

L'IA dans WordPress : les règles ont changé

Utilisez l'IA comme bras droit et non plus comme un simple exécutant

Mardi 7 juillet 2026 • Valentin Grenier • Développeur WordPress FSE

Le constat global

**Beaucoup ont testé l'IA dans WordPress,
et beaucoup ont été déçus.**

L'IA par défaut

Des résultats aléatoirement
satisfaisants, des hallucinations,
des conventions non respectées ...

L'IA configurée

Confiance de délégation, on passe
d'exécutant à reviewer. L'IA fait son
boulot, et on fait le nôtre.

V o t r e c o n s t a t

**Qui a testé l'IA dans WordPress
et s'est dit que ça ne valait
pas le coup ?**

Mon propre constat

Il n'y a plus un jour sans IA dans mon quotidien de développeur WordPress

- Utilisation de Claude Code tous les jours depuis mars 2026, principalement pour m'assister dans la génération de code
- Plus de la moitié de mon code est généré par IA
- L'IA suit désormais ma manière de coder, d'un projet à un autre
- D'agent réactif à bras droit de confiance
- Possibilité de piloter 4 projets en parallèle, me faisant passer d'exécutant à donneur d'ordres / reviewer.
- ... j'explose mes quotas tous les jours !

Ce qu'on va voir aujourd'hui

01

Contexte

La règle qui n'a pas changé depuis qu'on utilise l'IA

02

WordPress + IA

Les règles qui ont changé avec l'utilisation de l'IA dans WordPress

03

Gain durable

Comment faire en sorte que le gain apporté par l'IA dans WordPress soit durable

04

Cas d'usage

Vos propres constats, retours d'expérience, tout le long de la conférence.

01

Le contexte

La seule règle qui n'a pas changé

La règle qui n'a pas changé

Le contexte, une obligation

- Le **quoi** (résultat attendu) + le **comment** (marche à suivre)
- Deux niveaux distincts : le contexte **jetable** (le prompt) et le contexte **persistant** (CLAUDE.md, skills, MCP)

> Code-moi un thème avec une page d'accueil et un header.

> Crée un thème FSE à partir de mon modèle. Génère le theme.json depuis les tokens Figma via le MCP (lien : [...]). Respect des WPCS et de ma convention de nommage des blocs. Génère un plan pour créer le thème étape par étape, du theme.json à la création des templates.

02

WordPress + IA

Les règles qui ont changé et qui font
toute la différence aujourd'hui

Les règles qui ont changé pour les webmasters

Vous ne codez pas ? WordPress vient de s'ouvrir

Avant

Au-delà de l'éditeur, le manque de connaissance renvoyait vers les forums ou vers un développeur.

Construire un site avec Studio Code (app desktop, sans terminal)

"Crée-moi un site portfolio avec une page d'accueil, une galerie et une page contact."

Il monte le thème, le contenu et les blocs en local, et vérifie son propre rendu.

/annotate : cliquer sur un élément de la page et demander une modification.

Maintenant

On décrit ce qu'on veut en langage clair.
L'IA construit, reformule et explique.

Comprendre et entretenir avec le MCP Adapter et le plugin "AI"

"Depuis que j'ai activé cette extension, mon menu a disparu. Qu'est-ce qui cloche ?"

"Quels plugins font doublon ? Lesquels je peux désactiver sans rien casser ?"

"Explique-moi comment le multilingue est géré sur ce site."

Les règles qui ont changé pour les développeurs

L'IA dans WordPress lève des barrières à l'entrée

Reprendre un site qu'on n'a pas fait

Avant, on pouvait passer des heures à analyser et à comprendre un site créé par un autre prestataire. Maintenant, la base de code est cartographiée en quelques minutes.

Les standards WPCS

Avant, coder en respectant les WordPress Coding Standards pouvait s'avérer chronophage. Aujourd'hui, elles sont appliquées en amont de la génération de code.

Blocs Gutenberg sur-mesure

Apprendre React pour créer un bloc Gutenberg n'est finalement plus une bannière à l'entrée qui réservait les blocs sur-mesure aux connaisseurs.

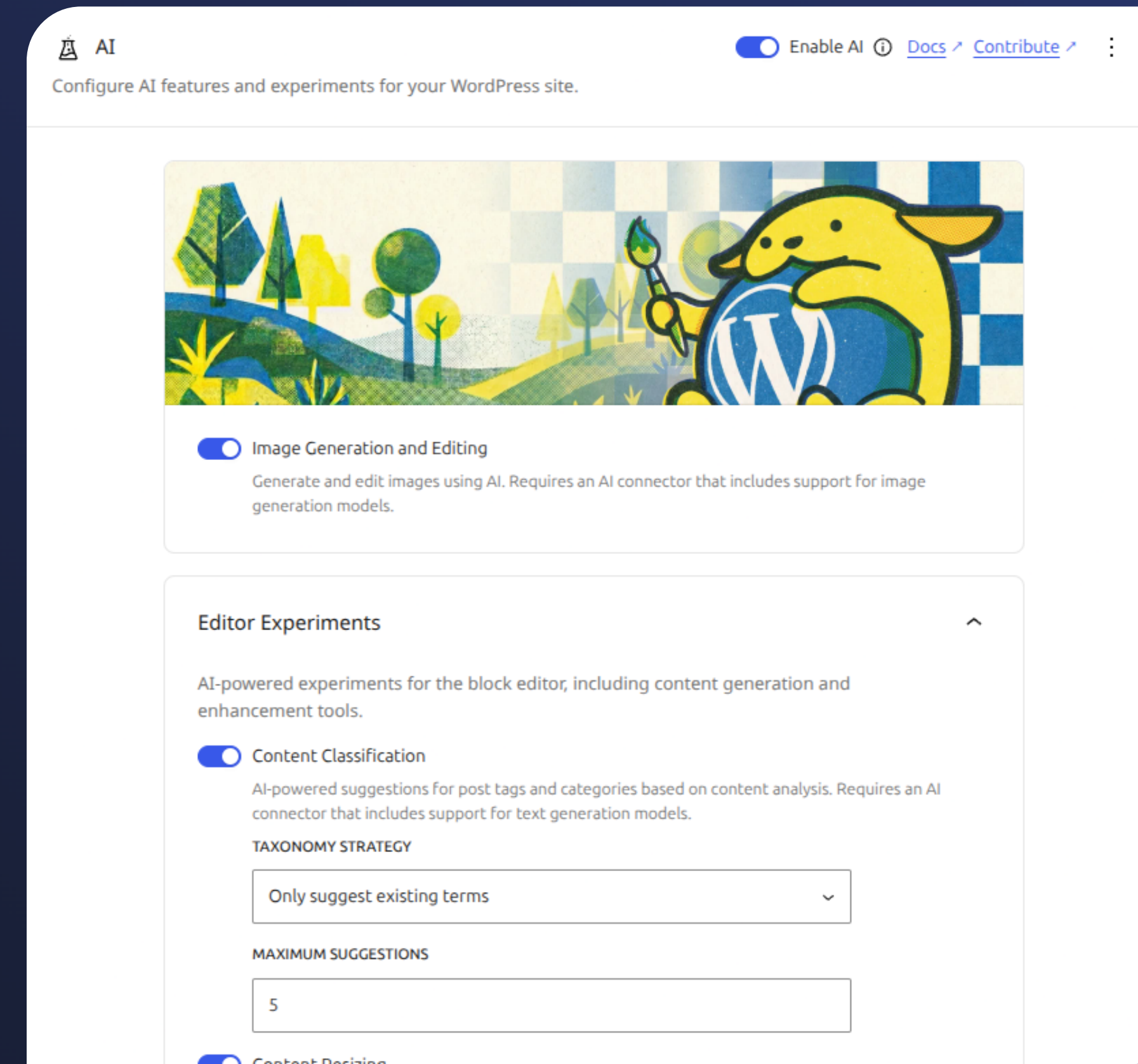
Les scripts jetables

WP-CLI, migrations, imports de fichier CSV ... avant, le temps passé pour du one-shot n'était pas rentable. Aujourd'hui, l'IA génère les scripts à usage unique en quelques secondes.

WordPress + IA

WordPress utilise l'IA

- **API Connectors + WP AI Client**
dans le core depuis 7.0
- **Réglages > Connecteurs** : clé API saisie
une fois (OpenAI, Anthropic, Google)
- **AI Experiments** : images à la une, textes
alternatifs, extraits générés
- L'IA au service du contenu, **directement**
dans l'éditeur



WordPress + IA

L'IA pilote le site

MCP Adapter expose les Abilities
du site à un agent externe

Un agent **interroge**
et agit via l'API

Le site devient
une **plateforme agentique**



```
"Repère mes articles sans meta description, rédige-les depuis le contenu et publie-les."
```

```
→ posts/query           trouve 8 articles sans meta description
→ posts/read             lit le contenu de chaque article
→ < raisonnement IA >    rédige une meta pertinente      ← impossible en WP-CLI
→ posts/update-meta      applique la meta description
```

```
✓ 8 articles optimisés pour le SEO
```

M C P F I G M A

DÉMO.

Retour d'expérience

Du design system Figma au theme.json

Cas réel : refonte du site de l'association WPFR

- **Récupérer les tokens** (couleurs, typographies, espacements) depuis Figma via le MCP
- **Générer le theme.json** de manière incrémentale : palette, fontSizes, spacing
- Demander à Claude Code de **créer une page "styleguide"** pour vérifier le rendu

Design system propre en entrée → theme.json propre en sortie

03

Rendre le gain durable

Configurer son IA une fois et la rendre active partout

Rendre le gain durable

Configuré une fois, actif partout



CLAUDE .md

- Le contexte projet chargé à chaque prompt : stack, conventions, structure
- À générer avec la commande **/init**, à faire évoluer avec le projet

Skills métier

- Les workflows récurrents encapsulés : création de bloc, audit WCAG, workflow GitHub ...
- L'IA suit la méthode définie ; à créer avec le plugin Claude Code officiel **skill-creator**

MCP

- Outils et services connectés à Claude Code : Notion, Figma, GitHub, WordPress ...
- L'IA agit dans l'écosystème local au lieu de travailler dans un bocal

Agents, hooks & commandes

- Sous-agents à qui déléguer, hooks qui s'exécutent automatiquement, commandes déclenchées à la demande
- L'IA suit des routines définies, au lieu d'exécuter au coup par coup

Rendre le gain durable

MCP Adapter : l'utiliser ou l'éviter ?

- Interroger et comprendre un site en langage naturel
- Piloter le site via son API : contenus, CPTs, réglages
- Cartographier les extensions et détecter les conflits
- Travailler sur un site distant sans accès au code ni au terminal

- Beaucoup d'opérations passent par WP-CLI, bien moins coûteuses en tokens
- Ne peut pas installer de plugins/thèmes, éditer les fichiers, accéder à la base de données
- Pour les cas cités au dessus, l'agent qui travaille sur la base de code reste plus adapté

Passer à l'action

Ce que vous pouvez mettre en place dès demain

Donner du contexte

- Lancer `/init` dans un projet WordPress
- Claude Code scanne la codebase et génère un premier fichier `CLAUDE.md`

Explorer un site hérité

- Ouvrir un site local, peu connu ou non
- Demander à Claude Code de l'expliquer

Créer des `skills`

- Repérer une tâche répétée :
 - générer un CPT
 - créer un bloc sur-mesure
 - refactoriser un fichier CSS
 - détecter une image trop lourde ...
- Encapsule-la dans un skill réutilisable d'un projet à l'autre

Auditer l'existant

- Inventaire des plugins installés

Connecter un agent

- Activer le MCP Adapter sur un site local, exposer quelques Abilities
- Piloter le site en langage naturel depuis Claude

Activer l'IA dans WP

- Installer AI Experiments sur un site local (WordPress 7.0)
- Configurer le connecteur de votre choix, rentrer la clé API, et générer du contenu

Pour conclure

Déléguer, ce n'est pas lâcher

- L'IA étend la portée, rend plus rigoureux, et absorbe une grande partie des tâches
- Mais déléguer, c'est aussi vérifier, relire, valider : **on ne délègue pas la responsabilité**
- L'IA exécute, l'humain garde la main

Anecdote

- Audit d'un plugin de formulaire de contact demandé à Claude Code
- Objectif : identifier la cause de l'échec d'envoi des formulaires
- Bug corrigé ...
- ... mais Claude a trouvé des centaines de données personnelles exposées en clair.



Télécharger les slides

Merci pour votre écoute !

Maintenant, parlons de vos retours d'expérience.

Valentin Grenier • studio-val.fr • Développeur WordPress FSE