

NOUVEAU

Drupal™



Le Guide Drupal 8

Décryptage de la nouvelle version

Smile & Acquia™

OPEN SOURCE SOLUTIONS

📍 www.smile.fr

📍 www.drupal8-guide.com

☎ +33 (0)1 41 40 11 00

✉ contact@smile.fr

🐦 [twitter: @GroupeSmile](https://twitter.com/GroupeSmile)

📍 www.acquia.com/fr

☎ +33 (0)1 79 97 25 70

✉ sales@acquia.com

🐦 [twitter: @acquia_fr](https://twitter.com/acquia_fr)



PREAMBULE

CE LIVRE BLANC

Afin d'informer et préparer la communauté Francophone Drupal de la sortie de Drupal 8 en version stable, les sociétés partenaires Smile & Acquia se sont mobilisées pour livrer un guide compréhensif dédié à la dernière version du CMS haut de gamme.

Ce guide dévoile les raisons pour lesquelles Drupal 8 est la version la plus attendue et la plus surprenante. Vous pourrez y découvrir une analyse complète de Drupal 8, en parcourant des éléments à la fois fonctionnels et techniques.

Vous découvrirez, également, comment la solution a évolué par rapport à ses dernières versions et comment celle-ci répondra aux besoins de différents profils d'utilisateurs.

Très bonne lecture !

PRESENTATION DE SMILE

Smile est une **société d'ingénieurs experts** dans la mise en œuvre de **solutions open source** et l'intégration de systèmes appuyés sur l'open source. Smile est membre actif d'associations de promotion du logiciel libre telles que l'**APRIL**, l'association pour la promotion et la défense du logiciel libre, le **PLOSS** – le réseau des entreprises du Logiciel Libre en Ile-de-France.

Smile compte près de 800 collaborateurs dans le monde, dont plus de 650 en France (septembre 2015), ce qui en fait *le premier intégrateur français et européen de solutions open source*.

Depuis 2000 environ, **Smile mène une action active de veille technologique** qui lui permet de découvrir les produits les plus prometteurs de l'open source, de les qualifier et de les évaluer, de manière à proposer à ses clients les produits les plus aboutis, les plus robustes et les plus pérennes.

Cette démarche a donné lieu à **toute une gamme de livres blancs** couvrant différents domaines d'application. La gestion de contenus (2004), les portails (2005), la business intelligence (2006), la virtualisation (2007), la gestion électronique de documents (2008), les PGI/ERPs (2008), les VPN open source (2009), les Firewall et Contrôle de flux (2009), les Middleware orientés messages (2009), l'e-commerce et les Réseaux Sociaux d'Entreprise (2010), le Guide de l'open source et NoSQL (2011), Mobile et Recensement et audit (2012) et plus récemment Big Data et ERP open source pour l'e-commerce (2014).

Chacun de **ces ouvrages présente une sélection des meilleures solutions open source** dans le domaine considéré, leurs qualités respectives, ainsi que des retours d'expérience opérationnels.



Au fur et à mesure que des solutions open source solides gagnent de nouveaux domaines, Smile sera présent pour proposer à ses clients d'en bénéficier sans risque. Smile apparaît dans le paysage informatique français comme **le prestataire intégrateur de choix** pour **accompagner** les plus grandes entreprises dans l'adoption des meilleures solutions open source.

Ces dernières années, Smile a également étendu la gamme des services proposés. Depuis 2005, un département consulting accompagne nos clients, tant dans les phases d'avant-projet, en recherche de solutions, qu'en accompagnement de projet. Depuis 2000, Smile dispose d'un studio graphique, devenu en 2007 Smile Digital – agence interactive, proposant outre la création graphique, une expertise e-marketing, éditoriale, et interfaces riches. Smile dispose aussi d'une agence spécialisée dans la TMA (support et l'exploitation des applications) et d'un centre de formation complet, Smile Training.

Enfin, Smile est implanté à Paris, Lille, Lyon, Grenoble, Nantes, Bordeaux, Marseille et Montpellier. Et présent également en Suisse, au Benelux, en Ukraine, en Russie, au Maroc et en Côte d'Ivoire.

Smile et Drupal

Fin 2008, après de nombreux tests et analyses, nous avons décidé d'intégrer Drupal aux solutions de gestion de contenu que nous proposons chaque jour à nos clients. **Il s'en est suivi un plan stratégique de montée en compétence et d'investissements.**



A ce jour, **Smile dispose de plus de 150 collaborateurs formés et experts de Drupal** et ce nombre ne fait que grossir chaque jour. En effet de nombreux recrutements sont prévus sur les années qui viendront **pour confirmer notre position de premier intégrateur Drupal en France.**

Quelques références Drupal de Smile





PRESENTATION D'ACQUIA

Acquia a été fondé en 2007 par le créateur de Drupal, Dries Buytaert. Acquia a été créé à l'origine pour accompagner les projets Drupal à grande échelle, en termes d'hébergement et de services autour de celui-ci. Aujourd'hui, Acquia propose sous forme de souscriptions une suite de solutions logicielles, qui en plus d'accompagner la conception et l'élaboration des expériences digitales, sur tous les supports, offre des outils pour les optimiser et les personnaliser.

Ses clients se reposent donc sur un hébergement de premier ordre, sécurisés, scalables, optimisés pour Drupal qui répond à leurs besoins spécifiques, en terme de support (24/7, 365, avec un SLA de 99.95 sur les applications), de performance, d'expertise et d'accompagnement.

Acquia possède également une haute expertise dans la conduite de projets importants et connaît une croissance phénoménale depuis sa création. Parmi ses clients, citons des entreprises telles que Warner, J&J, Pfizer, NBC Universal, the Weather Channel (top 10 des sites les plus visités) et plus près de chez nous Veolia, Saint-Gobain, etc.

L'Acquia Platform permet aux marques de construire et de développer des expériences digitales personnalisées pour chaque visiteur quelque soit le support sur lequel il navigue.

- Digital Experience Cloud
 - Environnement managé à haute disponibilité pour des expériences digitales stratégiques qui doivent répondre aux normes de sécurité les plus exigeantes
 - Acquia Cloud Site Factory : solution pour le déploiement de sites multiples depuis un codebase unique simplifié
 - Acquia Cloud est la solution pour quelques uns des plus grands événements planétaires comme les Grammy.com et Emmys.com, et pour les sites de quelques-uns des artistes les plus populaires de NBC Universal et Warner Music Group
 - Acquia gère, monitore, et sécurise l'environnement du site du client.
- Digital Engagement Services:
 - Les meilleures expériences digitales sont personnelles. Elles sont propres au contexte de chaque individu. Mieux comprendre ses clients, booste l'engagement et la conversion.
 - Vous pouvez aussi construire des segments pertinents, du contenu automatiquement ciblé avec l'A/B testing, et l'implémentation d'expériences hyper-personnalisées
 - Proposer le meilleur de vos contenus et de vos produits pour chaque visiteur individuel. Mesurer et optimiser vos offres



Smile & Acquia

Pour accélérer notre développement et notre reconnaissance sur l'expertise Drupal, Smile a signé début 2010 un contrat de partenariat avec Acquia. Ce partenariat nous permet d'accéder à de nombreuses ressources de qualité et d'être au plus près de l'évolution du produit.

Smile est à l'heure actuelle l'un des rares partenaires Acquia à avoir atteint le niveau Select (le plus haut niveau) en France.

WWW.SMILE.FR



SOMMAIRE

WWW.SMILE.FR

PREAMBULE..... 2

CE LIVRE BLANC.....2

PRESENTATION DE SMILE2

PRESENTATION D’ACQUIA.....4

SOMMAIRE 6

INTRODUCTION A DRUPAL 8..... 7

LES GARANTIES DE DRUPAL.....7

L’ARCHITECTURE DRUPAL.....8

DRUPAL 8 ET LES DIFFERENTS PROFILS8

LE GUIDE DRUPAL 8 D’ANGELA BYRON 16

AMELIORATIONS POUR MOBILES.....19

MULTILINGUE++.....21

LES CONCEPTEURS DE SITE À L’HONNEUR.....27

LES ASPECTS TECHNIQUES POUR LES DEVELOPPEURS 33

AMELIORATIONS POUR LE DEVELOPPEUR FRONTEND.....33

AMELIORATIONS POUR LES DEVELOPPEURS BACK-END43

DES AMELIORATIONS EN PROFONDEUR DÈS LA VERSION CORE50

LA POO A PORTEE DE MAIN51

FOIRE AUX QUESTIONS..... 61

REMERCIEMENTS66



INTRODUCTION A DRUPAL 8

LES GARANTIES DE DRUPAL

Drupal offre une polyvalence et une modularité rarement égalée. Il dispose d'une des communautés d'utilisateurs les plus actives, avec plus de 1 million de membres. La nouvelle version de la solution propose une multitude de nouveautés permettant une expérience utilisateurs plus intuitive et la construction de plateformes web flexibles et puissantes. Avec ces nouvelles propositions et améliorations, Drupal 8 est toujours une solution :

Communautaire

Drupal se base sur un réseau mondial de plus d'1 million d'utilisateurs dont 31 000 développeurs dans plus de 230 pays. Cette communauté donne vie à des milliers de modules au quotidien ; chacun ajoutant de nouvelles fonctionnalités répondant aux principaux enjeux des utilisateurs. Fort de ce réseau, Drupal couvre, jour après jour, un périmètre fonctionnel de plus en plus large.

Flexible

L'ensemble des modules mis à disposition par la communauté (forum, galerie photos, sondage, formulaire, newsletter, messagerie, chat, enquête, paiement en ligne, calendrier partagé, etc.) assure à Drupal une flexibilité importante. Avec Drupal, Il existe forcément une extension qui répond à votre besoin.

Leader

Twitter, The Economist, Mercedes Benz, Warner Music Group, Total, Le Bon Marché et Radio France font partie de ces acteurs majeurs qui font confiance à Drupal pour leurs sites. Drupal a déjà conquis aujourd'hui les plus grands groupes et est considéré comme un des outils les plus complets.

Sécurisé

La sécurité est un aspect fondamental de Drupal. Il est soumis aux tests de sécurité les plus poussés, à la fois par les membres de la communauté et par des experts en sécurité à travers le monde. Les plus hautes organisations politiques nationales et internationales (l'Etat français, la Maison Blanche, le gouvernement Australien, etc.) et les plus grands groupes internationaux font confiance à Drupal pour gérer leurs informations les plus sensibles.



L'ARCHITECTURE DRUPAL

Le tableau ci-dessous détaille les principales différences d'architecture entre Drupal 7 et Drupal 8 :

Environnement	Drupal 7	Drupal 8
Plateformes matérielles (serveurs) utilisables	Apache, Nginx 1.0, Hiawatha ou Microsoft IIS 7	Apache 2.0 et + (recommandée) Nginx 1.1 et +
Bases de données utilisables	MySQL 5.0.15 ou +, PostgreSQL 8.3 ou +, SQLite 3.3.7 ou +. Possibilité : SQL Server et Oracle avec des modules supplémentaires mais fortement déconseillé	MySQL 5.0.15 ou + MariaDB 5.1.44 ou + Percona 5.1.70 ou + PostgreSQL 8.3 et + avec PDO, SQLite 3.4.2 et +
Langages de développement	PHP 5.3 ou +	PHP 5.5.9 et +
Navigateurs	IE8 et + Firefox 5.x et + Opera 12 et + Safari 5.x et + Google Chrome	IE9 et + Firefox 5.x et + Opera 12 et + Safari 5.x et + Google Chrome

WWW.SMILE.FR

DRUPAL 8 ET LES DIFFERENTS PROFILS

Depuis leur création il y a 20 ans déjà, les sites internet n'ont cessé d'évoluer.

Au milieu des années 90, le lancement d'un site internet était confié aux services informatiques. C'était encore des projets un peu marginaux, surtout en France : la mode était encore au minitel, plus ancré aux usages de l'époque et l'équipement nécessaire pour se connecter à Internet était conséquent.

Jusqu'à la fin des années 90, Internet était « boudé », par une grande partie des professionnels qui préféraient investir plus massivement dans des médias plus « traditionnels » comme la télévision, la radio ou encore la presse.

Avec les évolutions techniques, la concurrence accrue des FAI (fournisseurs d'accès) et des revendeurs d'ordinateur, internet s'est démocratisé au sein des familles, relayant petit à petit le minitel à l'état d'objet de collection.



Au début des années 2000, Internet est devenu une « arme de communication massive » pour les entreprises. La communication est désormais immédiate, internationale, globale... plus difficilement maîtrisable aussi. Elle revêt en tout cas, un aspect stratégique qui a métamorphosé les directions de la communication, les amenant à prendre de plus en plus la main sur la création des projets web menés jusqu'à présent par des équipes essentiellement techniques.

Depuis, d'autres profils s'y sont greffés comme les profils « e-marketing » qui ont révolutionné l'approche éditoriale des sites « institutionnels » en appliquant notamment des préceptes issus du monde de l'e-commerce.

Aujourd'hui, construire et administrer un site est devenu un travail collectif, mené par différents profils qui ont des compétences spécifiques mais complémentaires. L'évolution des comportements sur la manière de concevoir un site a aussi façonné l'approche d'un grand nombre de solutions : ce n'est plus l'humain qui s'accommode à l'outil mais l'outil qui s'adapte (au mieux) aux besoins et enjeux de chacun via des fonctionnalités propres à chacun, en supprimant notamment tout le « bruit environnant ».

Drupal 8 pour les développeurs

Le rôle du développeur ne se cantonne pas uniquement à écrire du code et à connaître les API de Drupal. Sur ce CMS, la connaissance des différents modules à installer est nécessaire pour simplifier les développements, limiter le code custom et favoriser la maintenance du site.

Le développeur est aussi confronté à des problématiques de charges, d'optimisation continue du code – y compris sur des solutions en voie de disparition – et doit produire, dans un temps souvent très limité, des sites robustes qu'il faut ensuite déployer, tout aussi rapidement, en production.

Comment Drupal s'adresse aux développeurs ?

En tirant profit de technologies éprouvées et « prêtes à l'emploi », Drupal 8 permet aux développeurs de faire plus de choses en moins d'étapes.

Avec son nouveau système de configuration basé sur des fichiers (fini les configurations en base de données !), ses capacités en terme d'interopérabilité (webservice / REST), ou tout simplement par le fait qu'il utilise le framework Symfony 2, Drupal 8 fait usage des meilleures technologies et techniques disponibles aujourd'hui en utilisant notamment des bibliothèques comme Composer, PHPUnit, Guzzle ou Zend Feed Component, etc.

Ainsi, le développement d'un site en Drupal 8, peut plus facilement être confié à des profils qui ne sont pas forcément familiers au développement de Drupal 7. Même si un vernis « Drupal » reste nécessaire, un développeur PHP qui connaît Symfony2 saura se débrouiller sans trop de difficultés.



Drupal dispose déjà d'une très grande communauté à laquelle se rajoute désormais celle de Symfony2. C'est un mariage de raison, pour le meilleur... rien que le meilleur. L'arrivée du Framework va « normaliser » les développements en adoptant des standards et des bonnes pratiques de mises en œuvre. La programmation procédurale a laissé place à la programmation orientée objet (POO) permettant notamment une plus grande flexibilité et modularité de Drupal. Cette normalisation passe aussi par des fonctionnalités Drupal qui se retrouvent (enfin !) dans le noyau comme le multilinguisme ou encore (et surtout !) « Views ».

Drupal 8 permet aux développeurs de pouvoir transporter une configuration complète d'un environnement à un autre (typiquement d'un environnement de recette vers un environnement de production) ou encore de sérialiser des données exploitables par la suite en JSON ou en XML sans compétences avancées requises.

Chez certains intégrateurs, le développeur peut aussi avoir le rôle de « thèmeurs ». Il s'occupe plus spécifiquement de la construction du « thème Drupal », c'est-à-dire un ensemble de gabarits. Ce montage technique est en PHP et inclus aussi la mise en place des feuilles de styles CSS spécifiques (pour gérer les polices, leurs corps, leurs graisses et leurs couleurs).

Sur certaines solutions, ce montage requiert un langage spécifique propre à la solution CMS, mais ce n'est pas le cas sur Drupal 8. L'une des nouveautés de cette nouvelle version concerne l'arrivée de Twig, le moteur de template de Symfony qui remplace PHPTemplate. Ce moteur est utilisé sur de nombreuses autres solutions du marché et n'est pas spécifique à Drupal. La syntaxe y est plus condensée, plus propre et plus adaptée au diagnostic d'erreurs.

Drupal 8 pour les website builders

Les website builders sont des profils un peu atypiques. Ils connaissent bien les rouages de Drupal et ont une vision à la fois technique et fonctionnelle de la solution. Ils ont notamment vocation à faire rapidement évoluer le site en s'appuyant le plus possible soit sur le fonctionnement natif de la solution, soit sur des modules existants.

Ils peuvent aussi vérifier la faisabilité technique d'une demande, montrer l'implémentation d'une fonctionnalité dans Drupal aux équipes - essentiellement des profils MOA – qui pourront par la suite l'expérimenter et en valider le fonctionnement.

Comment Drupal s'adresse aux « website-builders » ?

C'est un fait, Drupal 8 est nativement plus riche que Drupal 7 : Views, CkEditor (un éditeur de texte qui permet de générer du HTML), le multilinguisme (pour contribuer du contenu dans différentes langues)... ou encore Entity reference (pour facilement lier des contenus, des utilisateurs, de la taxonomie... entre eux) y sont directement intégrés et permettent déjà un grand nombre de choses.

Drupal 8 permet aussi aux « website-builders » de personnaliser facilement la page d'accueil par des blocs repositionnables sur lesquels on peut désormais définir des champs. Il est par ailleurs



possible de les traduire plus facilement dans d'autres langues, de jouer sur la visibilité en fonction de critères (langue, l'IP, l'hôte, l'URL...) ou encore de les dupliquer pour une réutilisation suivant des contextes d'affichages différents, y compris sur la même page.

Le website builders a désormais à sa disposition de nouveaux types de champs comme « date », « e-mail », « téléphone » pour alimenter les différents types de contenus qu'il peut construire via « Field UI » (pour les contenus) ou « Webform » (pour les formulaires). Concrètement, lorsque ces champs sont à remplir via un smartphone ou une tablette, l'affichage est automatiquement optimisé. Par exemple, un clavier numérique sur le champ « téléphone », un menu déroulant de choix de dates sur le champ « date », le symbole « @ » sur le champ e-mail... C'est peut-être un détail, mais cela est bien pratique à l'usage.

Drupal 8 dispose d'une nouvelle version du thème « Bartik » que les utilisateurs de Drupal 7 connaissent bien. Sa principale nouveauté : il offre un thème 100% Responsive, opérationnel immédiatement, sans manipulation supplémentaire.

Mais la principale innovation concerne surtout le thème Back-office. Il est lui aussi 100% Responsive et permet une administration du site, quel que soit l'appareil utilisé : desktop, smartphone ou tablette.

En conclusion, Drupal 8 repousse les obstacles de la construction de site et permet aux website builders d'avoir plus de liberté pour personnaliser leur site de différentes manières et à différents endroits.

Drupal 8 pour les DSI

La DSI a des objectifs différents de ceux des profils « développeurs ». Ils doivent s'assurer du potentiel de la solution à pouvoir répondre à différentes problématiques.

Il est tout d'abord question de savoir si l'entreprise dispose des compétences en interne pour d'une part, mener à bien ce projet et, d'autre part, pouvoir la maintenir. Cela soulève bien entendu des questions sous-jacentes : faut-il faire monter en compétence des personnes sur la technologie ou, au contraire, faire externaliser la mise en place de la solution ?

Se pose aussi des questions plus techniques, notamment sur la maturité et la durabilité du produit : Est-ce que la solution est suivie par une communauté qui lui assure une pérennité pour encore de nombreuses années ou est-ce un produit en fin de vie ? La solution est-elle facilement extensible à d'autres fonctionnalités et si oui, est-ce simple ? Il y a-t-il un suivi réactif concernant les failles de sécurité ?

Enfin, concernant l'aspect budgétaire, est-ce que la solution est soumise à des frais de licence et de support ? Est-ce un investissement unitaire ou annuel ?



En conclusion, les DSI doivent s'assurer qu'ils utilisent le bon outil technique, les bonnes compétences et ressources en interne ou externe pour minimiser le risque, y compris le risque technique d'un produit qui ne serait pas assez mûre.

Comment Drupal s'adresse aux « DSI »

Rappelons tout d'abord que Drupal est téléchargeable gratuitement et ne nécessite aucunement l'achat de licence ou l'obligation d'avoir recours à du support payant. Le produit est 100% Open Source et permet un libre accès au code source, de pouvoir le modifier librement et le reverser sans passer par un quelconque contrat vous liant avec un éventuel éditeur.

Drupal dispose d'une communauté très importante (plus d'un million de personnes), constituée de profils hétérogènes. On y retrouve notamment des « développeurs » ou des « mainteneurs » de code, travaillant pour Acquia, « l'éditeur » de Drupal ou complètement bénévoles. La communauté est réputée pour être très proactive sur :

- La résolution des failles de sécurité (sur le noyau et sur les modules a minima les plus connus).
- La mise en place sous forme de modules de nouvelles fonctionnalités pour couvrir tout type de besoins qu'ils soient nécessaires, émergents ou « à la mode ».

L'une des grandes forces de Drupal, c'est son extensibilité. Il y a une grande chance qu'une fonctionnalité qui doit être mise en place existe déjà grâce aux multitudes de modules Drupal. Cela est un aspect qui réduit ainsi les coûts de développement mais aussi de maintenance.

La capacité du CMS à pouvoir répondre à différentes problématiques (site institutionnel, e-commerce, intranet, RSE...) via des modules ou des distributions spécifiques ne remet pas en cause son approche technique qui reste la même. Il n'est donc pas obligatoire d'avoir des compétences spécifiques autres que de connaître un minimum Drupal pour pouvoir répondre à ces problématiques.

Drupal 8 est nativement « Responsive » et sait afficher vos différents contenus sur n'importe quel desktop, smartphone ou tablette. Bien sûr, il est nécessaire d'adapter le thème prévu par défaut à votre propre ergonomie et graphisme mais les fondamentaux y sont pour le Front-Office. Quant au Back-Office, il est lui aussi entièrement opérationnel pour être administré sur le terminal de votre choix.

Enfin, rappelons en gage de sérieux de cette solution que Drupal est actuellement installé pour des organisations hétérogènes et de tailles différentes : des associations, des partis politiques, des gouvernements, des entreprises...

Aujourd'hui, Drupal « propulse » des sites aussi prestigieux que le PS, l'UMP, le Ministère des finances, le Gouvernement Français, l'Élysée, PPR, Action contre la Faim, Amnesty international,



la Fédération française de Tennis (FFT), la maisons blanche, les différents sites de Radio France, le Musée du Louvre, Total, Voyages SNCF, Slate.fr...

Nul n'est à l'abri d'une faille de sécurité, mais Drupal bénéficie d'une « police » spécifique qui veille au bon fonctionnement de la solution et à la résolution rapide des failles de sécurité lorsqu'elles sont déclarées.

Drupal 8 pour l'e-marketeur

Depuis quelques années, un site « institutionnel » ne se résume plus seulement à la production de contenu éditorial et multimédia. Un site peut désormais servir de « tremplin » pour fidéliser et convertir les internautes en futurs acheteurs potentiels.

Les outils classiques comme la lettre d'information ou le référencement ne sont plus suffisants : l'avènement des réseaux sociaux, des sites e-commerce et la multiplication des nouveaux canaux de diffusion de l'information ont transformé « l'internaute passif » en « consommateur actif » : personnalisation et recommandations des contenus en fonction de ce qui est vu et cliqué, participation en postant des commentaires et des sujets de forums ou encore en partageant des contenus sur les réseaux sociaux.

L'accès aux « statistiques de consultation » a été remplacé par des solutions de « mesures d'audiences » évoluées qui permettent au-delà du nombre de visites / visiteurs / nombre de pages vues, de segmenter les visiteurs du site suivant plusieurs critères (origine géographique, device utilisé ou encore les parcours effectués sur le site)

Les lettres d'informations classiques sont devenues des newsletters personnalisées au(x) centre(s) d'intérêt des visiteurs. Il est désormais possible d'en connaître le taux d'ouverture, de savoir ce qui a été cliqué ou encore de programmer des relances.

Au-delà du rôle « stratégique » de son profil, l'e-marketeur doit :

- pouvoir déployer facilement et rapidement les bons contenus, au bon endroit, au bon moment, sur le bon équipement et dans la bonne langue
- disposer des métriques sur le succès des campagnes menées sur le site au travers des contenus, les réseaux sociaux, ou plus classiquement, l'email ou la newsletter
- définir des axes de segmentations du site ainsi que des règles d'enregistrement de « lead » potentiel dans une solution tierce pour pouvoir adresser de nouveaux clients.

Comment Drupal s'adresse aux e-marketeur

Drupal 8 facilite la création et la mise à jour des contenus. Il propose nativement de nombreuses fonctionnalités comme « Views » ou « CkEditor » qui sont « out-of-the-box », c'est-à-dire opérationnelles directement après l'installation de la solution et sans développement supplémentaire, mise à part un peu de configuration via l'interface.



La contribution peut se faire sur tout type de support, puisqu'au traditionnel Desktop, il est désormais possible de contribuer aussi sur smartphone et tablette. Enfin, Drupal permet de segmenter les contenus en fonction des audiences, de la langue ou de la source utilisée, notamment via un système de filtres hérités de « Views » qui sera possible d'étendre à d'autres critères.

Le multilinguisme a été, lui aussi, largement amélioré dans cette nouvelle version. Il n'est plus nécessaire d'installer de nombreux modules pour avoir une interface réellement adaptée pour des contributeurs géographiquement décentralisés, intervenant dans différentes langues.

Drupal 8 permet aussi une meilleure adhérence de systèmes tiers via le module « RESTful web services ». L'intégration d'un outil de Marketing Automation ou d'un CRM est simplifiée et la nouvelle interface REST accessible à des profils non techniques, peut tout à fait faire office de « Backbone » pour centraliser vos efforts marketing.

Drupal 8 pour les administrateurs de contenus

Les attentes des administrateurs de contenus ne sont finalement pas si éloignées de celles des équipes Marketing. La gestion de contenus reste le besoin principal des administrateurs. Ils doivent pouvoir produire du contenu le plus simplement et le plus rapidement possible.

Il faut simplifier les actions nécessaires à la création et modification d'un nouveau contenu : limiter le nombre de champs à saisir, avoir une logique de contribution « séquentielle » ... Il faut aussi automatiser au maximum les actions adjacentes : définir en amont les workflows et leurs suivis, optimiser le SEO, limiter le profil aux tâches qui lui incombent (ne pas lui afficher par exemple des fonctionnalités d'administration telles que la gestion du système sur lesquelles il n'a aucun droit).

Comment Drupal s'adresse aux administrateurs de contenus

Drupal 8 permet facilement de créer ou de modifier des contenus via l'éditeur WYSIWYG CkEditor 4, qui est désormais embarqué par défaut. A noter que cette solution reste remplaçable par un autre éditeur de votre choix, comme ce fut le cas pour Drupal 7, où, la mise en place du module WYSIWYG permettait le support de solution plus ou moins connues (et plus ou moins conseillées !) comme CKEditor, EpicEditor, jWysiwyg, markItUp, openWYSIWYG, TinyMCE, Whizzywig, YUI editor....

L'autre nouveauté, qui a par ailleurs était annoncée dès le début de l'initiative Drupal 8 (mai 2012 – cf. : <https://www.acquia.com/fr/blog/announcing-spark-authoring-improvements-drupal-7-and-drupal-8>), est l'utilisation de la distribution « Spark » pour faire de l'« inline Editing ». Concrètement, à la manière du module « Edit » pour Drupal 7, cela permet de pouvoir modifier directement le contenu en Front-Office, sans passer par les menus du Back-office.



Rappelons que la gestion de contenus reste globalement un point fort de la solution quelque soit la version de Drupal utilisée : il suffit de remplir simplement les champs qui constituent le type de contenu, voire d'en créer des nouveaux.

Drupal permet ainsi de créer facilement des contenus mais aussi des listes, des messages, des galeries, des tableaux, des cartes, des blocs.

WWW.SMILE.FR



LE GUIDE DRUPAL 8 D'ANGELA BYRON

Grâce à Angela Byron, Directrice du développement communautaire chez Acquia, nous pouvons aller plus loin pour examiner les nouvelles fonctionnalités et possibilités offertes par Drupal 8.

Angela Byron est une fervente adepte du concept Open source. Son travail consiste à examiner des correctifs pour la version de base Drupal Core et à les valider, à assister les contributeurs de la communauté, à coordonner l'équipe d'infrastructure et à répandre la bonne parole. Elle est également un contributeur clé du logiciel Drupal 8.

Angela est l'auteure principale du premier livre sur Drupal d'Oreilly intitulé « Using Drupal » (Utiliser Drupal). Le pseudo d'Angela sur Drupal.org est « webchick ».



WWW.SMILE.FR

Introduction

Que vous soyez concepteur de site, développeur de module ou de thème, ou simplement utilisateur final d'un site Drupal, Drupal 8 met à votre disposition tout ce dont vous avez besoin, et bien plus encore. Ce livre blanc va vous présenter ce qui change radicalement dans Drupal 8 pour les utilisateurs finaux, les concepteurs de site, les développeurs d'interface utilisateur et d'environnement de programmation (back-end).

Étant donné que Drupal 8 est toujours en cours de développement, il se peut que certains détails changent encore avant sa sortie. Les fonctionnalités de Drupal 8 sont en principe arrêtées, et les informations que vous trouverez ici devraient donc rester pertinentes. Les fonctionnalités de Drupal 8 ayant un équivalent dans Drupal 7 seront signalées, si elles existent.

Expérience de Création

Drupal a tout mis en œuvre pour proposer une expérience unique aux créateurs de contenu, car ce sont eux qui utilisent les sites Drupal au quotidien. Voici quelques-uns des changements que vous constaterez.



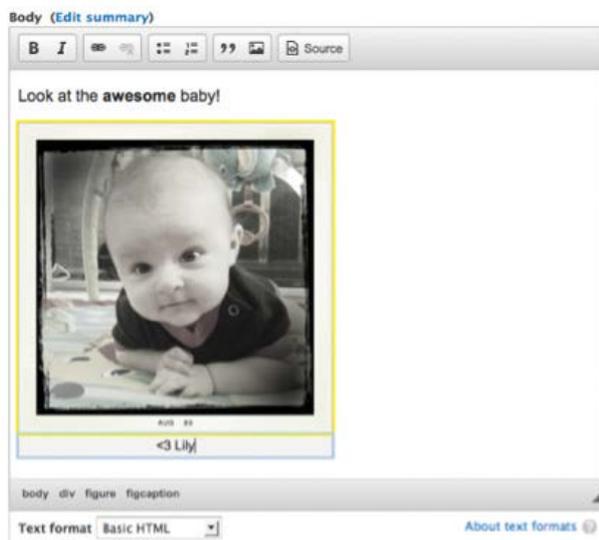
Spark

Spark est une initiative d'Acquia créée par Dries Buytaert dans le but [d'améliorer l'expérience de création du logiciel Drupal de base](#). L'équipe de développement Acquia a effectué des analyses des produits concurrents de Drupal (propriétaires et open source) et a fourni un travail intensif en collaboration avec les contributeurs Drupal Core jusqu'à la sortie. Ils ont apporté leur aide pour améliorer la version de base de Drupal tout en créant des adaptations et des [améliorations essentielles de Drupal 8 UX applicables à Drupal 7](#).



Editeur WYSIWYG

Il est loin, le temps où on écrivait les codes HTML à la main (au moins pour l'installation par défaut). Drupal 8 est livré avec l'éditeur WYSIWYG [CKEditor](#). En plus de ce qu'on peut attendre d'un éditeur WYSIWYG classique (des boutons pour le gras, l'italique, les images, les liens, etc.), cet éditeur prend également en charge les légendes éditables des images grâce aux nouvelles fonctionnalités des [Widgets](#) de CKEditor spécialement développées pour être utilisées dans Drupal. Notre priorité était de [conserver les avantages des contenus structurés de Drupal dans notre mise en œuvre WYSIWYG](#).



Drupal 8 est également doté d'une interface administrateur avec une fonction glisser-déposer qui permet d'ajouter et de supprimer des boutons de la barre d'outils WYSIWYG. Les balises HTML autorisées pour un type de texte donné sont alors automatiquement synchronisées, ce qui améliore considérablement leur simplicité d'utilisation. Les boutons sont réunis en groupes (groupes de boutons). Leur libellé n'est pas visible à l'œil nu, mais peut être lu par le lecteur d'écran, ce qui procure une expérience d'édition à la fois incroyable et accessible aux visiteurs du site internet.

Le module Editor de Drupal 8 englobe l'intégration du WYSIWYG, permettant ainsi d'intégrer pleinement d'autres bibliothèques dans la contribution.

Edition sur place

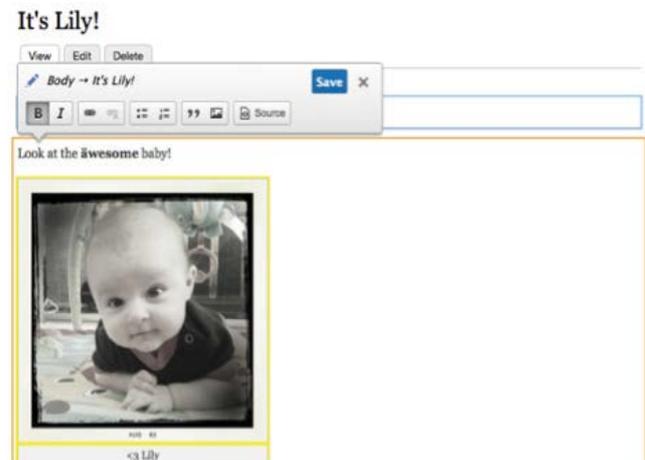
Dans Drupal 7, pour apporter une correction sur un site internet (par exemple corriger une faute d'orthographe ou ajouter une image manquante), il fallait utiliser un formulaire d'arrière-plan distinct de l'interface utilisateur du site sur laquelle apparaît le contenu. Le bouton de



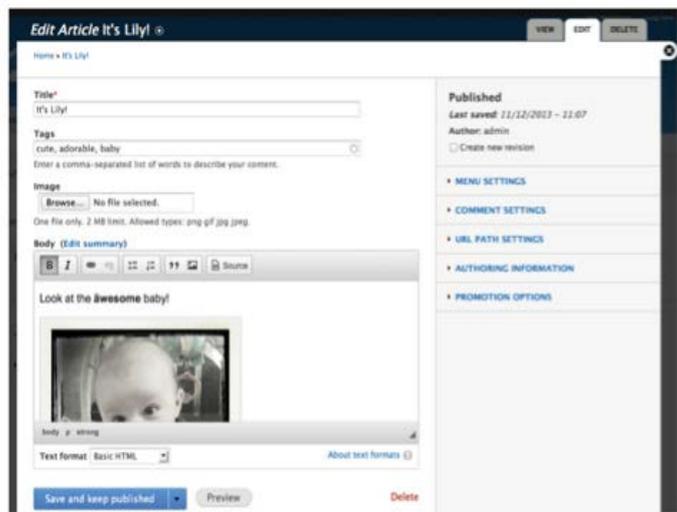
prévisualisation n'était d'aucune aide, car la prévisualisation n'était visible que sur le thème d'administration (deux fois, au cas où vous manquerez la première).

La nouvelle fonction d'édition sur place de Drupal 8 permet aux éditeurs de cliquer sur n'importe quel champ d'une partie de contenu et de l'éditer directement au niveau de l'interface utilisateur du site, sans avoir recours à un formulaire d'arrière-plan. Les contenus complets des nœuds, des blocs personnalisés et d'autres encore sont également éditables sur place.

Cette fonction d'édition sur place a été transposée dans Drupal 7 sous le nom de module [Quick Edit](#) (dénommé avant module [Edit](#)).

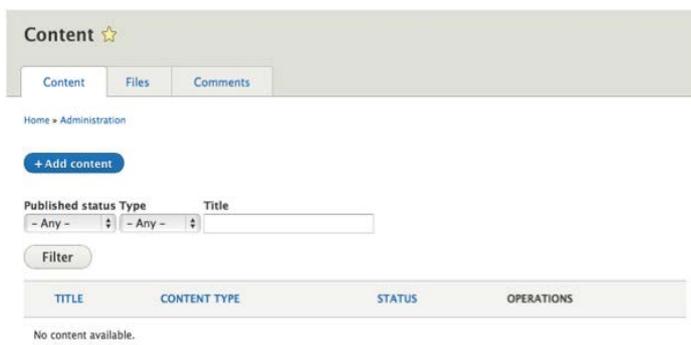


Une page de création de contenus repensée



Les [efforts menés par la communauté de l'équipe de Drupal en vue de faciliter son utilisation](#) ont donné naissance à une toute nouvelle page de création de contenus dans Drupal 8. Elle dispose de deux colonnes : une pour les champs principaux (la véritable partie « contenu » de votre contenu) et une autre pour les « bonus » (les paramètres optionnels utilisés moins souvent). Nous espérons ainsi que cette nouvelle présentation procurera aux créateurs de contenu une expérience moins prenante et qu'ils pourront se concentrer davantage sur les tâches en cours.

Un thème administrateur modernisé



Même s'il est [toujours en développement](#), le thème de Drupal 8 est mis au goût du jour par rapport à Drupal 7, notamment grâce au [nouveau guide de style pour le thème d'administration « Seven »](#).



Prise en charge native des ébauches

La prise en charge des « workflow » de publication est d'ores et déjà prévue dans les API du core (mode brouillon, ébauche). Cette modification devrait rendre les modules de publication de contenu tels que [Workbench](#) plus accessibles dans Drupal 8 et les versions ultérieures.

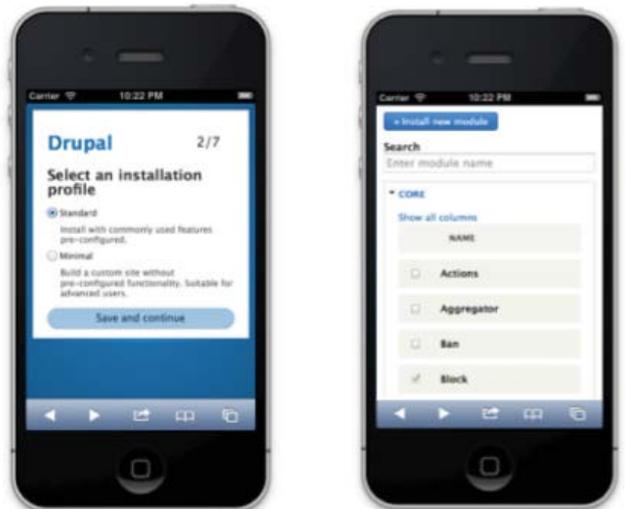
AMÉLIORATIONS POUR MOBILES

Outre les améliorations de l'expérience de création, l'autre point sur lequel se sont penchés les utilisateurs finaux de Drupal, concerne l'accessibilité des fonctionnalités aux appareils mobiles, cela afin de s'adapter à l'explosion mondiale du nombre de terminaux mobiles.

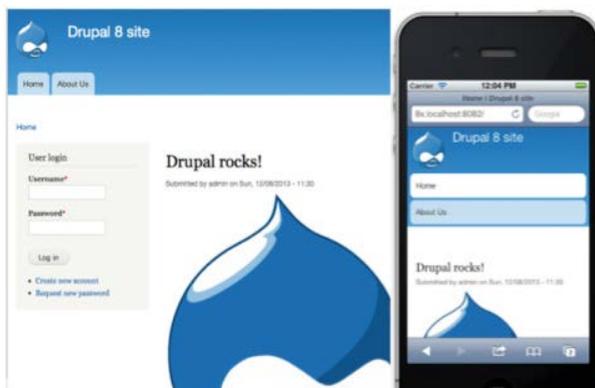
Mobile avant tout

Comme vous pourrez le constater, Drupal 8 a été conçu en vue de privilégier l'adaptation aux mobiles à tous les niveaux, depuis l'installation jusqu'à la page des modules. Les nouvelles fonctionnalités, telles que l'édition sur place, sont également conçues pour être utilisées sur un écran de taille réduite. Essayez Drupal 8 sur l'appareil mobile de votre choix, et [dites-nous](#) ce que vous en pensez.

Vous trouverez également une barre de recherche sur la page des modules. Téléchargez le [Module Filter](#) (Filtre) pour bénéficier d'une expérience similaire sur Drupal 7.



Tout adapter et redimensionner (thèmes, images, tableaux...)



Pour faire face au nombre inimaginable d'appareils mobiles connectés à Internet qui feront leur apparition dans les cinq années à venir et au-delà, Drupal 8 intègre son approche adaptable à tout ce qu'il fait.

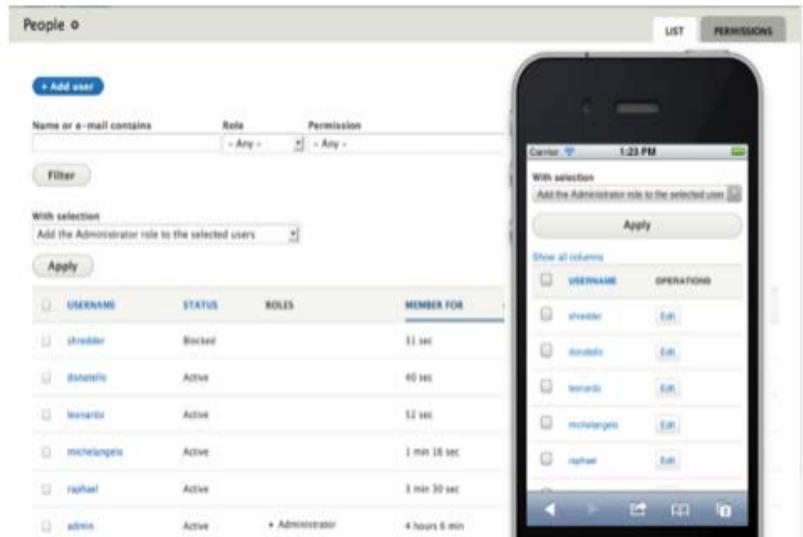
Tout d'abord, les thèmes principaux sont désormais adaptables et les éléments tels que les menus et les blocs sont automatiquement redimensionnés pour s'adapter à la taille de l'écran d'un appareil

mobile (si la fenêtre de visualisation est trop étroite, les éléments horizontaux sont orientés



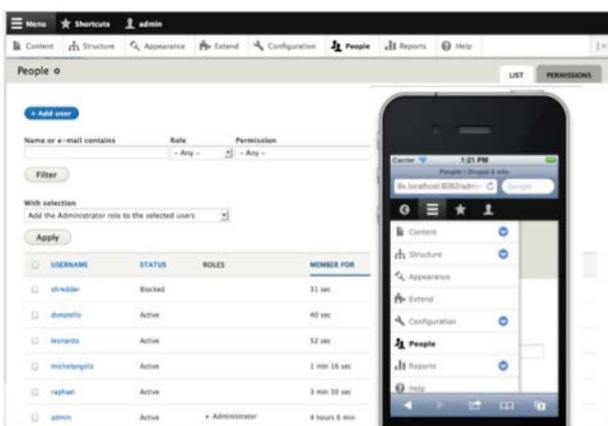
verticalement). Les images qui s'affichent en grand format sur un ordinateur voient leur taille réduite pour s'adapter à l'écran d'une tablette ou d'un smartphone grâce à la prise en charge intégrée d'images adaptables.

Drupal 8 prend également en charge les tableaux adaptables, de telle sorte que les colonnes des tableaux peuvent se voir attribuer un statut d'importance variable (élevée, moyenne ou faible). Sur les écrans larges, toutes les colonnes s'affichent, puis, selon la taille de l'écran, les colonnes les moins importantes sont masquées pour que le tableau soit parfaitement dimensionné à l'écran. Cette API est également intégrée au module Views (Affichages), ce qui vous permet de configurer vos propres écrans d'administration adaptables.



Les modules [Responsive Bartik](#) et [Responsive Tables](#) permettent d'obtenir des fonctions similaires dans Drupal 7. De nombreux thèmes de base adaptables à Drupal 7, notamment [Omega](#) et [Zen](#), vous aident à concevoir une présentation adaptable pour votre site internet.

Une barre d'outils simple d'utilisation sur mobile

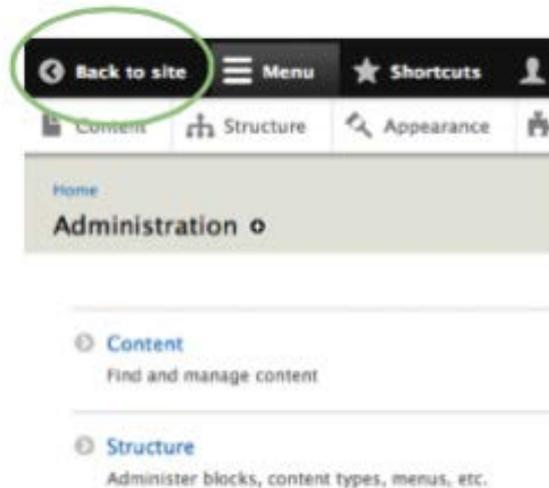


Drupal 8 contient désormais une barre d'outils administrateur toute nouvelle qui s'élargit et s'affiche automatiquement à l'horizontal sur les grands écrans et est réduite à une icône et s'affiche à la verticale sur les écrans plus petits. Comme toutes les nouvelles fonctionnalités de l'interface utilisateur dans Drupal 8, celle-ci fait l'unanimité en termes d'accessibilité, et permet ainsi aux utilisateurs de lecteurs d'écran de se déplacer aisément d'une partie du site à une autre.

Si cette fonctionnalité vous intéresse sur Drupal 7, téléchargez le module [Mobile Friendly Navigation Toolbar](#).



Performance de l'interface utilisateur



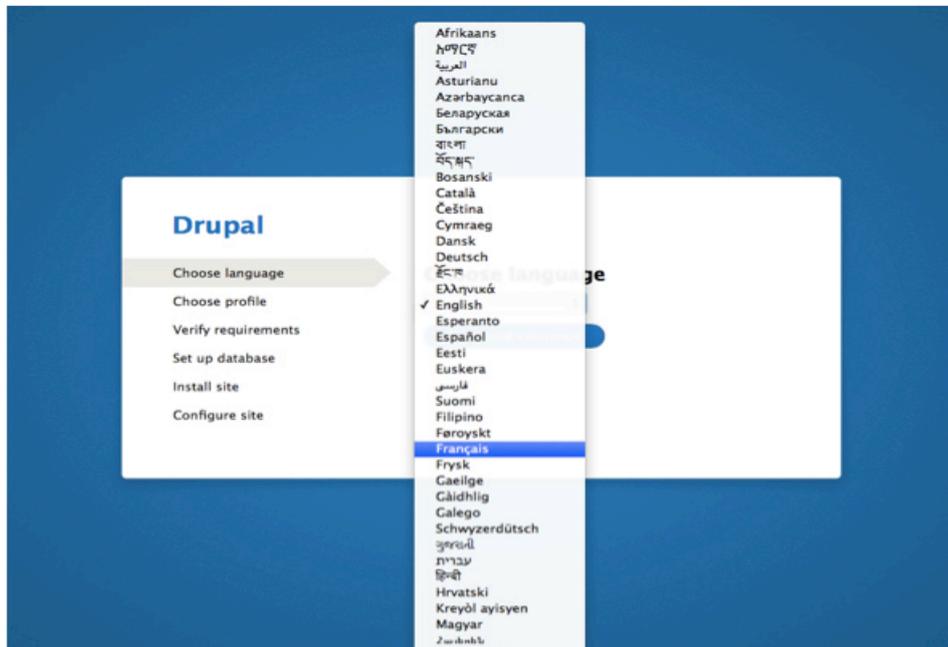
L'un des facteurs les plus importants pour la réussite ou l'échec d'une expérience mobile est la performance pure du site internet. Voilà pourquoi une grande partie du travail a consisté à réduire l'empreinte de Drupal 8 sur l'interface utilisateur. Dans de nombreux cas, le JavaScript natif a remplacé le jQuery, et évite le chargement de fichiers JavaScript par des utilisateurs anonymes. De même des versions alternatives « allégées » utilisables sur mobile ont remplacé les fonctionnalités JavaScript pures. C'est le cas par exemple du module Overlay (Calque) : le simple lien « Back to site » (Retour au site) de la barre d'outils administrateur est visible dans l'espace administrateur. Voir [Escape Admin](#), pour voir l'équivalent sur Drupal 7.

MULTILINGUE++

La « [Multilingual Initiative](#) » (Initiative multilingue) (D8MI) menée par [Gábor Hojtsy](#), membre d'Acquia, avec la participation de plus de [1 000 contributeurs](#), est un enjeu de développement décisif pour Drupal 8. Regardez plus en détails l'excellente série [Drupal 8 Multilingual Tidbits](#) de Gábor si vous voulez en savoir plus sur D8MI.



Multilingue avant tout

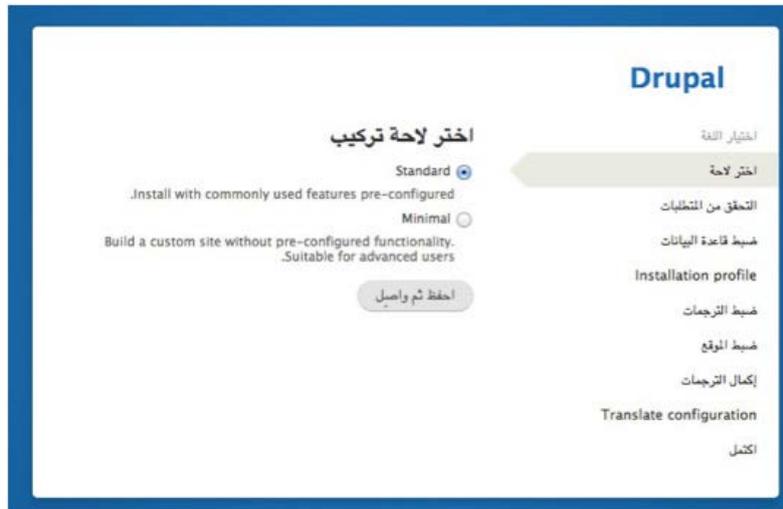


WWW.SMILE.FR

Le concept de base résolument multilingue de Drupal 8 est visible dès le tout début de l'installation.

Drupal 8 détecte la langue de votre navigateur, puis sélectionne automatiquement cette option dans le menu déroulant pour vous faciliter la tâche. Mieux : si vous installez Drupal dans une langue autre que l'anglais (ou si vous ajoutez ultérieurement une autre langue à votre site), Drupal 8 télécharge automatiquement la dernière traduction de l'interface sur <https://localize.drupal.org>, ce qui vous permet de procéder à l'installation complète et à la définition des paramètres dans votre propre langue. En cela, il diffère de Drupal 7, qui affiche pour les utilisateurs un long texte compact (en anglais) leur expliquant comment mettre les fichiers téléchargés dans les répertoires adéquats pour installer le logiciel.

Cette fonctionnalité s'applique également aux langues qui se lisent de droite à gauche, telles que l'arabe (Drupal 8 étant encore en développement, il est possible que certains segments de texte traduits soient manquants).



Drupal 8 a délaissé le principe de l'anglais en tant que langue « spéciale ». Si vous sélectionnez une langue autre que l'anglais, l'option « anglais » n'apparaît plus dans la configuration de votre site, à moins que vous ne l'activiez volontairement. Vous pouvez également rendre l'anglais « traduisible » pour convertir certains segments de texte, tels que « Log in / Log off » (Se connecter/Se déconnecter) ou « Sign in / Sign off » (S'inscrire/Se désinscrire).

Moins de modules pour plus d'efficacité

La conception d'un site multilingue avec Drupal 7 nécessite environ 30 modules et de nombreuses manipulations compliquées. Dans Drupal 8, toutes ces fonctionnalités (et d'autres) ont été réparties selon quatre modules, qui confèrent à eux seuls à l'interface de Drupal 8 un aspect plus multilingue que tous les modules dédiés de Drupal 7.

« **Language** » (Langue) fournit à Drupal 8 une assistance linguistique sous-jacente. C'est le module de base et il est requis par les autres modules multilingues.

« **Configurations Translation** » (Traduction de configurations) permet la traduction des blocs, des menus, des vues et d'autres éléments (Similaire à la fonction d'internationalisation de Drupal 7).

« **Content Translation** » (Traduction de contenu) permet la traduction d'éléments tels que les nœuds, les termes taxonomiques et les commentaires (diffère du module « Content Translation » dans Drupal 7 Core ; il ressemble davantage à [Entity Translation](#)).

« **Interface Translation** » (Traduction d'interface) permet la traduction de l'interface utilisateur elle-même (il s'agit de l'équivalent du module de base « Locale » (Paramètres locaux) dans Drupal 7).

Vous vous demandez sûrement : « Pourquoi quatre modules, et pas seulement un ? ». Parce que les sites en une seule langue autre que l'anglais sont eux aussi des cas d'utilisation valides, et même certains sites multilingues peuvent ne pas avoir besoin de ces fonctionnalités (par exemple, si on veut toujours garder le contenu généré par les utilisateurs dans sa langue d'origine). Ce type



de granularité offre aux concepteurs de site la possibilité de choisir la combinaison qui répond le mieux à leurs exigences spécifiques.

Sélection de la langue pour TOUS les éléments

View name and description

Human-readable name
Content
A descriptive human-readable name for this view. Spaces are allowed

View language
✓ English
Hungarian
Lists and other textual elements in this view.

View tag
default
Optionally, enter a comma delimited list of tags for this view to use in filtering and sorting views on the administrative page.

View description
Find and manage content.
This description will appear on the Views administrative UI to tell you what the view is about.

Apply Cancel

WWW.SMILE.FR

Tout est traduisible, des paramètres de configuration du système et des composants du site, tels que les blocs, les vues et les menus, aux valeurs de champs individuels ou aux contenus.

Pour ce qui est des entités de contenu (commentaires, nœuds, utilisateurs, termes taxonomiques, etc.), vous disposez d'encore davantage d'options, par exemple, de la possibilité d'afficher ou non le sélecteur de langue, de configurer les nouveaux contenus pour qu'ils prennent par défaut la langue du site, la langue favorite de l'auteur du contenu ou d'autres valeurs.

Des interfaces utilisateur de traduction rationalisées

De très gros efforts ont été faits pour améliorer l'expérience de l'utilisateur avec les fonctionnalités multilingues de Drupal 8. Vous découvrirez des interfaces de traduction mieux pensées et mieux intégrées tout au long de cette expérience.



WWW.SMILE.FR

<p>Singular form One translation file imported for the enabled modules.</p> <p>Plural form @count translation files imported for the enabled modules.</p>	<p>Singular form A bekapcsolt modulokhoz egy fordítási fájl lett betöltve.</p> <p>Plural form A bekapcsolt modulokhoz @count fordítási fájl lett betöltve.</p>
<p>Singular form Changed the content type of 1 post from %old-type to %type.*</p> <p>Plural form Changed the content type of @count posts from %old-type to %type.*</p>	<p>Singular form Egy tartalomtípus %old-type helyett %type lett.</p> <p>Plural form @count tartalomtípus %old-type helyett %type lett.</p>
<p>All (@count)</p>	<p>Mind (@count)</p>
<p>Pass (@count)</p>	<p>Áment (@count)</p>
<p>Fail (@count)*</p>	<p>Megbukott (@count)</p>
<p>Singular form 1 debug message</p> <p>Plural form @count debug messages</p>	<p>Singular form 1 hibakereső üzenet</p> <p>Plural form @count hibakereső üzenet</p>



Custom language settings

- Comment
- Content
- Custom Block
- Taxonomy term
- User

Content

CONTENT TYPE	CONFIGURATION
Article	<p>Default language</p> <p>Site's default language (Hungarian) ↓</p> <p>English Hungarian - Not specified - - Not applicable -</p>
Basic page	<p>Site's default language (Hungarian) ↓</p> <p>Current interface language Author's preferred language</p>

Taxonomy term

VOCABULARY	CONFIGURATION
Tags	<p>Default language</p> <p>Site's default language (Hungarian) ↓</p> <p>Explanation of the language options is found on the languages list page.</p> <p><input type="checkbox"/> Show language selector on create and edit pages</p>

Save

Support de translitération

L'ajout du module [Transliteration](#) à la base de Drupal 8 est vraiment pratique. Il convertit automatiquement les caractères spéciaux tels que « ç » et « ü » en « c » et en « u » pour que les noms d'appareil, de fichiers téléchargés, de chemins et de résultats de recherche soient plus agréables à lire.

...Et bien plus !

Voici quelques ajouts pour les concepteurs de site qui valent la peine d'être mentionnés :

- Plusieurs pages de gestion de contenus construites grâce au module Views facilitent la personnalisation linguistique et plus particulièrement, celle des espaces administrateur, auxquels on peut ajouter des filtres de langue, des colonnes de langue, etc.



- Contrairement à l'intégration du module « Entity Translation » de Drupal 7, le module de base « Content Translation » de Drupal 8 s'intègre bien à la fonction « Search » (Recherche) dans la version de base, et l'API de recherche recueille également davantage d'informations linguistiques.
- Le système de sélection de la langue prend désormais en charge une autre langue administrateur, ce qui facilite la gestion des sites multilingues par leurs administrateurs.

LES CONCEPTEURS DE SITE À L'HONNEUR

Bien que les [améliorations de l'expérience](#) de création et les [améliorations mobiles](#) de Drupal 8 soient plutôt orientées utilisateurs finaux et concepteurs de contenu de sites internet Drupal, Drupal 8 intègre également des améliorations notables des outils de conception des sites.

Views intégré dans le core

Le module « [Views](#) », qui est le [module le plus fréquemment utilisé par les contributeurs de Drupal](#), est désormais inclus nativement dans le nouveau Drupal 8, ce qui est une bonne nouvelle. Et non seulement le module « Views » est inclus dans la version de base, mais la plupart des listes d'administration telles que « Content » (Contenu), « People » (Personnes) et « Files » (Fichiers), et les différents blocs de barre latérale, de nombreux fils RSS et la page d'accueil par défaut, ont également été convertis dans « Views ». Cela permet de personnaliser tous ces éléments en quelques clics, par exemple, d'ajouter une recherche par mots clés (Full Text) dans la liste

TITLE	CONTENT TYPE	AUTHOR	STATUS	UPDATED	OPERATIONS
News flash!	Article	admin	Published	10/20/2013 - 22:22	[edit] [delete]
A really important blog post	Article	admin	Published	10/20/2013 - 22:22	[edit] [delete]

```
{ "nid": [{"value": "1"}], "uid": [{"value": "b39de75e-57de-4d11-8b54-844e81ad8516"}], "vid": [{"value": "1"}], "type": "value", "article": [{"value": "en"}], "title": [{"value": "Lorem ipsum"}], "nid": [{"target_id": "1"}], "status": [{"value": "1"}], "created": [{"value": "1382475100"}], "changed": [{"value": "1382475179"}], "promote": [{"value": "1"}], "sticky": [{"value": "0"}], "revision_timestamp": [{"value": "1382475179"}], "revision_uid": [{"target_id": "1"}], "log": [{"value": "1"}], "body": [{"value": "p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum."}]
```



« People », ou de courts descriptifs d'images dans la liste « Content ».

Tous les aspects de Views que vous connaissiez et appréciez sont inclus, et vous bénéficiez également de fonctions de gestion d'administration optimisées pour le mobile, d'une expérience utilisateur sans égale, d'une accessibilité améliorée, de la possibilité de créer des listes de tableau adaptables et d'exposer toutes les listes via un web service REST consultables par le biais d'une application mobile ou de n'importe quel autre service externe.

Des blocs améliorés et plus nombreux

Dans Drupal 8, vous découvrirez quelques nouvelles fonctionnalités ayant trait aux blocs. D'abord, tout comme dans « Views », de nombreux éléments de site programmés en dur ont été convertis en blocs, notamment les aides à la navigation, le nom du site et le slogan ; bien d'autres sont en préparation. Il est ainsi plus facile de mettre en forme les pages dans l'interface utilisateur.

Ensuite, vous n'êtes plus limité au placement d'un seul bloc dans une zone unique : vous pouvez désormais placer d'autres blocs et les réutiliser dans plusieurs endroits. Un bloc « Navigation » pourra par exemple être utilisé dans l'en-tête et le pied de page. Plus besoin d'utiliser le module [Multiblock](#) !

Enfin, vous pouvez créer des types de bloc personnalisés, ce qui vous permet de maîtriser la granularité des différents styles, des différents champs, et bien plus encore. Ainsi, vous avez la possibilité de créer un bloc « Ad » (Annonce) grâce à un champ « Ad code » (Code annonce) pouvant contenir des bribes de code JavaScript à partir d'un service d'annonces à distance.

The screenshot shows the Drupal 8 Structure page. At the top, there are tabs for 'Bartik' and 'Seven'. Below the tabs is a breadcrumb trail: 'Home » Administration » Structure'. A paragraph explains the drag-and-drop interface for adding and configuring blocks. Below this is a section titled 'Demonstrate block regions (Bartik)' with a '+ Add custom block' button. The main content area is a table with columns 'BLOCK', 'CATEGORY', 'REGION', and 'OPERATIONS'. The table is divided into four regions: 'Header', 'Help', 'Highlighted', and 'Featured'. The 'Header' region is currently empty. The 'Help' region contains one block: 'System Help' from the 'System' category, located in the 'Help' region, with a 'Configure' button. The 'Highlighted' and 'Featured' regions are also empty. On the right side, there is a 'Place blocks' sidebar with a search filter and three expandable sections: 'FORMS' (containing '+ Search form' and '+ User login'), 'LISTS (VIEWS)' (containing '+ Recent comments', '+ Recent content', and '+ Who's online'), and 'MENUS' (containing '+ Administration' and '+ Footer').

BLOCK	CATEGORY	REGION	OPERATIONS
Header			
No blocks in this region			
Help			
+ System Help	System	Help	Configure
Highlighted			
No blocks in this region			
Featured			
No blocks in this region			



Allowed number of values
Limited 1

Type of item to reference*
Content

Save f

REFERENCE TYPE

Reference method*
Default

Content types*
 Article
 Basic page

Sort by
- None -

December-2013

Sun	Mon	Tue	Wed	Thu	Fri	Sat
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	31	1	2	3	4

yyyy-mm-dd

Modes de formulaire

En plus de la fonctionnalité « View modes » (Modes d’affichage) de Drupal 7, qui permet la création de plusieurs formats d’affichage en fonction du contexte (par exemple, affichage d’une image au format icône dans la vue de présentation de votre contenu et d’une image à la taille réelle dans la vue par défaut), Drupal 8 intègre la notion de « form modes » (modes de formulaire), qui représente l’équivalent pour les formulaires de saisie de données. Voici un exemple de configuration du formulaire d’enregistrement d’un utilisateur à partir d’un formulaire d’édition utilisateur, qui vous permet de masquer les champs les plus abstraits, pour une expérience d’utilisation plus simple.

Manage form display ☆

Settings Manage fields Manage form display Manage display

Default Register

Home » Administration » Configuration » People » Account settings » Manage form display

Show row weights

FIELD	WIDGET	
+ User name and password	Visible	
+ Picture	Image	Preview image style: Thumbnail (100x100) Progress indicator: throbber
+ Bio	<input checked="" type="checkbox"/> Text area (multiple rows) <input type="checkbox"/> Hidden	Number of rows: 5
Disabled		
+ Timezone	- Hidden -	
+ Language settings	- Hidden -	
+ Contact settings	- Hidden -	

Save



Visite

Le nouveau module d'exploration de Drupal 8, met à la disposition des concepteurs de site, des astuces sous forme d'info-bulles contextuelles. De vues globales aux interfaces administrateur, le module permet de présenter les nouvelles terminologies et guide les utilisateurs dans l'accomplissement des étapes nécessaires à la configuration des éléments du site.

WWW.SMILE.FR

Home » Administration » Structure » Views

Displays

Page [+ Add](#)

Display name: Page

Displays in this view

A view can consist of multiple displays. A display is a way of outputting the results E.g. as a page or in a block. The available displays in your view are show here.

Next 1 of 10

Content: Node operations bulk form
Content: Title (Title)
Content: Type (Content Type)

PAGE SETTINGS

Path: /admin/content/node

Menu: Tab: Content | Paren

Access: Permission |

[Access the Content overview](#)

HEADER

FOOTER

NO RESULTS BEHAVIOR

À la fois moins et plus, grâce aux modules

Vous ne trouverez pas dans Drupal 8 certains des modules qui étaient présents dans Drupal 7, à savoir « Blog », « Dashboard » (Tableau de bord), « Open ID » (ID ouvert), « Overlay » (Calque), « PHP filter » (Filtrage PHP), « Poll » (Sondage), « Profile » (Profil) et « Trigger » (Déclenchement) (et le thème « Garland »). Vous découvrirez à leur place de nouveaux modules pour lesquels les fonctionnalités ont été réparties en sous-parties, telles que « Menu Links/Menu UI », « Block/Custom Block », « Ban/History/Actions » (auparavant compris dans User/Node/System module), etc.

L'ouvrage de Heather James, intitulé « [Drupal 8 Site Building Preview – Less Is More](#) », contient un résumé complet de l'état des modules, notamment les modules de contribution devenus obsolètes à présent, en raison des fonctionnalités présentes dans la version de base de Drupal 8.



The screenshot shows the 'Extend' page in Drupal 8. At the top, there are buttons for 'List', 'Update', and 'Uninstall'. Below that, there is a search bar with the text 'Enter module name'. The main content area is titled 'CORE' and contains a table with the following modules:

NAME	DESCRIPTION
<input type="checkbox"/> Actions	Perform tasks on specific events triggered within the system.
<input type="checkbox"/> Aggregator	Aggregates syndicated content (RSS, RDF, and Atom feeds) from external sources.
<input type="checkbox"/> Ban	Enables banning of IP addresses.
<input checked="" type="checkbox"/> Block	Controls the visual building blocks a page is constructed with. Blocks are boxes of content rendered into an area, or
<input type="checkbox"/> [truncated]	Allow users to create and manage related content in an outline.

A sa sortie, Drupal Core comprendra suffisamment de fonctionnalités, afin que, pour la première fois, les concepteurs de site puissent créer des sites relativement sophistiqués sans devoir installer plus de 30 modules de contribution. Hourra !

Processus de migration

Bien que l'interface graphique ne soit pas encore présente dans Drupal 8 Core, la mise à niveau vers cette version majeure sera facilitée par un framework de migration, un avantage offert par le port D8 des modules [Migrate](#) et [Migrate Drupal-to-Drupal](#). Les mécanismes de migration de Drupal 6 (déjà présents dans Drupal 8.x) et Drupal 7 (en développement intensif) seront disponibles dans la version finale qui sera proposée. La principale différence pour vous en tant que concepteur de site est qu'au lieu de garder votre site hors ligne pendant des heures durant la tentative de mise à jour du schéma de votre base de données par plusieurs scripts, vous pourrez continuer à faire tourner Drupal 6/7 pendant que vous concevrez votre nouveau site en Drupal 8. Vous pourrez continuer d'effectuer la migration de vos contenus (fourni par les modules Core/contrib/custom) jusqu'à ce que le transfert soit effectué. Vous n'aurez qu'un changement de routeur/DNS à effectuer à la fin, ce qui ne nécessite qu'une très courte interruption du système.

Pour plus d'informations sur le processus de mise à jour vers la version principale améliorée de Drupal 8, faites un tour sur le blog de Moshe Weitzman « [Drupal 8 – Improved Upgrade Process](#) », lancé en décembre 2013.



LES ASPECTS TECHNIQUES POUR LES DEVELOPPEURS

AMELIORATIONS POUR LE DEVELOPPEUR FRONTEND

Drupal 8 contient de nombreuses améliorations de l'API frontend, notamment le HTML5, les bibliothèques, des améliorations de l'accessibilité, de nouveaux thèmes et des éléments d'interface graphique, pour n'en citer que quelques-unes.

HTML5

Tout l'affichage de Drupal a été converti pour que l'on puisse utiliser les balises sémantiques du HTML5, et non les balises XHTML de Drupal 7. Cela signifie que vous trouverez des balises telles que <nav>, <header>, <main> et <section> dans les modèles par défaut de Drupal. Cet effort général vise à alléger le balisage par défaut.

Le HTML5 offre également de nouveaux types d'entrées de formulaire, telles que « date », « tel » (tél) et « email » (adresse électronique), qui peuvent fournir des interfaces utilisateur ciblées sur les appareils mobiles (par exemple, n'afficher que le pavé numérique lorsque vous entrez des numéros de téléphone), ce qui permet d'uniformiser les données saisies. L'API de création de formulaires de Drupal propose ces nouveaux types, ce qui vous permet également de créer de nouveaux types de champ. Dans Drupal 7, l'équivalent se trouve dans le module [Elements](#).

En outre, vous remarquerez les remplacements par du HTML5/CSS3 de nombreux éléments qui nécessitaient auparavant de trouver des solutions personnalisées : redimensionnement des zones de texte et les classes first/last/odd/even (premier/dernier/impair/pair), remplacées par des pseudo-sélecteurs CSS; les groupes de champs déroulants ont été majoritairement remplacés par l'élément <details>.

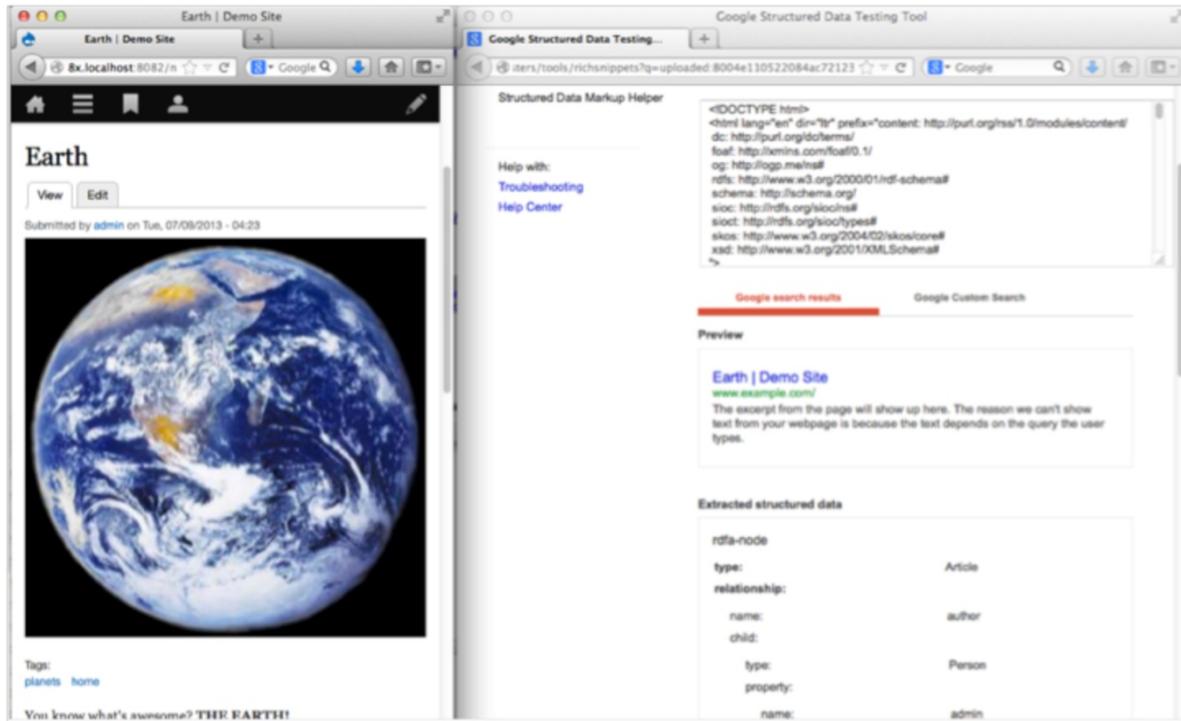


Nouvelles bibliothèques et nouveaux assistants pour l'interface

Bien que Drupal soit passé sur jQuery depuis la version 5, puis sur jQuery UI depuis Drupal 7, Drupal 8 intègre une gamme élargie de bibliothèques frontend, par exemple [Modernizr](#) (qui permet de savoir facilement si un navigateur supporte les fonctions tactiles ou le HTML5/CSS3), [UndersCore.js](#) (une bibliothèque d'assistant JS allégée) et [Backbone.js](#) (un système JavaScript de modèle-vue-contrôleur). Ces bibliothèques supplémentaires permettent de créer des applications faciles d'utilisation sur mobile et dotées d'une interface riche dans Drupal. Elles sont utilisées par la plupart des fonctionnalités améliorées d'[Authoring Experience](#) (Expérience de création) et [Mobile](#) de Drupal 8.

Sortie native sur Schema.org

Dans sa quête d'amélioration du référencement naturel par les moteurs de recherche, le module RDFa de Drupal 8 permet la création d'un balisage à partir de [schema.org](#). Les moteurs de recherche tels que Google, Yahoo!, Bing et Yandex peuvent ainsi plus facilement extraire des données, par exemple l'identité de l'auteur d'une partie d'un contenu, afin de donner du sens à ce contenu.



Une accessibilité sans cesse améliorée

Drupal 8 a apporté des améliorations aux [performances d'accessibilité, déjà exemplaires de Drupal 7](#). Drupal 8 utilise largement les [attributs WAI-ARIA](#) pour donner du sens aux composants graphiques d'administration, notamment l'éditeur sur place et la barre d'outils adaptable. Pour l'environnement des développeurs, Drupal 8 propose de nombreux nouveaux [outils d'accessibilité sur JavaScript](#) (JS), ce qui permet aux développeurs de modules de créer facilement des applications accessibles. Des efforts importants ont été réalisés pour [fournir un test automatisé des fonctionnalités d'accessibilité grâce à la bibliothèque Quail](#).

Nouveau système de thème : Twig

Drupal 8 présente [Twig](#), qui succède au système de thème basé sur PHP de Drupal 7 et ses versions antérieures. Twig, comme beaucoup d'autres moteurs de templates équivalents pour d'autres projets, permet aux concepteurs ayant des compétences en HTML/CSS de modifier le balisage sans être un expert en PHP. Par exemple, plutôt que de connaître les différentes syntaxes entre les tableaux associatifs et les objets, et apprendre comment s'en servir, il suffit d'utiliser une déclaration `{{ foo.bar }}`, et le tour est joué. Seuls des circuits logiques avec des conditions ou des boucles peuvent être contenus dans les balises `{% ... %}`.

Vous trouverez ci-dessous un extrait de `page.html.twig` (l'équivalent de `page.tpl.php` dans Drupal 7) qui vous donnera un aperçu des fonctionnalités de Twig, ainsi que la prise en charge de balises HTML5 et d'ARIA.



```
<?php
  <main role="main">
    <a id="main-content"></a>{# link is in html.html.twig #}

    <div class="layout-content">
      {{ page.highlighted }}

      {{ title_prefix }}
      {% if title %}
        <h1>{{ title }}</h1>
      {% endif %}
      {{ title_suffix }}

      {{ tabs }}

      {% if action_links %}
        <nav class="action-links">{{ action_links }}</nav>
      {% endif %}

      {{ page.content }}

      {{ feed_icons }}
    </div>{# /.layout-content #}

    {% if page.sidebar_first %}
      <aside class="layout-sidebar-first"
        role="complementary">
        {{ page.sidebar_first }}
      </aside>
    {% endif %}

    {% if page.sidebar_second %}
      <aside class="layout-sidebar-second"
        role="complementary">
        {{ page.sidebar_second }}
      </aside>
    {% endif %}

  </main>
?>
```

Comment créer ces variables si vous ne pouvez plus utiliser du PHP directement dans les templates ? Grâce à la fonction `THEME_preprocess_HOOK()`, vous pouvez faire comme d'habitude (bien qu'elles se trouvent dans un fichier nommé `THEME.theme` au lieu de `template.php`). Twig applique d'office une séparation de la présentation et de la logique métier, et toutes les variables insérées dans les fichiers modèles sont automatiquement effacées, ce qui réduit de loin les risques de danger tels que les vulnérabilités XSS et rend la création de thèmes (ou « theming ») dans Drupal 8 plus sûre que jamais.

L'autre information positive à propos de Twig est que lorsque vous passez en mode de débogage en utilisant `debug: true;` dans le fichier `services.yml` de votre site, des commentaires utiles sur les codes s'affichent dans le balisage généré par Drupal. Ils vous indiquent où trouver le template que vous essayez de modifier et la « suggestion de thème » particulière utilisée pour créer le balisage. Par exemple :



```
<div class="content">

<!-- THEME DEBUG -->
<!-- THEME HOOK: 'node' -->
<!-- FILE NAME SUGGESTIONS:
  * node--1--full.html.twig
  * node--1.html.twig
  * node--article--full.html.twig
  * node--article.html.twig
  * node--full.html.twig
  x node.html.twig
-->
<!-- BEGIN OUTPUT from 'core/themes/bartik/templates/
node.html.twig' -->
<article data-history-node-id="1" data-quickedit-entity-
id="node/1" role="article" class="contextual-region node
node--type-article node--promoted node--view-mode-full
clearfix" about="/node/1" typeof="schema:Article">
  ...
</article>

<!-- END OUTPUT from 'core/themes/bartik/templates/node.
html.twig' -->

</div>
```



C'est un peu comme avoir le fameux module [Theme developer \(Développeur de thème\)](#) intégré à la version de base !

Rapide par défaut

Wim Leers, le directeur spirituel des performances chez Acquia suggère que le meilleur moyen de rendre Internet globalement plus rapide est de [rendre les CMS de premier plan rapides par défaut](#). Cela implique que les CMS doivent mettre leurs paramètres hautes performances à un seul endroit plutôt que de forcer leurs utilisateurs à les chercher dans plusieurs emplacements. C'est exactement ce qui a été fait pour Drupal 8. Vous remarquerez que Drupal 8 intègre des fonctionnalités d'agrégation de CSS et de JavaScript dès le départ pour une installation par défaut plus rapide, et c'est tant mieux !

Cela signifie pour vos développeurs d'interface que Drupal ne vous offre pas des conditions idéales pour créer des thèmes par défaut, à moins que vous ne désactiviez manuellement les paramètres de performance (même toucher à la CSS de la version de base directement n'entraînera aucune modification).

Heureusement, Drupal 8 intègre un fichier `sites/example.settings.local.php` qui permet justement de modifier le code. Ce fichier définit en dur les paramètres de performance pour pouvoir les désactiver, ce qui est extrêmement utile dans le cadre d'un environnement de développement. Il suffit de copier ce fichier, de le renommer `sites/default/settings.local.php` et dé-commenter les lignes suivantes dans `sites/default/settings.php` :

```
<?php
# if (file_exists(__DIR__ . '/settings.local.php')) {
#   include __DIR__ . '/settings.local.php';
# }
?>
```

Votre nouveau fichier `settings.local.php` pointe vers `development.services.yml`, qui contient des paramètres par défaut spécifiques à Twig pour la phase de développement, par exemple des paramètres qui activent le mode de débogage et désactivent le cache. Passer ces paramètres sur `true` rendra certainement votre site de développement plus lent, mais facilitera grandement la création de thèmes, car vous serez en mesure de voir immédiatement le résultat de vos modifications dans les modèles de Twig sans avoir à vider le cache.

Parmi les autres nouveautés concernant les performances de l'interface, bien que Drupal 8 intègre toujours les dernières versions de jQuery et de jQuery UI, il semble de plus en plus s'écarter des bibliothèques de ce genre au profit du très ordinaire JavaScript et ce, afin de maintenir la rapidité de l'interface, facteur important des appareils mobiles. L'installation par défaut de Drupal 8 ne charge aucun programme JavaScript pour l'utilisateur !



[Même s'il reste du travail](#) en matière d'optimisation des performances, une fois lancé, Drupal 8 devrait proposer une expérience bien plus rapide sur l'interface pour les visiteurs du site. Hourra !

Nouveaux éléments pour l'interface utilisateur

Drupal 8 intègre de nombreux composants d'interface utilisateur sur les pages du backoffice, notamment des boîtes de dialogue et des boutons multi-actions, qui étaient présents dans le module [Chaos tool suite \(ctools\)](#) de Drupal 7 et versions antérieures. Drupal 8 utilise des types de bouton, de « primary » (principal), qui est l'action de formulaire par défaut et était coloré en bleu dans Seven, et de « danger », coloré en rouge. Ces concepts sont censés aider les utilisateurs à faire les bons choix lorsqu'ils sont face à plusieurs possibilités sur un formulaire.





Des thèmes adaptables

Comme mentionné dans l'article des [Améliorations mobiles](#), Drupal 8 intègre de nombreuses fonctionnalités nouvelles et adaptables, notamment les thèmes, barres d'outils, images et tableaux.

Pour prendre en charge ces fonctionnalités, les thèmes peuvent désormais déclarer des [Breakpoints](#) (points de rupture) - (la hauteur, la largeur et la résolution à partir desquelles la présentation est modifiée afin de s'adapter aux navigateurs et appareils mobiles) qui peuvent être utilisés par les fonctionnalités adaptables. Vous remarquerez cependant que les [réglages de déplacement des points de rupture pour les fichiers *.info.yml de thème et de module](#) sont un point sur lequel nous travaillons encore pour ces raisons précises, en changeant la mise en œuvre exacte.

[Drupal 8 devrait également intégrer](#) le [nouvel élément <picture>](#) que les navigateurs commencent à prendre en charge. Cela représente une nette amélioration des performances de l'interface, en particulier sur les appareils mobiles, car cela permet de transmettre des images plus petites (les images étant en règle générale l'élément le plus volumineux à charger sur une page) pour les plus petits écrans, ce qui économise des données. (Merci à [Marc Drummond](#) pour cette information).

Une nouvelle méthode pour sélectionner le JS/CSS à ajouter à une page

Concernant les performances : avant, lorsque vous vouliez ajouter du CSS ou du JS à une page, vous deviez utiliser respectivement les fonctions `drupal_add_css()` et `drupa_add_js()`. Ce n'est plus le cas ! Vous n'avez plus qu'à insérer n'importe quelle ressource internet en JS/CSS dans les propriétés `#attached` d'un bloc de la page. Par exemple :

seven.theme

```
function seven_form_node_form_alter(&$form, &$form_state) {  
  ...  
  $form['#attached'] = array(  
    'css' => array(drupal_get_path('module', 'node') . '/css/  
    node.module.css'),  
  );  
  ...  
}
```

Bien que cela fonctionne très bien pour les ressources internet uniques qui n'ont pas de lien de dépendance, la méthode la plus commune recommandée consiste à enregistrer une ou plusieurs ressource(s) JS/CSS (avec leurs éléments dépendants) en tant que bibliothèque dans `MODULE/THEME.libraries.yml`, puis à ajouter une référence dans les propriétés `#attached`. Par exemple :

seven.libraries.yml



```
maintenance-page:
  version: VERSION
  js:
    js/mobile.install.js: {}
  css:
    theme:
      maintenance-page.css: {}
  dependencies:
    - system/maintenance

install-page:
  version: VERSION
  js:
    js/mobile.install.js: {}
  css:
    theme:
      install-page.css: {}
  dependencies:
    - system/maintenance

drupal.nav-tabs:
  version: VERSION
  js:
    js/nav-tabs.js: {}
  dependencies:
    - core/matchmedia
    - core/jquery
    - core/drupal
    - core/jquery.once
    - core/jquery.intrinsic
```



seven.theme

```
<?php
function seven_preprocess_install_page(&$variables) {
  // ...
  $libraries = array(
    '#attached' => array(
      'library' => array(
        'seven/maintenance-page',
        'seven/install-page',
      ),
    ),
  );
  drupal_render($libraries);
}
?>
```

Bien que ça ne soit pas aussi pratique que d'inscrire directement dans le code `drupal_add_FOO()`, cela signifie que ces ressources peuvent être mises en cache pour de meilleures performances et sont facilement réutilisables parmi les différentes parties de la base du code.

Adieu Internet Explorer 6, 7 et 8

Note plus mélancolique, la dernière amélioration en date importante est qu'enfin, Drupal a pris une décision saluée par les concepteurs de sites internet du monde entier : Drupal 8 Core ne prend plus officiellement en charge IE 6, 7 et 8, ce qui donne la possibilité d'utiliser jQuery 2.0 et d'autres codes prenant en charge les navigateurs en HTML5/CSS3. (Des discussions sont également en cours pour déterminer s'il faut [arrêter de prendre en charge Android 2.3 et les versions antérieures](#) pour les mêmes raisons).



Le [html5shiv](#) (un polyfill HTML5 pour les navigateurs moins performants) a été intégré dans D8 Core en guise de cadeau d'adieu, pour qu'IE 8 et les versions antérieures ne soient pas complètement ignorées. Le projet [IE 8](#) de contribution est disponible pour ceux qui doivent absolument disposer d'une version compatible avec IE 8 pour les interfaces en version de base. Nous travaillons d'ores et déjà sur des développements frontend plus rapides, qui ne seront pas soumis à des restrictions imposées par les navigateurs vieux de 5 ans et plus. Encore une bonne nouvelle !

Quoi de plus ?

Même si Drupal 8 est presque prêt pour sa sortie, certains aspects des API ne sont pas encore déterminés (et le balisage n'est pas fixé jusqu'à la version RC). Voici quelques-uns des efforts restants à effectuer sur l'interface qui nécessiteront du code ou des revues de code :



- Bien que ça ne soit pas encore finalisé (astuce : c'est un très bon moyen de découvrir Twig), nous continuons à travailler sur la conversion de toutes les fonctions des thèmes de la version de base en template Twig pour que la création de thèmes fonctionne de la même façon, indépendamment des résultats sur la page, dans de très rares cas où les fonctions de thème sont toujours nécessaires à la performance. Cela devrait rendre la création de thème plus abordable, car cela évite aux développeurs d'apprendre le PHP pour faire de légères modifications dans presque tous les cas.
- La conversion des éléments de page (titres, tables, actions, messages) en blocs et le déplacement de la fonction du module `menu_block` dans la version de base élimineraient les variables uniques au profit d'un positionnement/d'une mise en forme (ainsi que mise en cache) des éléments de page partout en tant que blocs.
- Le mouvement Dream Markup a émergé pour que l'on retire tous les éléments superflus (les `<div>` en trop, etc.) du balisage de Drupal. Ce mouvement a montré un intérêt récent pour cette simplification à l'événement DrupalCon Austin, qui a donné lieu à une proposition pour mettre à nu le balisage par défaut de Drupal de base afin d'en enlever tous les éléments optionnels et de fournir un thème de base comprenant les classes/conteneurs et l'équivalent avec la situation actuelle. Il se passe des choses intéressantes.
- Depuis DrupalCon Austin, le groupe Headless Drupal a formulé le désir de simplifier l'utilisation complète des interfaces de personnalisation, par exemple dans Angular JS en plus de Drupal.

AMELIORATIONS POUR LES DEVELOPPEURS BACK-END

Drupal 8 vous propose de nombreuses améliorations pour les développeurs back-end, notamment une API pour configurer votre système. Toutes les entités sont désormais matérialisées comme des objets. Les mécanismes de mise en cache ont également été améliorés, de même que l'intégration grâce à des services de tiers, et de nombreuses fonctionnalités orientées web services sont intégrées. On n'arrête pas le progrès.

Un nouveau système de gestion des configurations

La mise en place dans Drupal 8 du système de gestion des configurations est sans aucun doute la nouveauté la plus attendue, à la fois par les développeurs et les concepteurs de site. Dans Drupal 7 et les versions antérieures, le contenu et la configuration étaient enregistrés dans la base de données (parfois avec un mélange des deux dans la même table), ce qui rendait le déploiement des modifications de la configuration d'un environnement à l'autre (par exemple du développement à la production) très compliqués. Toute une série de solutions ont jusqu'ici palié ce manque, notamment [hook_update_N\(\)](#), le module [Features \(Fonctionnalités\)](#) et la bonne vieille méthode qui consiste à recopier soigneusement les modifications effectuées pendant le développement sur un bout de papier et les transcrire manuellement en production. Cependant,



toutes ces solutions tentaient de contourner le problème fondamental de Drupal Core, qui n'intégrait pas le déploiement de la configuration au départ ; c'était du moins le cas avant Drupal 8.

Dans Drupal 8, toute modification de configuration (aussi bien via les formulaires standards de configuration, tels que le nom du site, [ConfigEntity](#) tel que « Views », les rôles des utilisateurs et les types de contenu) s'effectue via une API de gestion de configuration unifiée. Chaque environnement dispose d'un stockage « actif » dans lequel les paramètres de configuration sont écrits et lus à chaque chargement de page. Il dispose d'un stockage « intermédiaire » qui conserve les modifications des paramètres issues d'autres environnements qui sont sur le point d'être importés pour être vérifiées. En termes de performance, le stockage actif se trouve dans une table config de la base de données (assez analogue à la table des variables dans Drupal 7 et versions antérieures), bien que l'emplacement de stockage soit échangeable. [Le module Configuration Development](#), par exemple, écrit la configuration active sur les fichiers YAML dans le système de fichiers, ce que fait le core Drupal avec le stockage intermédiaire (staging).

Drupal 8 intègre également la possibilité d'effectuer des importations et des exportations de configurations uniques ou complètes. Il est aussi possible d'effectuer ces mêmes opérations par les commandes [config-* de Drush](#), ce qui est pratique lorsque l'on utilise des systèmes de gestion de version tels que Git.

Le déroulement classique (après avoir effectué n'importe quelle modification sur votre site Drupal 8) est :

1. Sur le site de développement, exportez la configuration « active » de votre site. Vous recevez un fichier archive (tar) comprenant de nombreux lots de fichiers YAML.
2. En production, importez les fichiers, ce qui les met dans la zone de configuration « intermédiaire ».
3. Dans l'interface de configuration, consultez la liste des paramètres de configuration qui ont été modifiés puis consultez le différentiel des modifications.
4. Si les modifications sont acceptables, importer la nouvelle configuration, ce qui remplacera le stockage actif actuel des résultats par le contenu du stockage intermédiaire ; ces derniers deviendront alors les nouvelles valeurs que Drupal utilisera pour créer des pages.



WWW.SMILE.FR



Bien sûr, certains paramètres sont propres à un environnement donné et vous ne souhaitez pas les voir déployer sur les autres environnements. Par exemple, l’horodatage qui stocke la dernière commande cron lancée. Dans ce cas, il existe une API « sœur » de la configuration API nommée « [State API](#) » pour ces paramètres plus éphémères.

Synchronize ☆

Synchronize | Single Import/Export | Full Import/Export

Home » Administration » Synchronize configuration

Import configuration the

1 changed

NAME

system.site

Import all

View changes of system.site ✕

OLD	NEW
 name: Site-Install	 name: Site-Install
 mail: admin@example.com	 mail: admin@example.com
- slogan: "	+ slogan: 'I like bananas!'
 page:	 page:
 403: "	 403: "

[Back to 'Synchronize configuration' page.](#)

Qu’en est-il du déploiement de contenu ?

Drupal 8 n’intégrant pas la prise en charge de la migration de contenus tels que les nœuds, les utilisateurs et les termes de taxonomie entre les sites (même si cela pourrait arriver dans des



versions ultérieures telles que 8.1.0 ou 8.2.0), Drupal a eu la bonne idée d'ajouter des UUID (identifiants universels uniques) à chaque contenu, tels que b2423870-b19b-45e7-8407-076aee906870. Ces UUID peuvent être utilisés pour déterminer si un contenu existe déjà sur un site donné, en évitant par la même occasion les conflits entre les identifiants numériques des contenus, ce qui rend les importations/exportations de contenu infiniment plus faciles. Gardez un œil sur le [module Deploy](#) pour avoir une [version de Drupal 8](#) qui propose cette fonctionnalité. Si vous êtes encore sur Drupal 7, vous pouvez accéder à cette fonction grâce au module [Universally Unique Identifier](#).

Des entités, partout des entités !

Le concept d'entité est une fonctionnalité essentielle de Drupal 7, il offre la possibilité d'ajouter des attributs (champs) aux contenus, mais aussi aux utilisateurs et termes de taxonomie. Cependant, l'API en place était sérieusement limitée et nécessitait l'utilisation de modules, tels que le [module Entity API](#), afin d'étoffer les fonctionnalités basiques telles que sauvegarder et supprimer une entité.

Dans Drupal 8, le système d'entités a été entièrement revu, afin non seulement de combler les oublis dans les fonctions de Drupal 7, mais également d'améliorer l'expérience des développeurs. Toutes les entités sont désormais classées en tant qu'objets, ce qui permet d'obtenir une interface de type [EntityInterface](#) standard (plus besoin de deviner quels sont les 100 « hooks » nécessaires pour la création d'entités), avec des informations intégrées sur le langage actif. Comparez et relevez les différences :

```
<?php
# Drupal 7 code.
$node->title
$node->body[$langcode][0]['value']

with

# Drupal 8 code.
$node->get('title')->value
$node->get('body')->value
?>
```

Presque tout ce que vous pouvez créer plus d'une fois a été converti en entité, ce qui assure davantage de cohérence au développement Drupal. Il existe pour cela deux sortes d'entité :



« Config entities » (entités de configuration) et « Content entities » (entités de contenu). Quelle est la différence ?

Les entités de contenu intègrent également de nouvelles fonctionnalités par rapport à Drupal 7, par exemple, les révisions sur les nœuds, mais également sur les blocs personnalisés, ainsi que la possibilité d'ajouter des commentaires à n'importe quelle entité de contenu (il est même possible de commenter des commentaires). L'article sur les « [Améliorations pour les concepteurs de site](#) » contient plus d'informations sur les autres fonctionnalités relatives aux entités.

Entités de contenu

- Champs personnalisés
- Stockées dans les tables de base de données (par défaut)
- Créées à l'aide de l'interface pour la plupart

Exemples

- Nœuds
- Blocs personnalisés
- Utilisateurs
- Commentaires
- Termes de taxonomie
- Liens vers les menus
- Agrégateurs de flux/actualités

Entités de configuration

- Déployées sur différents environnements
- Stockées dans le système de configuration
- Le plus souvent créées via le Backoffice

Exemples

- Types de contenu
- Types de blocs personnalisés
- Rôles des utilisateurs
- Vues
- Vocabulaires de taxonomie
- Menus
- Styles d'images

Un hook_schema() en perdition ?



Qu'est-ce que cela signifie pour vous en tant que développeur ? Cela signifie qu'entre l'« Entity API » et la « Configuration/State API », il n'y aura quasiment plus de raison de créer et de gérer manuellement vos propres tables en base de données dans Drupal 8. En utilisant les API standard, vous profitez d'une écriture de codes plus compacts et bénéficiez de la portabilité vers d'autres bases de données telles que MongoDB.

Web Services

Une des fonctionnalités majeure de Drupal 8 est la mise en place d'une API REST fournie par la suite de modules [RESTful Web Services](#). Cette brique supplémentaire permet à Drupal de fonctionner de paire avec des applications mobiles, de faciliter les échanges entre sites mais aussi d'améliorer l'intégration d'applications tierces. Plus concrètement, il sera possible d'exposer les ressources disponibles (nodes, taxonomy, users, etc.), les méthodes HTTP qui seront autorisées par rapport à ces ressources (par exemple GET, POST, PATCH, DELETE) et les formats (JSON) et authentifications qui sont utilisés pour accéder à ces ressources. Consultez le [module de contribution REST UI](#) qui fournit une interface pour cette configuration. Pour chaque méthode HTTP autorisée, vous pouvez configurer les autorisations d'accès aux ressources, selon les rôles de votre site. Cela permet par exemple aux utilisateurs anonymes de « recevoir » (méthode GET), mais seuls les administrateurs peuvent « publier » (méthode POST).

Une fois que les différents modules de web services « RESTful » sont correctement configurés, vous pouvez accéder à bon nombre de données, dans un format exploitable, qui représentent le contenu de votre site, tel que :

```
...
  [title] => Array
    (
      [0] => Array
        (
          [value] => Hello, world!
          [lang] => en
        )
    )
...
[body] => Array
  (
    [0] => Array
      (
        [value] => <p>This is my <strong>awesome
        </strong> article.</p>
        [format] => basic_html
        [summary] =>
      )
    )
  )
...

```

À quoi est-ce que ça peut bien servir ? À plein de choses ! Voici un exemple d'informations de Drupal 8 récupérées au format JSON et qui s'affichent dans une [application mobile jQuery](#) autonome.

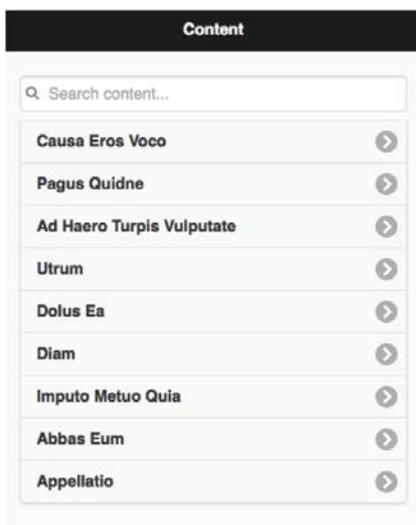


Drupal 8 intègre une nouvelle bibliothèque appelée [Guzzle](#) caractérisée par une syntaxe simple pour récupérer et poster des données sur Drupal ou pour échanger avec des web services tiers tels que Twitter ou Github.

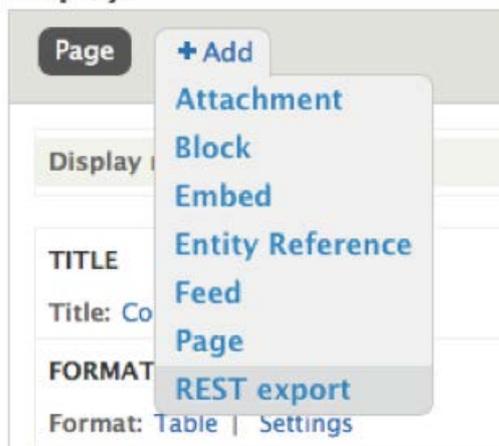
La possibilité d'ajouter un affichage « REST export » dans n'importe quelle vue constitue une autre fonctionnalité de web services proposée dans Drupal 8.

Cela signifie que vous pouvez aisément créer des flux JSON ou XML de contenus dynamiques et personnalisables à partir de votre site Drupal. Il suffit pour cela de les sélectionner ensemble en cliquant dessus !

WWW.SMILE.FR



Displays





Mise en cache améliorée

Pour finir, la mise en cache dans Drupal 8 a été améliorée.

- Le module [Entity cache](#) (Mise en cache d'entité) est désormais inclus dans le core
- Les « [cache tags](#) » permettent de vider le cache de façon plus précise lorsque le contenu ou les paramètres sont mis à jour sur le site
- Toutes les fonctionnalités de cache telles que l'agrégation en CSS/JS sont activées [par défaut](#), ce qui rend Drupal 8 rapide dès la première utilisation.

Même si nous travaillons toujours sur les [améliorations globales des performances de D8](#), ces améliorations devraient permettre un chargement extrêmement rapide des pages.

DES AMÉLIORATIONS EN
PROFONDEUR DÈS LA
VERSION CORE

Drupal 8 a effectué des évolutions majeures de son API tout en intégrant ce qui se fait de mieux dans l'écosystème PHP.

« Proudly Found Elsewhere » (Fièrement trouvé ailleurs)

Le concept du « Proudly found elsewhere » témoigne d'un changement dans la façon de penser des développeurs Drupal. En effet, trouver les meilleurs outils pour accomplir une certaine tâche et les incorporer au logiciel, au lieu de créer une application particulière appartenant en propre à Drupal, a été le leitmotiv des développeurs.

Vous pouvez constater ce changement de philosophie dans de nombreux composants de Drupal 8. Parmi les bibliothèques externes que nous avons inclus se trouvent [PHPUnit](#) pour le test unitaire, [Guzzle](#) afin d'effectuer des requêtes HTTP (vers des web services), plusieurs composants de [Symfony](#) (le tutoriel « [Create your own framework on top of the Symfony2 Components](#) » est excellent pour en apprendre plus sur ces composants) et [Composer](#) comme gestionnaire de dépendances externes et de chargement automatique de classes, entre autres.

Mais ce changement de philosophie se constate aussi dans la base du code lui-même. Nous avons effectué [d'importantes modifications de l'architecture](#) dans Drupal 8 afin de nous adapter à la façon dont on organise le code dans l'écosystème PHP : une programmation découplée, orientée objet (OO) et utilisant les éléments du langage PHP dans ses versions les plus récentes, telles que les espaces de nom et les traits.



LA POO A PORTEE DE MAIN

Prenons quelques exemples d'extraits de codes pour illustrer dans les faits l'architecture « Proudly Found Elsewhere » de Drupal 8.

Drupal 7 : example.info

```
name = Example
description = An example module.
core = 7.x
files[] = example.test
dependencies[] = user
```

Tous les modules dans Drupal nécessitent un fichier .info pour pouvoir être enregistrés dans le système. L'exemple ci-dessus est typique d'un module de Drupal 7. Le format du fichier ressemble à un fichier « INI » facilement éditable, mais il comprend également des « Drupalismes » tels que la syntaxe en `array[]` qui empêche l'utilisation de fonctions standard PHP de lecture/écriture des fichiers INI. La clé `files[]`, qui déclenche le chargement automatique des classes personnalisées pour ajouter du code dans la base de registre, est particulièrement « drupalienne », et les développeurs de modules qui codent en orienté objet doivent ajouter une ligne `files[]` pour chaque fichier qui définit une classe, ce qui [peut devenir stupide](#).

Drupal 8 : example.info.yml

```
name: Example
description: An example module.
core: 8.x
dependencies:
  - user
# Note: New property required as of Drupal 8!
type: module
```

Dans la droite ligne du concept « Proudly Found Elsewhere », les fichiers infos dans Drupal 8 sont désormais de simples fichiers [YAML](#), les mêmes qui sont utilisés par d'autres langages et dans d'autres framework. La syntaxe est très similaire (principalement : à la place de =, et des tableaux formatés différemment), et il est toujours aussi facile de lire et d'écrire dans ces fichiers. La clé gênante `files[]` a été supprimée et remplacée par la norme [PSR-4](#) pour l'auto-chargement des classes par le biais de [Composer](#). En d'autres termes, c'est en suivant une convention spécifique sur le nommage/dossier de la classe (`modules/example/src/ExampleClass.php`), que Drupal peut charger automatiquement le code orienté objet sans nécessiter un enregistrement manuel.



Drupal 7: hook_menu()

example.module

```
<?php
/**
 * Implements hook_menu().
 */
function example_menu() {
  $items['hello'] = array(
    'title' => 'Hello world',
    'page callback' => '_example_page',
    'access callback' => 'user_access',
    'access arguments' => 'access content',
    'type' => MENU_CALLBACK,
  );
  return $items;
}

/**
 * Page callback: greets the user.
 */
function _example_page() {
  return array('#markup' => t('Hello world.'));
}
?>
```

Il s'agit d'un module assez basique dans Drupal 7 qui définit une URL sur /hello qui, lorsqu'on y accède, fait des vérifications pour s'assurer que l'utilisateur dispose des permissions d'« accès au contenu » avant d'appeler la fonction `_example_page()`, qui affiche « Hello world. » à l'écran en tant que page entièrement chartée. Le `hook_menu()` est un exemple de ce qui est connu de façon péjorative sous le nom de ArrayPI, qui est un modèle d'architecture courant dans Drupal 7 et les versions antérieures. Le problème posé par les ArrayPI est qu'ils sont difficiles à taper (par exemple, n'avez-vous jamais oublié le `return $items`, puis passé 30 minutes à chercher comment résoudre le problème?), ils ne disposent pas d'auto-complétion dans l'EDI (environnement de développement intégré) pour les propriétés disponibles, enfin la documentation doit être mise à jour manuellement à chaque fois qu'une clé est modifiée ou ajoutée. [La documentation sur hook_menu\(\)](#) montre que le défaut vient également du fait que trop de choses essaient d'être faites à la fois. Ce programme est utilisé pour enregistrer les associations/accès entre les chemins et les pages, mais également pour afficher les liens dans l'interface de plusieurs façons, changer de thème, et bien plus.

Drupal 8: Routes + contrôleurs

example.routing.yml



```
example.hello:  
  path: '/hello'  
  defaults:  
    _content: '\Drupal\example\Controller\Hello::content'  
  requirements:  
    _permission: 'access content'
```

src/Controller/Hello.php

```
<?php  
namespace Drupal\example\Controller;  
  
use Drupal\Core\Controller\ControllerBase;  
  
/**  
 * Greet the user.  
 */  
class Hello extends ControllerBase {  
  public function content() {  
    return array('#markup' => $this->t('Hello world.'));  
  }  
}  
?>
```

Dans le nouveau [système de routage](#) de Drupal 8, la création des routes chemin/accès est maintenant assurée par un fichier YAML qui utilise la même syntaxe que le [système de routage Symfony](#). La logique de « fonction de callback » est désormais assurée par une classe « Controller » (comme dans le modèle standard [modèle-vue-contrôleur](#)) dans un fichier nommé de façon spécifique, conformément à la norme PSR-4. Dans le module exemple ci-dessus, le « controller » est déclaré dans son propre espace de nom pour permettre au module de nommer ses classes comme il le souhaite, sans se soucier d'éventuels conflits avec d'autres modules.

Finalement, la classe hérite de la logique de la classe `ControllerBase` donnant ainsi au contrôleur Hello l'accès à toutes les méthodes et capacités du `ControllerBase`, telles que `$this->t()` (la façon orientée objet de nommer la fonction `t()`). Étant donné que `ControllerBase` est une classe conforme à la norme PHP, toutes ses méthodes et propriétés seront complétées automatiquement dans les EDI ; vous n'aurez donc pas à deviner ce qu'il peut ou ne peut pas faire pour vous.



Drupal 7: hook_block_X()

block.module

```
<?php
/**
 * Implements hook_block_info().
 */
function example_block_info() {
  $blocks['example'] = array(
    'info' => t('Example block'),
  );
  return $blocks;
}

/**
 * Implements hook_block_view().
 */
function example_block_view($delta = '') {
  $block = array();
  switch ($delta) {
    case 'example':
      $block['subject'] = t('Example block');
      $block['content'] = array(
        'hello' => array(
          '#markup' => t('Hello world'),
        ),
      );
      break;
  }
  return $block;
}
?>
```

Voici un exemple typique de la façon dont on définit un nouveau composant dans Drupal (blocs, effets d'image formats de texte, etc.) : une sorte de « hook » _info déclaratif, associé à un ou plusieurs autres « hooks » pour exécuter d'autres actions (voir, appliquer, configurer et bien plus). En plus de ces éléments, qui sont pour la plupart des ArrayPI, on constate qu'il y a des API bien plus difficiles à prendre en main, car l'API générale elle-même ne peut absolument pas être découverte, à moins d'inspecter rigoureusement tous les [fichiers .api.php](#) des différents modules à supposer qu'ils existent, ce qui n'est pas garanti) pour découvrir les fonctions « hooks » nommées par « magie » dont vous avez besoin pour définir l'application de tel ou tel comportement. Certains sont nécessaires, d'autres non. Parviendrez-vous à deviner lequel est lequel ?

Drupal 8 : Les blocs (et bien d'autres éléments) en tant que plugins

Dans Drupal 8, ces API mystères sont en majorité transférées vers le nouveau [système de plugin](#), qui ressemble à ce qui suit :

src/Plugin/Block/Example.php



```
<?php
namespace Drupal\example\Plugin\Block;

use Drupal\Core\Block\BlockBase;

/**
 * Provides the Example block.
 *
 * @Block(
 *   id = "example",
 *   admin_label = @Translation("Example block")
 * )
 */
class Example extends BlockBase {
  public function build() {
    return array('hello' => array(
      '#markup' => $this->t('Hello world.')
    ));
  }
}
?>
```

Cet exemple ressemble beaucoup à celui du contrôleur ; un plugin est une classe qui, dans le cas présent, s'étend à partir d'une classe de base ([BlockBase](#)) qui prend soin d'implémenter certains mécanismes pour vous. L'API des blocks elle-même est déclarée dans l'interface [BlockPluginInterface](#) et implémenté par la classe BlockBase.

Vous remarquerez qu'en règle générale, les interfaces exposent et renseignent plusieurs API de façon à ce que l'EDI soit découverte et agréable à utiliser. Le meilleur moyen de découvrir les nouvelles API de Drupal 8 consiste à naviguer parmi les interfaces proposées.

Les commentaires au-dessus des classes sont appelés « annotations ». À première vue, l'utilisation des commentaires PHP pour spécifier des métadonnées qui affectent la logique applicative semble étrange, mais cette technique est désormais largement utilisée par de nombreuses bibliothèques modernes en PHP et acceptée par la communauté. Cela offre l'avantage de conserver les métadonnées de classe dans le même fichier et juste à côté de la définition de la classe.

Drupal 7 : « Hooks »

Dans Drupal 7 et ses versions antérieures, le mécanisme d'extension utilisé est le concept de « [hooks](#) ». En tant que créateur d'une API, vous pouvez déclarer un « hook » utilisant des fonctions telles que `module_invoke_all()`, `module_implements()`, `drupal_alter()`, etc. Par exemple :



```
<?php
// Compile a list of permissions from all modules for
// display on admin form.
foreach (module_implements('permission') as $module) {
    $modules[$module] = $module_info[$module]['name'];
}
?>
```

Si vous voulez qu'un module réponde à cet événement, il vous faut créer une fonction nommée `modulename_hookname()` et déclarer ses résultats d'une façon reconnaissable par l'utilisation d'une fonction « hook ». Par exemple :

```
<?php
/**
 * Implements hook_permission().
 */
function menu_permission() {
    return array(
        'administer menu' => array(
            'title' => t('Administer menus and menu items'),
        ),
    );
}
?>
```

Bien qu'il s'agisse d'un mécanisme astucieux, ce choix d'architecture est surtout dû à l'ancienneté de Drupal (Drupal a démarré en 2001, à l'époque où le PHP3 battait son plein et où le code orienté objet et autres n'étaient pas encore plébiscités), certaines choses ne sont pas si évidentes :

- Ce mécanisme d'extension qui « nomme une fonction d'une façon particulière » tient vraiment du drupalisme, et les développeurs qui s'initient à Drupal peinent à le comprendre au début
- Au moins quatre fonctions différentes peuvent lancer un « hook » : `module_invoke()`, `module_invoke_all()`, `module_implements()`, `drupal_alter()` et bien d'autres encore. Cela rend les extensions disponibles très difficiles à trouver dans Drupal.
- Il n'y a pas de cohérence entre ce qu'attendent les « hooks ». Certains « hooks » sont de type info et requièrent un tableau (parfois un tableau de tableau de tableau de tableau), d'autres sont des « hooks » de type info qui ne répondent qu'en cas d'événement particulier, par exemple, le lancement d'une tâche cron ou la sauvegarde d'un nœud. Vous devez lire la documentation relative à chaque « hook » pour comprendre quelles entrées et quelles sorties ils attendent.



Drupal 8 : Événements (« Events »)

Bien que les « hooks » dominent encore largement dans Drupal 8 dans le cas de comportements commandés par événements (les « hooks de type info ont été en grande partie transformés en annotations YAML ou en plugins), les portions de Drupal 8 compatibles Symfony (par exemple bootstrap/exit, système de routage) ont pour la plupart été transférées dans le [système Event Dispatcher de Symfony](#). Dans ce système, les événements sont distribués lors de l'exécution lorsque certaines logiques sont en cours, et les modules peuvent « abonner » leurs classes aux événements auxquels ils veulent qu'elles réagissent.

Pour le prouver, regardons de plus près la configuration API de Drupal 8, qui est stockée dans Core/lib/Drupal/Core/Config/Config.php. Elle définit de nombreuses méthodes CRUD telles que save(), delete(), etc. Chaque méthode déclenche un événement lorsque sa tâche est accomplie, ce qui permet aux autres modules de réagir. Voici par exemple Config::save() :

```
<?php
public function save() {
    // <snip>Validate the incoming information.</snip>

    // Save data to Drupal, then tell other modules this was
    // just done so they can react.
    $this->storage->write($this->name, $this->data);
    // ConfigCrudEvent is a class that extends from Symfony's
    // "Event" base class.
    $this->eventDispatcher->dispatch(ConfigEvents::SAVE, new
    ConfigCrudEvent($this));
}
?>
```

Lorsque le processus s'arrête, il y a au moins un module qui a besoin de réagir lorsque la configuration est sauvegardée : le module du core Language. Car si le paramètre de configuration qui vient d'être modifié était la langue du site par défaut, les fichiers PHP compilés doivent être rechargés pour que la modification puisse être effective.

Pour ce faire, le module Language accomplit trois tâches :

1. Il enregistre une classe éligible à un événement dans son fichier language.services.yml (il s'agit d'un [fichier de configuration du Service Container de Symfony](#) pour l'enregistrement des codes réutilisables)

```
language.config_subscriber:
  class: Drupal\language\EventSubscriber\ConfigSubscriber
  tags:
    - { name: event_subscriber }
```



2. Dans la [classe référencée](#), il exécute le [EventSubscriberInterface](#) et déclare une méthode `getSubscribedEvents()` qui énumère les événements pour lesquels il devrait être alerté. Il fournit également pour chaque événement une ou plusieurs fonctions de callback qui doivent être lancées lorsque l'événement débute :

```
<?php
class ConfigSubscriber implements EventSubscriberInterface {
    static function getSubscribedEvents() {
        $events[ConfigEvents::SAVE] [] = array('onConfigSave', 0);
        return $events;
    }
}
?>
```

3. Il définit la méthode de callback qui contient le code à exécuter lorsque la configuration est sauvegardée :

```
<?php
public function onConfigSave(ConfigCrudEvent $event) {
    $saved_config = $event->getConfig();
    if ($saved_config->getName() == 'system.site' &&
        $event->isChanged('langcode')) {
        // Trigger a container rebuild on the next request.
        PhpStorageFactory::get('service_container')
            ->deleteAll();
    }
}
?>
```

Pour couronner le tout, nous avons intégré un utilitaire de registre plus explicite pour qu'un seul module puisse souscrire plusieurs classes à des événements individuels. Cela nous permet d'éviter les situations du passé où il y avait des déclarations conditionnelles (de type switch) dans les « hooks » ou encore de nombreux blocs de code sans rapport et se gênant les uns les autres. Au lieu de cela, on a la possibilité de séparer la logique en plusieurs classes séparées et distinctes. Cela signifie également que notre logique d'événement est chargée au moment où elle doit être exécutée, et qu'elle ne monopolise pas constamment de la mémoire PHP.

Il est également assez simple de déboguer les événements et de tracer leur exécution. Au lieu d'avoir toute une flopée de fonctions PHP, de procédures qui peuvent ou non avoir été utilisées pour invoquer votre « hook », c'est le même « Event Dispatcher » qui est utilisé dans tout le système. De plus, trouver les exécutions est aussi simple que d'exécuter une commande `grep` pour trouver la bonne constante de classe, par exemple : `ConfigEvents::SAVE`.



Le système d'événements complète logiquement la transition vers une méthode orientée objet. Les plugins prennent en charge les « hooks » de type info ainsi que les « hooks » invoqués suite à un autre « hook » info. La plupart de nos anciens systèmes sont remplacés par des systèmes de registre explicites en YAML. Le système d'événements remplace les « hooks » de type événement et intègre une méthodologie de souscription performante capable d'étendre des fonctionnalités du core.

...et beaucoup, beaucoup plus !

Vous pourrez trouver une bonne introduction aux modifications de l'API de Drupal 8 sur [le site api.drupal.org](http://le.site.api.drupal.org) qui contient également une liste des sujets groupés qui pourront vous guider dans Drupal 8.

WWW.SMILE.FR

The screenshot shows the Drupal API reference page for Drupal 8. The page has a blue header with the text 'Drupal API' and navigation links for 'Drupal 6', 'Drupal 7', 'Drupal 8', 'Other projects', and 'Develop for Drupal'. Below the header, there is a section for 'API reference' with a breadcrumb trail: 'Drupal 4.6', 'Drupal 4.7', 'Drupal 5', 'Drupal 6', 'Drupal 7', and 'Drupal 8'. The main content area includes a welcome message, a search box for 'Search Drupal 8', and a navigation menu with categories like 'Topics', 'Classes', 'Functions', 'Files', 'Namespaces', 'Services', 'Constants', 'Globals', and 'Deprecated'. The page also lists 'Essential background concepts', 'User interface', and 'Storing and retrieving data' with sub-topics.

Vous pouvez aussi vous rendre sur <https://drupal.org/list-changes> pour accéder à la liste complète des modifications de l'API entre Drupal 7 et Drupal 8. Toutes les modifications enregistrées de l'API comprennent des exemples de codes avant/après pour faciliter votre migration, ainsi que des conseils qui vous expliquent les modifications et leurs raisons.



Change records for Drupal core

Published Draft Review

Add new change record

Keywords Introduced in branch Introduced in version

Change node created is greater than or equal to

Impacts
Site builders, administrators, editors
Module developers
Themers

Introduced in branch/version	Notice created	Change
8.x	21-Mar-2014	REST URI paths changed to canonical paths
8.x	17-Mar-2014	Plugins can depend on a module
8.x	17-Mar-2014	.module and .profile files are no longer required; ModuleHandler::getModuleList() now returns Extension objects
8.x / 8.x-dev	14-Mar-2014	Distribution level settings added to install profiles
8.x	13-Mar-2014	drupal_load() has been removed
8.x	12-Mar-2014	Shortened directory structure for some plugin types
8.x	12-Mar-2014	'show' variable removed from admin_block theme hook
8.x	10-Mar-2014	COMMENT_HIDDEN & COMMENT_CLOSED & COMMENT_OPEN converted to constants on CommentItemInterface
8.x / 8.x	10-Mar-2014	user_authenticate() has been replaced by a 'user.auth' service
8.x	07-Mar-2014	show and hide functions removed from Twig in favor of a new "without" Twig filter
8.x	07-Mar-2014	drupal_get_filename() always returns pathname of main extension file
8.x	06-Mar-2014	drupal_implode_tags() and drupal_explode_tags() replaced with Drupal\Component\Utility\Tags class
8.x	05-Mar-2014	PluginBags have been moved to Drupal\Core
8.x / 8.0-alpha9	04-Mar-2014	Added ability to define fields and field alterations for specific entity bundles

Les modules...

Il est vrai que l'évolution vers un code moderne et orienté objet implique en général davantage de texte que les codes de procédure. Pour vous aider à franchir ces obstacles, consultez les projets suivants :

- [Drupal Module Upgrader](#) : Si vous cherchez un moyen d'exporter vos modules de Drupal 7 à Drupal 8, n'allez pas plus loin que ce projet. Il peut vous dire ce dont vous avez besoin de modifier (avec des conseils pour signaler les modifications pertinentes) ou alors convertir automatiquement votre code pour Drupal 8. Vous pouvez en apprendre davantage sur le module DMU en visionnant cette [interview en podcast avec la personne en charge de la maintenance](#).
- [Console](#) : Pour les nouveaux modules, ce projet est un générateur de code qui génèrera automatiquement des fichiers .module/.info, des structures de répertoire en PSR-4, du YAML, des registres de classe pour les routes, entre autres !
- La plupart des développeurs de Drupal Core ne jure que par l'EDI [PhpStorm](#), et la dernière version comprend de [très nombreuses fonctionnalités intéressantes pour les développeurs](#). Si vous faites partie des meilleurs contributeurs de l'écosystème de Drupal, vous pouvez l'obtenir gratuitement ! (Veuillez remarquer qu'il ne s'agit pas d'un placement de produit. Vous pouvez rejoindre #drupal-contribute à n'importe quelle heure du jour ou de la nuit et constater s'il peut s'écouler une heure sans que quelqu'un mentionne PhpStorm).



FOIRE AUX QUESTIONS

À quand la sortie de Drupal 8 ? Pourquoi passer à la version supérieure ? Je me pose beaucoup de questions.

Pourquoi devrais-je m'intéresser à Drupal 8 ?

Drupal a été conçu d'abord et avant tout comme un outil pour les développeurs et propose tout un ensemble d'API (interfaces de programmation) permettant de concevoir des éléments de site internet à base de codes, tels que des formulaires d'insertion de contenu, des pages d'administration et des blocs en barre latérale. Dans les différentes versions de Drupal, et en particulier dans Drupal 7, l'accent avait été mis sur l'accessibilité de Drupal aux utilisateurs moins techniques, c'est pourquoi Drupal proposait des interfaces utilisateur pour les tâches fondamentales (installation, modélisation de données, architecture d'information, pages de renvoi, etc.). Aujourd'hui, la plupart des sites de Drupal téléchargent et configurent de nombreux modules de contribution pour accéder à des fonctionnalités telles que l'éditeur WYSIWYG, les « Views » (Affichages), etc. Et grâce à cette combinaison du core et des extensions, Drupal fait fonctionner [certains des sites les plus importants sur Internet](#).

Drupal 8 s'inscrit dans la droite ligne de Drupal 7, et intègre par défaut des fonctionnalités très attendues, telles que les améliorations de l'expérience de création, les fonctionnalités multilingues complètes, ainsi que de nombreuses fonctionnalités pour les concepteurs de site. Drupal 8 est plus en accord avec ce qui se fait aujourd'hui sur Internet grâce à une approche mobile et une interface réorganisée. Et, fidèle à ses origines de développement, il propose de nombreuses fonctionnalités pour l'interface de programmation et une base de code modernisée et orientée objet. Globalement, Drupal 8 est une version bien plus performante, qui offre des possibilités aux créateurs de contenus, aux concepteurs de site, aux développeurs ainsi qu'aux designers. Ce logiciel est conçu pour s'adapter aux technologies à venir, de façon à pouvoir servir de base solide à vos projets, quels que soient les technologies, les appareils et les services qui seront édités par la suite.

Cela étant dit, Drupal 7 est une plateforme stable, fiable et mature qui vous servira encore plusieurs années à venir. Et de nombreuses fonctionnalités de Drupal 8 sont disponibles sous une certaine forme pour Drupal 7. Drupal 8 sera excellent, mais c'est aussi le cas de Drupal 7. De toute façon, c'est toujours une bonne chose de commencer à découvrir Drupal 8 dès maintenant, vous serez ainsi familiarisé avec ce dernier lorsqu'il répondra aux besoins de vos projets futurs.



Drupal 8 a vraiment l'air génial ! Comment se fait-il qu'il ne soit pas encore sorti ?

Plus que quelques semaines à attendre ! Lorsque les tous derniers [problèmes critiques](#) (bug et tâches) auront été résolus, une version « release candidate » de Drupal sera créée. Une fois que cette version publiée ne comportera plus de problèmes critiques signalés, Drupal 8.0.0 sera étiqueté et mis sur le marché sous les applaudissements !

Vous pouvez voir si la sortie de Drupal est imminente à n'importe quel moment en vous rendant sur <https://www.drupal.org/drupal-8.0/get-involved>. Le bloc dans la barre latérale indique le nombre de problèmes critiques qu'il reste à résoudre.

Drupal 8 Critical Countdown

Drupal 8 enters release candidate phase once there are zero critical issues.

87 critical issues remaining

Read more about [how to help Drupal 8!](#)

Que se passera-t-il une fois que Drupal 8 sera sorti ?

En commençant avec Drupal 8.0.0, le projet Drupal [se dirigera peu à peu vers une nouvelle phase de sortie](#), qui en plus des résolutions des bugs et des versions de sécurité habituelles par mois (8.0.1, 8.0.2...) inclura des sorties de versions mineures par semestre (8.1.0, 8.2.0, etc.). Ces versions pourront comprendre de nouvelles fonctionnalités, des améliorations de l'API compatibles avec les versions antérieures, et bien plus. Après la sortie de plusieurs versions mineures, une version supportée sur le long terme (LTS) de Drupal 8 sera créée et la conception de Drupal 9 pourra commencer.

Cela signifie que les adeptes de Drupal ne devront plus attendre pendant *N* années avant d'avoir de nouvelles fonctionnalités ; nous pouvons itérer sur les fonctionnalités et les API tous les mois jusqu'à ce que la plateforme ait atteint sa maturité. Cela signifie également que ceux qui sont plus prudents et qui veulent de la stabilité plutôt que du clinquant peuvent se reposer sur les versions LTS et ne changer de version qu'une fois en quelques années (et même ne pas passer par les versions majeures). Hourra !

Quand pourrai-je vraiment commencer à utiliser Drupal 8 ?

La réponse dépend de qui vous êtes :



- Si vous êtes un développeur de modules, vous pouvez vous intéresser à Drupal 8 dès maintenant. Il est toujours possible de fournir un retour sur les API et Drupal 8 intègrera tout ce dont vous avez besoin pour pouvoir accéder à vos projets. Mais gardez en tête que certaines des API de Drupal 8 pourront être modifiées avant sa sortie en vue de résoudre certains problèmes critiques, il est donc possible que vous deviez faire des ajustements de code après la version RC.
- Si vous êtes un auteur de documentation, un traducteur ou un concepteur, vous remarquerez que l'interface utilisateur, le texte de l'interface et le balisage dans Drupal 8 ne seront pas finalisés avant la sortie de la version RC, il vous faudra donc attendre la version RC1 pour vous concentrer entièrement sur les documentations, les traductions ou les thèmes qui s'affichent pour l'utilisateur.
- Si vous êtes un utilisateur de la première heure de Drupal avec des développeurs dans une équipe qui n'ont pas peur d'accéder aux modules et de résoudre les bugs pendant l'utilisation, et qu'ils ont une date de lancement fixée au plus tard à 2015 ou 2016, vous pourrez commencer à concevoir vos sites D8 une fois que Drupal 8 sera disponible dans sa version bêta ou RC. Cela pourrait surtout être une bonne idée si vous avez besoin des certaines fonctionnalités proposées par Drupal 8.
- La plupart des utilisateurs voudront utiliser Drupal 8 quelques mois après sa sortie, lorsque plusieurs modules de contribution seront accessibles. Regardez attentivement le [graphique de l'utilisation du projet Drupal](#). Lorsque la courbe de D7 et celle de D8 se croisent, c'est que c'est probablement le bon moment pour vous pour sauter le pas car cela signifie qu'il y a plus d'utilisateurs de D8 que de D7 et que tout le travail a déjà été effectué pour vous.

Que puis-je faire en attendant ?

Utiliser Drupal 7. Drupal 7 est stable, mature, fiable, puissant et est doté d'un système bien soutenu qui sera maintenu grâce aux résolutions des bugs jusqu'après la sortie de la version LTS de Drupal 8, et sera soutenu grâce aux résolutions de sécurité jusqu'à la sortie de la version LTS de Drupal 9 (dans plusieurs années). Un certain nombre des excellentes fonctionnalités de Drupal 8 sont également disponibles dans Drupal 7 avec les modules de contribution.

Fonction de Drupal 8 Core	Équivalent communautaire dans Drupal 7
WYSIWYG	CKEditor : https://drupal.org/project/ckeditor
Édition sur place	Quick Edit : https://drupal.org/project/quickedit
Barre d'outils responsive	Mobile Friendly Navigation Toolbar : https://drupal.org/project/navbar
Thème HTML5 responsive	Thèmes de base Omega, Zen, Adaptive, Aurora, etc.



Thème admin responsive	Ember : https://drupal.org/project/ember
Images responsive	Picture : https://drupal.org/project/picture
Tableaux responsive	Responsive Tables : https://drupal.org/project/responsive_tables
Passage du Backoffice au Frontoffice	Escape Admin : https://drupal.org/project/escape_admin
Multilingue	Internationalization: https://drupal.org/project/i18n Entity Translation : https://drupal.org/project/entity_translation (et de nombreux autres modules)
Blocs comme des entités	Bean : https://drupal.org/project/bean
Gestion de la configuration	Features : https://drupal.org/project/features (fournit des fichiers exportables qui peuvent être utilisés dans le déploiement)
Web services	RESTful Web Services : https://drupal.org/project/restws

Et pour la migration de mon site ?

C'est une question essentielle !

- Pour le contenu de votre site (utilisateurs, articles, etc.) et pour de nombreux paramètres de configuration (variables, paramètres des blocs, etc.), Drupal 8 vous fournira une mise à niveau pour Drupal 6 (déjà dans le core) et Drupal 7 (actuellement en cours de conception) vers Drupal 8 qui comprendra les modules du core (Les modules de contribution et personnalisés devront écrire leur propre [migration](#) pour couvrir leurs données). En fait, vous continuerez de faire tourner votre site sur Drupal 6/7 tandis que vous concevrez votre site Drupal 8, puis vous lancerez un script similaire au script actuel update.php pour transférer ses contenus. Lorsque tout vous semblera en ordre, faites la bascule. Presque sans aucun temps de latence !
- Pour les modules de contribution de votre site, téléchargez et installez la version 7.x du module [Upgrade Status \(Migration de module\)](#), qui affiche une vue pratique du site de votre module et le statut actuel du transfert vers D8.



- Pour les modules personnalisés de votre site, vous devrez vous-même effectuer leur transfert. Le projet [Drupal Module Upgrader](#) peut vous aider à automatiser certains de ces processus et à générer un rapport sur les autres choses à modifier. (Il n'est cependant pas omniscient, il vous faudra donc quand même résoudre manuellement certaines choses).
- Pour le thème personnalisé de votre site, qui devra être converti en Twig, consultez le projet [Twigifier](#), qui essaie d'automatiser une bonne partie de ce travail.

Donc, en résumé, la mise à niveau dépend largement des spécificités de votre site et de la façon dont il est structuré. En général, il sera plus simple pour vous de faire le transfert vers Drupal 8 si vous choisissez bien des modules de contribution plutôt que des codes personnalisés. Prévoyez la conception de votre site actuel en conséquence.

Pour d'autres astuces pour que votre site Drupal 6/7 soit prêt à accueillir Drupal 8, rendez-vous sur www.acquia.com/blog/getting-your-site-ready-drupal-8.

Comment apporter mon aide ?

Plusieurs options s'offrent à vous pour contribuer à Drupal 8 :

- La façon la plus rapide est de nous [aider à résoudre les problèmes critiques](#). Gardez un œil attentif sur [Drupal Core Updates](#), qui contient toujours les dernières actualités qui nécessitent une attention particulière.
- Si vous découvrez seulement le développement de Drupal Core, ou que vous souhaitez avoir des conseils de la part d'une personne sur les choses utiles sur lesquelles travailler, consultez les [Core mentoring hours](#) (heures de monitorat sur Core) qui ont lieu deux fois par semaine sur IRC.
- Vous souhaitez apporter votre aide pour le chemin de migration de Drupal ? Consultez [l'équipe IMP \(Migrate dans Core\)](#).
- Vous souhaitez apporter votre aide pour la documentation de Drupal 8 ? Consultez les [priorités actuelles de la documentation](#).
- Vous voulez découvrir les API de Drupal 8 et aider les autres développeurs dans le processus ? Aidez-nous à transférer les [exemples pour les développeurs](#) vers Drupal 8.
- Vous voulez gagner du temps, pour vous et les autres, lors du transfert des modules ? Aidez-nous à écrire les routines pour [Drupal Module Upgrader](#).



REMERCIEMENTS

Nous pouvons applaudir virtuellement les plus de 2 700 contributeurs de Drupal 8 jusque-là ! Vous pouvez les [rejoindre dès maintenant !](#)

Nous remercions également les personnes ayant contribué à la rédaction et la relecture de ce livre blanc :

- Angela Byron, Director, Community Engagement, Acquia
- Grégory Bécue, Directeur Associé, Smile
- Jared Betro, Chargé d'Affaires, Smile
- Alan Moreau, Expert technique Drupal, Smile
- Leslie Martin, Chargée de Communication, Smile
- Emmanuel Quedville, Consultant externe.

N'hésitez pas à nous transmettre **vos avis et évaluations** sur ce livre blanc.
Une seule adresse : contact@smile.fr

Pour en savoir plus, nous vous invitons à consulter le **portail d'information sur Drupal 8** : www.drupal8-guide.com. Vous y trouverez des actualités, la roadmap, une sélection de modules, des explications sur la migration...

Vous souhaitez vous former ou former vos équipes à Drupal 8 ?
N'hésitez pas à contacter **Smile Training** ! Cours sur-mesure, inter-entreprise, cours particuliers ou séminaires : Smile Training, organisme agréé, est le leader de la formation open source !
Rendez-vous sur : <http://training.smile.eu/>