

WordPress 7.0 intègre l'IA dans son cœur : une révolution pratique, une question de souveraineté
WordPress 7.0 intègre l'IA dans son cœur : une révolution pratique, une question de souveraineté

□ Bernard Sfez - 2026-07-15 19:07



Avec sa version 7.0, « Armstrong », WordPress fait entrer l'intelligence artificielle directement dans son cœur. La promesse est séduisante : brancher une IA sur son contenu en quelques clics. Mais ce chemin par défaut envoie votre savoir vers des modèles hébergés par des tiers.

Cette simplicité a un prix que peu d'utilisateurs mesurent. Contenus internes, fichiers clients, notes stratégiques. Tout ce que vous confiez à une IA hébergée peut être journalisé, réutilisé ou soumis à une juridiction étrangère. Il existe pourtant une autre voie, souveraine par conception, et nous en parlerons à la fin de cet article.

Ce que change réellement WordPress 7.0

Le 20 mai 2026, WordPress a publié sa version majeure 7.0, baptisée « Armstrong ». Deux versions correctives ont suivi, la 7.0.1 le 9 juillet 2026, puis la 7.0.2 le 17 juillet, cette dernière étant une mise à jour de sécurité forcée corrigeant une faille critique. Au-delà de la modernisation du tableau de bord, de la palette de commandes (Command Palette) désormais disponible partout et du CSS personnalisé au niveau des blocs, le vrai titre de cette version tient en un sigle, IA.

Pour la première fois, WordPress fait entrer l'intelligence artificielle non plus par le biais d'une extension tierce, mais directement dans son cœur. Trois briques structurent ce changement :

- l'Abilities API, qui expose des opérations d'administration de façon encadrée
- l'AI Services Registry, qui permet de connecter des fournisseurs de modèles hébergés
- l'AI Client, l'interface unifiée autour de laquelle les extensions se standardisent désormais

Concrètement, cela signifie qu'un propriétaire de site peut désormais brancher une IA sur son contenu en quelques clics, là où il fallait auparavant assembler soi-même des extensions disparates. C'est un vrai gain de simplicité. Mais toute simplicité a un prix, et il vaut la peine de

savoir lequel.

Sous le capot, que fait vraiment l'IA dans le cœur de WordPress

Pour saisir l'enjeu, il faut comprendre ce que ces briques font réellement.

L'Abilities API ne « pense » pas. Elle expose des actions du site (créer un brouillon, classer un contenu, interroger des données) sous une forme structurée qu'un modèle peut déclencher de façon encadrée. C'est une passerelle, pas un cerveau.

Le cerveau, lui, reste le modèle de langage, et c'est là que se joue l'essentiel. Un modèle seul ne connaît pas votre contenu ; il ne connaît que ce sur quoi il a été entraîné. Pour qu'il réponde juste sur votre documentation, il faut lui fournir les bons extraits au bon moment. C'est le rôle du RAG (Retrieval-Augmented Generation). On découpe vos contenus, on les transforme en vecteurs (des empreintes numériques du sens), on les stocke, puis à chaque question on récupère les passages pertinents que l'on donne au modèle avant qu'il réponde.

Toute la qualité d'un chatbot se joue dans cette étape de récupération. Un RAG mal réglé renvoie des extraits hors sujet, et le modèle « brode » pour combler les trous : c'est l'hallucination. Un RAG rigoureux ne laisse jamais le modèle inventer, il l'oblige à s'appuyer sur des sources réelles. WordPress 7.0 fournit la tuyauterie pour brancher un modèle ; il ne règle pas, à votre place, la question de savoir quel modèle, où il tourne, et comment votre contenu lui est présenté.

L'écran Connecteurs : choisir et centraliser ses fournisseurs

Vient ensuite ce que WordPress appelle l'AI Services Registry, mais que l'administrateur découvre sous un nom plus concret dans son menu, Réglages > Connecteurs. C'est là qu'il décide quels services d'IA son site a le droit d'utiliser. WordPress livre trois connecteurs par défaut, OpenAI (les modèles GPT), Anthropic (Claude) et Google (Gemini), et l'administrateur saisit sa clé API une seule fois pour chacun. Toutes les extensions compatibles réutilisent ensuite cette connexion sans jamais redemander de clé, là où il fallait auparavant coller la même clé dans trois extensions différentes. Cette liste n'est pas fermée. N'importe quel autre fournisseur peut s'ajouter, dès lors que son extension s'enregistre auprès du système, et rien n'interdit techniquement de brancher un modèle que l'on héberge soi-même, sur son propre serveur.

C'est un point à retenir pour la suite, car il montre que le choix du modèle, et donc du lieu où part votre contenu, reste entièrement entre vos mains.

Depuis ce seul écran, l'administrateur active ou coupe un fournisseur et voit ce qui est branché. C'est une reprise en main de la gestion, pas encore de la confidentialité, car le contenu envoyé à un fournisseur externe part malgré tout sur ses serveurs. Nous y reviendrons.

L'AI Client : une prise unique pour toutes les extensions

Reste l'AI Client, l'interface commune par laquelle les extensions dialoguent avec les fournisseurs déclarés dans les Connecteurs. Pour l'administrateur, l'intérêt est très concret. Une extension écrite pour l'AI Client fonctionne avec n'importe quel fournisseur enregistré, ce qui lui permet de changer de modèle sans rien réinstaller. Il peut passer de Claude à GPT depuis l'écran Connecteurs, et ses extensions continuent de fonctionner sans modification. Là où il fallait auparavant jongler avec des connexions maison propres à chaque plugin, chacune avec sa clé et sa logique, il dispose désormais d'une prise unique et interchangeable.

Ce progrès a toutefois une contrepartie que l'administrateur avisé gardera en tête. Puisque la clé est enregistrée au niveau du site, toute extension installée peut s'en servir, sans cloisonnement ni approbation par plugin. Et ces clés, stockées en base de données, ne sont pas chiffrées, seulement masquées à l'écran. Une extension malveillante ou compromise a donc accès à une connexion payante et aux données qui transitent par elle. La commodité d'une prise unique est réelle ; sa surface d'exposition aussi.

Combien ça coûte ?

Brancher un modèle hébergé, c'est ouvrir un compteur. Chaque fournisseur a sa propre grille, mais tous facturent au token, cette unité qui découpe le texte en fragments. Contrairement à une extension que l'on installe une fois, un modèle hébergé se paie à l'usage : tant qu'il répond, il facture. Chaque question, chaque réponse, chaque extrait de contenu envoyé pour contextualiser la réponse consomme des tokens, donc de l'argent.

WordPress 7.0 a le mérite d'exposer ce compteur. L'écran Connecteurs suit la consommation de tokens, la latence et le taux d'erreur par fournisseur, ce qui permet à l'administrateur de surveiller sa dépense sans extension supplémentaire. Mais surveiller n'est pas maîtriser. Une extension mal réglée, une boucle entre deux plugins, un afflux soudain de trafic, et la facture peut grimper en quelques heures. Sans plafond de dépense fixé côté fournisseur, le budget file. On installe ainsi, dans une plateforme que l'on croyait à coût maîtrisé, un modèle de coût récurrent et variable qu'il faut apprendre à piloter.

J'ai branché un fournisseur... et maintenant ?

WordPress 7.0 vous a donné les outils, vous avez souscrit un plan chez OpenAI ou Anthropic, vous avez collé votre clé dans l'écran Connecteurs. Et là, question légitime : qu'est-ce que ça fait, réellement ?

Est-ce que ça affiche un chatbot pour vos visiteurs ? Non, pas en soi. Est-ce que ça vous aide à rédiger un nouvel article ? En partie, via les fonctions d'IA de l'éditeur, à condition qu'une extension les exploite.

Est-ce que ça améliore le contenu existant, reformule, résume, traduit, suggère des titres, détecte les pages faibles ? Seulement si un outil est branché pour le faire. Est-ce que ça répond aux clients, trie les demandes, alimente une base de connaissances interne ? Rien de tout cela n'est automatique.

Car c'est le cœur du malentendu. Enregistrer une clé API n'allume aucune fonctionnalité. Cela ouvre un robinet ; encore faut-il y brancher quelque chose. Les Connecteurs et l'AI Client sont une prise de courant, pas un appareil. Ce que l'entreprise veut vraiment, un assistant qui écrit à sa voix, qui améliore ses pages, qui répond juste à ses clients, reste à assembler, à régler et à ancrer dans son propre contenu. WordPress a livré l'électricité. Pas la lumière.

Ce qu'il vous reste à faire

Une fois la prise posée, tout reste à brancher. Il vous faudra trouver les bonnes extensions et vérifier qu'elles sont sérieuses, souscrire et payer les abonnements aux fournisseurs de modèles, régler la récupération de contenu pour éviter les réponses inventées, cloisonner les accès, poser des plafonds de dépense, tester la qualité des réponses avant de les exposer à vos clients, sécuriser des clés qui, rappelons-le, ne sont pas chiffrées, puis maintenir tout cela à jour à mesure que les modèles et les extensions évoluent. Rien d'insurmontable. Mais rien d'automatique non plus, et surtout, rien qui relève de l'après-midi de bricolage.

Le bilan

Reconnaissons-le, WordPress 7.0 est un beau progrès. L'IA entre dans le socle du CMS, avec une plomberie propre et unifiée : l'écran Connecteurs pour gérer ses fournisseurs, l'AI Client comme prise commune, l'Abilities API pour encadrer les actions, et un suivi de la consommation intégré. Le tout est ouvert, extensible, et pensé pour durer. Félicitations à la communauté, car c'est un travail considérable.

En face de ce progrès, il reste tout de même quelques questions. Sans faire de procès à personne, elles méritent d'être posées.

Le coût, d'abord. On quitte le modèle rassurant du forfait pour une facturation à l'usage, variable, qui grimpe avec le trafic et peut déraiser si rien ne l'encadre. À cela s'ajoute le prix des extensions qui exploitent réellement l'IA, et celui de leur maintenance dans le temps.

La sécurité, ensuite. Les clés d'API sont stockées en clair dans la base et accessibles à n'importe quelle extension installée. Chaque plugin ajouté élargit la surface d'exposition, et l'Abilities API, qui laisse l'IA déclencher des actions sur le site, introduit un risque nouveau : un texte malveillant, glissé par exemple dans un commentaire, pourrait tenter de déclencher une opération non prévue.

La confidentialité, surtout. Par défaut, votre contenu part chez un fournisseur tiers, potentiellement soumis à une juridiction étrangère. Et même en interne, une vraie question se

pose : sait-on précisément qui a accès à quoi ? Quelles données sont envoyées, à quel modèle, consultables par quelles extensions, et vues par quels utilisateurs ? Sans cloisonnement clair, la réponse est rarement nette.

Les risques induits, enfin. Chaque brique ajoutée, chaque extension branchée, chaque fournisseur connecté est un maillon de plus à surveiller, à mettre à jour, à sécuriser. L'IA n'ajoute pas seulement des possibilités, elle ajoute des responsabilités.

Autant de questions légitimes. Elles ont toutes une réponse, mais cette réponse demande du temps, des compétences et de la méthode. C'est bien beau, tout cela... mais qui va s'en occuper ?



La nécessité de la double compétence

Ces questions ont des réponses : il vous faut désormais, à vos côtés ou dans votre équipe, une compétence double, WordPress et IA. Or ce sont deux métiers distincts.

Celui qui entretient l'apparence de votre site, applique vos mises à jour et veille à sa bonne tenue fait un travail précieux, mais concevoir un assistant fiable, choisir et régler un modèle, sécuriser des accès et garantir des réponses justes relève d'une autre expertise. Si vous ne l'avez pas en interne, ou si votre prestataire habituel touche là aux limites de son domaine, la question reste entière.

On objectera peut-être que l'hébergeur propose déjà une IA. C'est vrai, et ces outils sont utiles, mais ils répondent à un autre besoin. L'assistant d'un Hostinger ou d'un IONOS vous aide surtout

à construire votre site ou à naviguer dans votre panneau d'hébergement ; ce n'est pas un assistant entraîné sur vos contenus pour répondre à vos clients. Et même hébergés en Europe et respectueux du RGPD, ces services tournent sur l'infrastructure du prestataire, avec son modèle et dans son écosystème. Le jour où vous changez d'hébergeur, l'assistant ne vous suit pas.

La vraie question n'est donc pas seulement « où sont mes données », mais « qui garde le contrôle, et puis-je repartir avec ? »

Vers qui se tourner

Plusieurs profils peuvent vous aider à les traiter : consultants WordPress, intégrateurs, experts en IA. L'essentiel est de trouver quelqu'un qui maîtrise vraiment les deux mondes, et qui prenne au sérieux la sécurité et la confidentialité plutôt que de les traiter comme des options.

De notre côté, c'est le terrain sur lequel nous travaillons depuis des années. La souveraineté des données n'est pas pour nous un argument récent, c'est une conviction de fond, forgée par l'habitude des environnements exigeants et de la cybersécurité en milieu hostile, là où l'on n'a pas droit à l'approximation. C'est cette exigence que nous mettons au service de votre assistant.

Ce que nous proposons est simple à énoncer. Un assistant privé, ancré dans votre propre contenu, testé avant sa mise en ligne sur des questions dont vous connaissez déjà les réponses. Vos données restent sur une infrastructure souveraine, en France ou en Europe, et ne partent jamais chez un fournisseur tiers. Un coût prévisible de service géré, plutôt qu'un compteur qui s'emballe. Un cloisonnement clair de ce que chaque audience a le droit de voir. Et parce que la pile repose sur des logiciels open source, vous n'êtes captif de personne : vous pouvez changer de modèle, d'hébergement ou de prestataire, et repartir avec vos données quand vous le souhaitez.

Le tout reste compatible avec le CMS que vous exploitez déjà, WordPress comme un autre, et répond dans la langue de vos utilisateurs, français, anglais, allemand ou autre. WordPress a livré l'électricité. Nous vous aidons à en faire de la lumière.

Découvrir notre chatbot IA souverain - Parlons de votre projet

Sources

- [Introducing the Connectors API in WordPress 7.0 - Make WordPress Core](#)
- [Introducing the AI Client in WordPress 7.0 - Make WordPress Core](#)
- [WordPress News - annonce de la version 7.0 « Armstrong »](#)
- [WordPress Developer Resources - Abilities API](#)