu qui doivent construire des sites puissants et modernes – du simple blog au site professionnel étendu. Il permet de gérer des flux de contributions complexes, offre une grande diversité de types de contenus et une interface d'administration efficace.

Yoran Brault est développeur, architecte et expert technique depuis plus de dix ans (notamment dans les techniques Java/J2EE) et développeur Drupal depuis plus de 5 ans. Il est l'auteur et le mainteneur des modules Drupal webDAV et FileBrowser, et contribue de façon régulière au forum Drupalfr.org. Créateur du site Artisan numérique en 2004, il a fondé sa société de services **arNumeral** en 2009 et a participé à la refonte du site **Mediapart.fr** sous Drupal 6.

Exploitez la puissance de Drupal pour construire vos sites web !

- Comprenez l'architecture de Drupal et réfléchissez... à quoi faire et ne pas faire avec un CMS aussi puissant
- Configurez votre environnement de développement (WAMP/LAMP)
- Administrez Drupal et ses modules, architecturez vos pages et proposez plusieurs langues
- Gérez vos utilisateurs, les rôles des contributeurs et flux de publication en toute sécurité (OpenID...)
- Enrichissez les types de contenus : articles, commentaires, discussions, FAQ, formulaires et champs, fils RSS, filtres et formats...
- Modifiez l'apparence avec les thèmes et personnalisez les modèles
- Optimisez la charge d'administration par l'automatisation des tâches récurrentes
- Créez votre propre module et évitez les écueils de la mise en production
- Évaluez les améliorations de Drupal 7

À qui s'adresse cet ouvrage ?

- *Développeurs web attirés par la puissance de Drupal*
- *Webmasters qui souhaitent être guidés dans la construction de leur site et le choix de leurs modules.*