

Introduction à **Drupal**

par Guillaume Boudrias,
Coop Laboratoires Praxis,
gboudrias@praxis.coop



version 1.0
Avril 2015

Table des matières

Présentation.....	3
À propos de ce document.....	3
Licence.....	3
Mises à jour.....	3
Feedback.....	3
Qu'est-ce que Drupal ?.....	4
Définition.....	4
Pré-requis.....	4
La communauté et les logiciels libres.....	4
Exemples de sites.....	4
Installation.....	5
Téléchargement.....	5
Créer la base de données.....	5
Créer le fichier de paramètres.....	5
Rouler le script d'installation.....	5
Configuration après installation.....	5
Principes généraux.....	6
Contenu.....	6
Node.....	6
Types de contenu.....	6
Champs.....	6
Taxonomie.....	6
Entités.....	7
Blocs.....	7
Menus.....	7
Rôles et utilisateurs.....	7
Utilisateurs.....	7
Rôles et permissions.....	8
Modules et thèmes.....	8
Modules essentiels.....	8
Views.....	8
Nouvelle vue.....	8
Interface d'édition.....	9
Rules.....	9
Nouvelle règle.....	9
Modifier une règle.....	9
Panels.....	10
Nouvelle page Panels.....	10
Webform.....	10
Nouveau formulaire.....	10
Modifier formulaire.....	10
Autres modules « contrib ».....	11
Modules de développement.....	11
Recommandations générales.....	12
Mises à jour et sécurité.....	12
Pratiques recommandées.....	12

Présentation

À propos de ce document

Ce document d'introduction à Drupal a été écrit pour compléter la formation Drupal organisée par Louis-Philippe Véronneau dans le cadre des ateliers de logiciel libre de la [SOGÉÉCOM](#).

Ce document concerne la version 7 de Drupal (la version stable actuelle). Ceci dit, les principes généraux devraient s'appliquer au moins aux versions 6, 7 et 8.

Licence

Ce document, publié selon les termes de la licence [Attribution - Partage dans les Mêmes Conditions 4.0 International \(CC BY-SA 4.0\)](#), peut être lu et copié librement.

Mises à jour

Une version à jour de ce document sera toujours disponible à l'adresse suivante:
<http://libre.sogecom.org/guides>

Feedback

Les commentaires sont les bienvenus à gboudrias@praxis.coop.

Qu'est-ce que Drupal ?

Définition

Drupal est un système de gestion de contenu gratuit, libre et à source ouverte. Il se compare aux autres systèmes de gestion de contenu tels que WordPress et Joomla. En n'utilisant que les fonctionnalités de base, Drupal permet de gérer un site web au contenu dynamique et changeant, avec plusieurs utilisateurs et des fonctionnalités normales telles que des commentaires et des bulletins d'information.

Drupal est aussi un framework de développement complexe, qui se compare à d'autres frameworks tels que Ruby on Rails et Django. Il permet de développer rapidement des applications web.

Pré-requis

Drupal est développé en PHP, et requiert donc un logiciel de serveur web PHP tel que Apache. La version stable la plus récente de Drupal est Drupal 7.

La configuration minimale recommandée pour Drupal 7 est la suivante :

- 60 mégaoctets d'espace disque
- Apache2
- MySQL 5.0.15 ou plus
- PHP 5.4 ou plus

Voir <https://www.drupal.org/requirements> pour les détails des configurations requises.

La communauté et les logiciels libres

Drupal est un logiciel libre, signifiant que le code source est disponible gratuitement, et que les modifications sont permises et encouragées par la communauté.

Le site officiel de Drupal est <http://www.drupal.org>. On y retrouve la documentation, des forums et surtout des milliers de modules (plugins) disponibles gratuitement, et maintenus par la communauté.

La communauté Drupal adhère à des valeurs communes à plusieurs projets de logiciel libre, incluant la flexibilité, le travail d'équipe et la modularité. Drupal et tous les modules sur drupal.org sont distribués sous licence libre GPL ([Gnu Public Licence](#)), qui encourage le partage de l'innovation et des améliorations avec la communauté.

Exemples de sites

- Le site de la communauté : <http://www.drupal.org>
- Site de pétitions de la maison blanche : <https://petitions.whitehouse.gov/>
- Oxfam: <https://www.oxfam.org>
- Site de e-commerce : <https://www.subcitizen.com/>
- Praxis (site de l'auteur): <http://praxis.coop>

Voir <https://www.drupal.org/case-studies> pour une liste exhaustive d'exemples.

Installation

Cette section est un résumé (et une traduction) du guide d'installation de la communauté:
<https://www.drupal.org/documentation/install>

Vous aurez besoin d'un serveur web sur lequel installer Drupal. Vous pouvez trouver un hébergeur sur <https://www.drupal.org/hosting>. Vous pouvez installer Drupal sur n'importe quelle machine avec les configurations minimales. Pour les tests locaux, je recommande d'installer Linux sur VirtualBox : <https://www.virtualbox.org/>

Téléchargement

Télécharger la version la plus récente: <https://www.drupal.org/project/drupal>

Extraire l'archive.

Créer la base de données

Vous aurez besoin d'une base de données MySQL dans laquelle Drupal conservera les informations.

La méthode pour créer une base de données varie selon l'hébergeur. Il y aura généralement une interface PHPMyAdmin.

Créer le fichier de paramètres

Copier le fichier sites/default/default.settings.php vers sites/default/settings.php.

Changer les permissions du nouveau fichier pour permettre l'exécution.

Rouler le script d'installation

Visiter le fichier install.php à partir de votre navigateur. Par exemple, si votre nom de domaine est exemple.com, visiter <http://exemple.com/install.php>.

Suivre les étapes affichées. Le script d'installation vous demandera les informations de connexion à la base de données, ainsi que des informations de base telles que le nom du site.

En cas de doute, utiliser le profil d'installation « Standard ».

Configuration après installation

Ces étapes ne sont pas obligatoires, mais faciliteront la gestion.

Configurer les tâches de routine cron : <https://www.drupal.org/cron>

Configuration clean-url : <https://www.drupal.org/getting-started/clean-urls>

Principes généraux

Contenu

Drupal est un système de gestion de contenu, donc la majorité des concepts sont liés à la structure du contenu.

Node

Un node (parfois appelé « noeud » en français) est l'unité de contenu de base de Drupal. Un node contient un titre, un « corps » (body, le texte) et des options de publications.

La façon la plus simple de créer une page à afficher est de créer un node. Les nodes contiennent des champs, au minimum le champ Titre et le champ Body (texte), parce qu'il faut un nom au contenu et quelque chose à afficher.

Pour créer un node, aller dans Contenu → Ajouter du contenu. Sélectionner le type de contenu approprié.

Types de contenu

Drupal permet de séparer le contenu en types pour en faciliter la gestion. Par exemple, on utilise souvent le type « Basic Page » pour le contenu dit « statique » (qui change peu) et les pages de documentation, et le type « Article » pour les articles plus courts et fréquents.

On peut en somme avoir différents champs pour différents types de contenu. Par exemple, on peut avoir un type de contenu « blogue » avec un champs « image » et des champs « références ».

Le profil d'installation standard contient deux types de contenu : Article et Basic Page.

Voir <https://www.drupal.org/node/21947> pour les différences entre les types de contenu disponibles par défaut.

On peut créer autant de types de contenu qu'on veut : Aller dans Structure → Type de contenu → Ajouter un type de contenu.

Champs

Un champ (« field ») contient un type de données spécifique et limité. Par exemple, on peut avoir un champ « quantité », qui ne peut contenir que des données numériques.

Drupal contient plusieurs types de champs par défaut. Les plus utilisés sont : Texte, image, référence à un terme.

L'utilité d'avoir plusieurs sites de contenu vient du fait que chaque type de contenu peut avoir une combinaison de champs différentes.

Pour ajouter un champ sur un type de contenu, aller dans Structure → Type de contenu → Gérer les champs.

Chaque champ peut être affiché plusieurs fois.

Taxonomie

La taxonomie permet de catégoriser le contenu.

Un terme de taxonomie peut être vu comme une catégorie ou un « tag ». Les termes sont organisés en hiérarchie dans un vocabulaire. Le vocabulaire « tag » est inclus par défaut.

Pour voir la liste des vocabulaires, aller dans Structure → Taxonomie.

Pour ajouter un vocabulaire, aller dans Structure → Taxonomie → Ajouter un vocabulaire

Pour ajouter un terme, aller dans Structure → Taxonomie → [Nom du vocabulaire] → Ajouter un terme

Entités

Tous les nodes, champs et termes de taxonomie sont des entités.

Le concept d'entité est abstrait et désigne généralement tout ce qui est du contenu entré par l'utilisateur.

Ceci dit, tous les types d'entités peuvent contenir des champs. Dans le cas des termes de taxonomie, les champs peuvent être utilisés pour enregistrer des méta-informations sur le terme. Par exemple, on peut avoir une image associée à un terme, ou un lien qui s'affiche en certaines circonstances.

La différence entre les nodes et les autres entités est que les nodes contiennent des informations de publication.

Les modules peuvent aussi créer des nouveaux types d'entités. Par exemple, les utilisateurs et les commentaires sont aussi des entités.

Blocs

Drupal gère ce qui est affiché sur une page avec des « blocs ». L'affichage des blocs peut être personnalisé selon la page ou l'utilisateur (et d'autres conditions). Un bloc peut être un « widget » ou du contenu personnalisé.

Les régions disponibles pour l'affichage des blocs varient selon le thème, mais sont généralement séparées en sections visuelles telles que « sidebar first », « header », « content », etc.

Pour voir et gérer les blocs disponibles, aller dans Structure → Blocs

On peut ajouter un bloc manuellement pour afficher du contenu spécial dans certaines conditions, par exemple pour afficher un avertissement sur certaines pages. Pour ajouter un bloc, aller dans Structure → Blocs → Ajouter un bloc.

Menus

Pour gérer les menus, aller dans Structure → Menus.

Le menu principal sera généralement affiché dans le haut du site. L'utilisation des menus varie selon le thème et peut être configuré dans Structure → Menus → Paramètres.

Rôles et utilisateurs

Utilisateurs

Pour ajouter un utilisateur, aller dans Personnes → Ajouter un utilisateur.

Les utilisateurs étant des entités, on peut leur ajouter des champs (ce qui constitue les « profils » des utilisateurs). Pour gérer les champs de profils, aller dans Configuration → Personnes → Paramètres de compte → Gérer les champs.

Rôles et permissions

Drupal permet de personnaliser les permissions des utilisateurs. Ces permissions sont organisées par rôles (essentiellement une catégorie d'utilisateur). Pour ajouter une permission à un utilisateur, il faut donc ajouter la permission à un des rôles qui lui est attribué, ou appliquer un nouveau rôle à l'utilisateur.

- Pour créer un rôle, aller dans Personnes → Droits → Rôles.
- Pour ajouter un rôle à un utilisateur, modifier l'utilisateur (cliquer sur Éditer sur la page de l'utilisateur).
- Pour gérer les permissions, aller dans Personnes → Droits.
- Pour voir la liste des utilisateurs, aller dans Personnes.

Modules et thèmes

Les fonctionnalités Drupal sont séparées par modules. Un module peut ajouter des types de contenus, interagir avec d'autres modules ou changer la façon dont Drupal fonctionne.

Pour voir la liste des modules et activer ou désactiver des modules, aller dans Modules.

Les thèmes sont comme des modules qui gèrent principalement l'aspect visuel du site.

Pour voir la liste des thèmes, aller dans Apparence.

Drupal.org contient des milliers de modules et thèmes qui permettent d'ajouter des fonctionnalités communes :

- Trouver un module: https://www.drupal.org/project/project_module
- Trouver un thème : https://www.drupal.org/project/project_theme

Modules essentiels

Views

<https://www.drupal.org/project/views>

Views permet d'afficher des listes de contenu (en français, des « vues »). Ces listes peuvent être affichées de plusieurs manières, telles qu'en liste à puces ou en tableau. Les vues peuvent afficher un nombre arbitraire d'élément avec des conditions arbitraires. Par exemple, sur une vue de page d'accueil on choisit souvent d'afficher seulement le contenu le plus récents de type « article ».

- Pour utiliser Views, activer Views et Views UI (le module d'interface) dans la liste des modules.
- Pour voir la liste des vues, aller dans Structure → Views

Plusieurs vues communes sont disponibles par défaut.

Nouvelle vue

Pour ajouter une nouvelle vue, cliquer sur « Ajouter une nouvelle vue ».

La liste de sélection « Montrer [Contenu] » détermine le type de vue qui sera créée. Le type est presque toujours Contenu (content), parce qu'on veut presque toujours afficher des champs de contenu. Les vues de type Utilisateur peuvent parfois être utiles pour la gestion.

Les modules peuvent ajouter des types de vues.

Pour les vues de contenu, on peut sélectionner le type de contenu à afficher.

Le format d'affichage détermine la façon dont les données seront affichées. La liste HTML crée une simple liste, le tableau crée un tableau HTML. La liste non mise en forme et la grille sont utiles pour personnaliser l'afficher avec du CSS.

Il y a plusieurs façons d'afficher les champs :

- **Résumés** : Affiche le format « résumé » des nodes, qu'on peut modifier dans la page d'édition de chaque node.
- **Publications complètes** : Affiche les nodes complets, tels qu'ils s'affichent sur la page du node.
- **Titres** : Affiche seulement les titres des nodes.
- **Champs** : Affiche des champs sélectionnés manuellement. C'est le type utilisé le plus fréquemment car il permet de choisir l'information à afficher de façon granulaire.

Interface d'édition

Format : Changer le format et les paramètres d'affichage.

Champs : Ajouter et modifier des champs.

Filter criteria (critères de filtre) : Ajouter ou modifier des filtres.

Sort criteria : Gère l'ordre dans lequel les nodes sont affichés.

Entête et pied de page : Permet d'ajouter du texte avant ou après les nodes

Filtres contextuels : Filtres basés sur des éléments changeants tels que l'URL

Relations : Permet de charger des éléments liés aux éléments sélectionnés

Formulaire exposé : Gère les options du formulaire lorsque les filtres sont exposés

Autre : Options avancées

Rules

<https://www.drupal.org/project/rules>

Permet d'effectuer des actions arbitraires lorsque certains événements se produisent. Souvent utilisé pour envoyer des courriels lors de création de contenu ou d'autres formes de modération. Les modules contrib contiennent parfois des règles pour faciliter la gestion.

Pour utiliser Rules, activer Rules et Rules UI (module d'interface) dans la liste des modules.

Pour voir la liste des règles, aller dans Configuration → Processus → Rules

Nouvelle règle

Cliquer sur « Ajouter une nouvelle règle ». Choisir un nom. Les étiquettes ne sont pas importantes.

Choisir un événement auquel réagir. Les événements disponibles dépendent des modules activés.

Modifier une règle

Events : Le ou les événements qui déclenche les actions.

Conditions : Les conditions de déclenchement. Les conditions disponibles dépendent des événements sélectionnés.

Actions : Ce qui arrive quand l'événement est déclenché.

Panels

<https://www.drupal.org/project/panels>

Panels permet de personnaliser les régions d'affichage sans avoir à modifier le code.

Pour utiliser Panels, activer Panels et Page manager dans la liste de modules. Page manager n'est pas automatiquement requis mais permet d'avoir des pages personnalisables.

Le module Panels In-Place Editor facilite grandement la gestion des blocs.

Nouvelle page Panels

Pour créer une nouvelle page, aller dans Structure → Pages → Add custom page. Remplir les informations et choisir le Type de variante « Panneau ». Cliquer sur Continuer.

Choisir une mise en page, soit une mise en page pré-faite telle que Colonnes, ou le constructeur Flexible pour pouvoir personnaliser la mise en page. Cliquer sur Continuer.

Les paramètres par défaut sont généralement appropriés, mais sélectionner l'option « Éditeur intégré » pour le moteur de rendu pour utiliser le In-Place Editor. Cliquer sur Continuer.

Pour ajouter du contenu, cliquer sur le rouage d'une des régions.

Les éléments qui peuvent être ajoutés dépendent des modules activés. Pour pouvoir ajouter des views, il faut activer le module « Views content panes ».

La façon la plus simple d'ajouter du contenu est l'option « Nouveau contenu personnalisé », mais il vaut mieux créer un node séparément et utiliser l'option « Noeud existant ». La différence est que le contenu personnalisé n'est pas un node, et donc ne s'intègre pas dans un workflow normal (il n'est pas disponible dans le menu d'administration « Contenu »).

Webform

<https://www.drupal.org/project/webform>

Webform permet de créer des formulaires facilement, qui seront remplis par les utilisateurs. Les webforms peuvent être utilisés pour des sondages, ou pour permettre aux gens de s'inscrire à un événement, ou encore pour permettre aux utilisateurs de vous contacter¹.

En somme, les webform couvrent les cas classiques où un formulaire HTML est utilisé, sans avoir à modifier le code. Pour utiliser Webform, activer le module Webform dans la liste de modules.

Pour voir la liste des formulaires, aller dans Contenu → Webforms

Nouveau formulaire

Pour créer un nouveau formulaire, aller dans Contenu → Ajouter du contenu → Webform

Le body dans ce cas-ci représente le texte qui sera affiché avant les éléments du formulaire.

Modifier formulaire

Aller dans l'onglet Webform (l'onglet Modifier contient tout ce qui est « node », c'est à dire les options de publication etc).

Le sous-onglet Form components contient les composants du formulaire.

L'onglet courriel contient les configurations liées aux courriels qui sont envoyés quand on soumet le formulaire.

L'onglet Form settings contient les configurations liées aux actions qui arrivent quand on soumet le formulaire (incluant surtout le message de confirmation).

¹ Le module Contact (inclus dans le core) crée des formulaires de contact automatiquement, pour le site et chaque utilisateur.

Autres modules « contrib »

Plusieurs modules sont très populaires dans la communauté :

- [Administration menu](#) : Rend le menu d'administration beaucoup plus utile.
- [Content Access](#) : Permet de rendre les permissions d'accès au contenu plus granulaires.
- [Commerce](#) : Permet de gérer le e-commerce.
- [Date](#) : Permet d'avoir des champs dates avec des « popups » de calendriers.
- [Login Destination](#) : Permet de rediriger les utilisateurs à une page différente lorsqu'ils se connectent
- [Module Filter](#) : Simplifie grandement la page d'administration des modules
- [Mollom](#) : S'intègre avec le service web Mollom (comme Captcha mais mieux)
- [Node Clone](#) : Permet de copier des nodes facilement
- [Pathauto](#) : Permet de définir automatiquement les alias d'URL (pour ne pas avoir node/chiffre pour tous les chemins de node par défaut).
- [Piwik](#) : S'intègre au logiciel libre de statistiques web Piwik.
- [References](#) : Permet aux nodes de faire référence à d'autres nodes.
- [Simplenews](#) : Permet de gérer des bulletins d'informations (newsletters).
- [XML Sitemap](#) : Permet aux engins de recherche de mieux indexer le site grâce au « sitemap » XML.
- Les modules de champs additionnels tels que [Email](#), [Link](#), etc.

Modules de développement

Plusieurs modules peuvent être utiles dans le cadre du développement d'un nouveau site :

[Drush](#) : Outil de ligne de commande Drupal. Permet de gérer Drupal plus efficacement à travers la ligne de commande.

[Backup and Migrate](#) : Permet de faire des copies de sauvegarde du site.

[Coder](#) : Permet de vérifier que le code de notre module respecte les standards de la communauté.

[Devel](#) : Permet de rouler du PHP dans l'interface, ainsi que de générer du contenu de tests en vrac.

[Devel debug log](#) : Permet de stocker des messages du style devel au lieu de les afficher immédiatement. Utile quand les actions arrivent durant le cron.

[Features](#) : Permet d'activer des Features, qui sont comme des paquets de modules.

[Features Tools](#) : Facilite de développement de nouveaux Features.

[Feeds](#) : Permet d'importer du contenu d'ailleurs automatiquement, par exemple on peut importer le contenu de flux RSS ou même d'une boîte de courriels.

[Theme developer](#) : Facilite le développement de nouveaux thèmes.

Les thèmes [Zen](#) et [Omega](#) ne sont pas des modules, mais des thèmes qui facilitent grandement la création de nouveaux thèmes (appelés sous-thèmes).

Recommandations générales

Mises à jour et sécurité

Il est important de garder les modules et le core Drupal à jour.

Le module Update manager est activé par défaut, et envoie des courriels lorsque les modules ne sont pas à jour.

En plus des améliorations fonctionnelles, les mises à jour sont importantes d'un point de vue de sécurité. Étant donné qu'il y a des milliers de modules Drupal et que ceux-ci sont développés par la communauté, il arrive qu'on découvre des vulnérabilités dans certains modules ou dans le core. Il faut alors mettre à jour les modules concernés et/ou le core pour avoir la nouvelle version sécuritaire.

Avoir des modules avec des failles de vulnérabilité augmente les chances que le site soit infiltré par un tiers avec des intentions malicieuses. Même si on ne croit pas détenir d'informations précieuses, un site compromis peut être impliqué dans des utilisations douteuses ou illégales telles que l'envoi massif de pourriels. De plus, tous les utilisateurs ont un courriel associé, et ceux-ci peuvent être revendus sur le marché noir.

Finalement, il n'est pas nécessaire d'avoir un site très connu pour être en « danger » d'infiltration. Certains hackers ont des « robots » qui naviguent le web automatiquement pour trouver les sites vulnérables. Plus longtemps on garde des versions non-sécuritaires, plus les chances sont grandes qu'un robot détecte et infiltre automatiquement le site, habituellement pour publier ou envoyer du spam.

Si on maintient plusieurs sites, le logiciel d'hébergement libre [Aegir](#) facilite grandement la gestion des mises à jour. C'est un logiciel conçu pour créer et gérer des installations Drupal de manière simple et efficace.

Pratiques recommandées

Voir <https://www.drupal.org/best-practices> pour une liste exhaustive.

- **Garder ses modules à jour.** Pour des raisons d'utilisabilité autant que de sécurité.
- **Avoir des copies de sauvegardes** (« backups ») pour les fichiers et la base de données.
- **Ouvrir une requête quand on remarque un problème.** Il y a de bonnes chances que vous ne soyez pas le premier à avoir le problème.
- **Ne jamais utiliser le filtre de texte PHP** (ou presque), et en général ne jamais insérer du code dans la base de données. Ceci peut grandement compliquer la gestion à long terme.
- **Ne jamais patcher le core.** Utiliser une version patchée d'un modules core complique grandement la gestion à long terme. Il vaut toujours mieux de développer un nouveau module.
- Utiliser le moins de modules possible, c'est à dire s'en tenir au core lorsque possible. Ceci facilite grandement les mises à jour majeures (entre les versions 7 et 8 de Drupal par exemple).
- Utiliser un site de test si possible. On dit souvent « Ne jamais coder en production ». Il vaut mieux ne pas déployer des changements au fonctionnement du site avant de les avoir testé. Aegir facilite grandement ceci car il permet de « cloner » des sites entiers avec un seul bouton.
- Éviter le module Boost (cache HTML locale). Utiliser plutôt un logiciel de cache externe tel que [Varnish](#) ou [Nginx](#).