addons presta

Lazy loading automatique

MANUEL D'UTILISATION · Version 1.0.0 · PrestaShop 1.7.8 – 9.x

1. À quoi sert le module

Une page de boutique contient souvent des dizaines d'images, et la plupart se trouvent sous la partie de la page que le visiteur voit en premier. Avec le lazy loading, le navigateur ne télécharge une image que lorsque le visiteur fait défiler la page jusqu'à elle : la page s'affiche plus vite et consomme moins de données. Les thèmes par défaut le font pour la plupart de leurs propres images, mais les thèmes tiers, les gabarits des modules et tout ce qui est saisi dans l'éditeur — descriptions de produits, pages CMS — ne le font généralement pas. Certains thèmes diffèrent même la seule image qui ne doit pas attendre : la grande image en haut de la page.

Le module ajoute le lazy loading aux images de votre boutique sans modifier un seul fichier du thème ou d'un module. Il veille aussi à ce que les premières images de chaque page se chargent tout de suite : c'est ce que Google mesure sous le nom de Largest Contentful Paint (LCP).

Compatible avec **PrestaShop 1.7.8, 8.x et 9.x** (PHP 7.2.5 ou plus récent), avec les thèmes Classic, Hummingbird et les thèmes tiers, ainsi qu'avec le multiboutique.

2. Installation

1. Téléchargez le module depuis **Mon compte** → **Historique des commandes** sur addonspresta.com.
2. Dans votre back-office, allez dans **Modules** → **Gestionnaire de modules** et choisissez **Installer un module**.
3. Déposez le fichier `ap_autolazyload_v1.0.0.zip` et attendez la confirmation.
4. Choisissez **Configurer** sur la carte du module.

Le module fonctionne dès l'installation, avec la technique native, et ne demande aucun autre réglage. Aucun cache n'est à vider : l'enregistrement des réglages s'en charge.

3. Fonctionnement

Le module agit à deux endroits.

Dans chaque gabarit. Chaque page est affichée à partir de gabarits : la page elle-même, les blocs des modules, la liste de produits chargée après un filtrage ou un tri, l'aperçu rapide, la fenêtre du

panier. Quand un gabarit a affiché son HTML, le module le lit, y trouve les balises `<img>` et `<iframe>` et leur ajoute le lazy loading. Les images saisies dans les descriptions de produits et les pages CMS font partie de ce HTML : elles sont couvertes elles aussi.

Dans la page terminée. Juste avant l'envoi d'une page au visiteur, le module la parcourt encore une fois : les premières images de la page sont réglées de façon à se charger tout de suite, et la première grande image reçoit une priorité haute.

Deux techniques sont proposées :

- **Native** (par défaut) — le module ajoute l'attribut `loading="lazy"`. Le navigateur décide du moment où il charge l'image. Aucun JavaScript n'est ajouté à la boutique et les moteurs de recherche voient des images ordinaires.
- **Script** — le module place dans `src` une petite image d'attente transparente aux proportions de l'image et met la vraie adresse de côté. Un petit script (environ 4 Ko, aucune bibliothèque) rend l'adresse quand l'image approche de l'écran. Cette technique permet de choisir à quelle distance les images commencent à se charger, de les faire apparaître en fondu et de différer les arrière-plans CSS en ligne.

Ce que le module ne touche pas :

- les images déjà gérées par un script de lazy loading de votre thème ;
- les images et gabarits que vous désactivez (voir *Gabarits exclus* et *Classes CSS* plus bas) ;
- les pixels de suivi et les images intégrées à la page en `data:` ;
- en technique « Script » : les icônes SVG et les images marquées comme données structurées (`itemprop`), qui doivent garder leur adresse ;
- les e-mails, les documents PDF comme les factures, et le back-office ;
- les images ajoutées par des scripts, les arrière-plans des feuilles de style et les vidéos.

Le module ne modifie aucun fichier du thème ni des modules et n'écrit rien dans les produits ou les pages. Il ne modifie que le HTML envoyé au visiteur ; le désactiver ou le supprimer rétablit immédiatement les pages d'origine. Il ne redimensionne, ne compresse et ne convertit aucune image, ne stocke que ses réglages (et, pendant un enregistrement, la liste des gabarits) et ne se connecte ni à addonspresta.com ni à aucun autre serveur.

4. Fonctions et réglages

Allez dans **Modules → Gestionnaire de modules → Lazy loading automatique → Configurer**. Le bloc Aide, en haut, mène à ce manuel et à la page du module sur addonspresta.com, dans la langue de votre back-office. Suivent trois panneaux de réglages, chacun doté d'un bouton **Enregistrer** qui les enregistre tous, puis le tableau **Gabarits avec images**.

Lazy loading

- **Différer le chargement des images** — active ou désactive tout le module pour la boutique.

- **Technique** — *Native : loading="lazy", sans script* ou *Script : image d'attente et chargeur*, décrites plus haut.
- **Les iframes aussi** — vidéos et cartes intégrées en `<iframe>` reçoivent `loading="lazy"`, avec les deux techniques.
- **Images chargées tout de suite** — nombre d'images, en haut de chaque page, qui ne sont jamais différées, de 0 à 20 (4 par défaut). La règle retire également le lazy loading qu'un thème a mis à ces images, comme à l'image principale d'un produit. Comptez ce qu'un visiteur voit sans défilement sur la page d'accueil et sur une page produit : logo, diapositive, image du produit. 0 désactive la règle.
- **Priorité haute pour la première grande image** — la première de ces images large d'au moins 300 pixels reçoit `fetchpriority="high"`, qui demande au navigateur de la télécharger en priorité. Il s'agit en général de la diapositive en page d'accueil et de l'image principale en page produit.

Gabarits

- **Gabarits concernés** — *Tous les gabarits de la boutique sauf les exclus* (par défaut) ou *Uniquement les gabarits listés ci-dessous*.
- **Gabarits listés** — utile au second choix : un gabarit par ligne.
- **Gabarits exclus** — gabarits jamais différés, quel que soit le choix ci-dessus. Par défaut, les deux gabarits de la galerie produit des thèmes Classic et Hummingbird, que la boutique affiche à nouveau quand un visiteur choisit une autre déclinaison.
- **Classes CSS qui désactivent le lazy loading** — séparées par des virgules, `no-lazy, skip-lazy` par défaut. Une image, une iframe ou un élément portant l'une de ces classes, ou l'attribut `data-no-lazy`, reste tel quel. Pratique quand une seule image d'une description est concernée : ajoutez-lui `class="no-lazy"` dans la vue source de l'éditeur.

Un gabarit est ce que la boutique affiche en une fois : une page (`index.tpl`, `catalog/product.tpl`, `cms/page.tpl`), le bloc d'un module (`module:ps_imageslider/views/templates/hook/slider.tpl`), une liste de produits chargée après un filtrage. Écrivez-le comme l'affiche le tableau **Gabarits avec images**. `*` remplace n'importe quels caractères : `catalog/listing/*` couvre toutes les pages de listes de produits, `module:ps_imageslider/*` tous les gabarits du diaporama. Un petit gabarit inclus par un autre fait partie de celui-ci ; pour laisser une seule image telle quelle, donnez-lui la classe `no-lazy`.

Technique par script

Ces réglages ne servent qu'avec la technique *Script*.

- **Commencer à charger avant que l'image soit visible** — distance de l'écran, en pixels, à laquelle une image commence à se charger (de 0 à 2000, 300 par défaut). Une valeur plus grande charge plus tôt : un défilement rapide montre moins d'emplacements vides.
- **Apparition en fondu des images** — une image chargée apparaît en un court fondu.
- **Copie de chaque image dans `<noscript>`** — ajoute l'image d'origine pour les visiteurs et robots sans JavaScript. Les pages s'alourdissent un peu ; Google exécute JavaScript et n'en a pas besoin.

- **Les arrière-plans CSS en ligne aussi** — éléments dotés d'une image d'arrière-plan dans leur attribut `style`, une façon courante de construire des bannières. Les arrière-plans des feuilles de style ne sont pas concernés.

Gabarits avec images

Le tableau montre quels gabarits votre boutique a affichés et ce que le module a fait dans chacun. Choisissez **Enregistrer pendant 10 minutes**, ouvrez dans votre boutique les pages à vérifier — accueil, catégorie, produit, page CMS — puis rechargez la page de configuration. L'enregistrement s'arrête de lui-même ; en dehors de lui, la boutique ne note rien. Chaque nouvel enregistrement, et chaque enregistrement des réglages, commence une nouvelle liste.

Colonne	Signification
Gabarit	nom à copier vers les listes ci-dessus
Images et iframes	combien le gabarit en contient
Différées par le module	combien le module en modifie
Laissées telles quelles	combien il n'en modifie pas, et pourquoi : déjà différées, désactivées, illisibles en toute sécurité, ou rien à différer
État	Concerné, Exclu par vous, Non listé ou Rien à faire

« Illisibles en toute sécurité » signifie que la balise est écrite d'une façon que le module ne prend pas le risque de modifier ; l'image se charge alors comme auparavant. « Désactivées » concerne les images avec `data-no-lazy`, une classe de la liste ci-dessus, une priorité haute ou `loading="eager"`.

5. Multiboutique

Les réglages sont enregistrés pour la boutique sélectionnée dans l'en-tête du back-office ; le tableau **Gabarits avec images** s'affiche quand une seule boutique est sélectionnée, car il dépend du thème de cette boutique. Chaque boutique peut avoir sa propre technique et ses propres listes, même si plusieurs boutiques partagent un thème. Dans le contexte « Toutes les boutiques » ou d'un groupe, seul un employé qui gère chaque boutique de ce contexte peut enregistrer. Une seule licence couvre toutes les boutiques de l'installation.

6. Désinstallation

Gestionnaire de modules → Lazy loading automatique → Désinstaller supprime les réglages du module et la liste enregistrée ; les pages redeviennent telles que votre thème les a faites. Si vous souhaitez conserver les réglages, choisissez plutôt **Désactiver**.

7. Questions fréquentes et dépannage

Comment vérifier que cela fonctionne ? Affichez une page de la boutique, faites un clic droit sur une image située loin dans la page, puis choisissez **Inspecter**. En technique native, la balise porte `loading="lazy"` ; en technique « Script », elle porte `data-ap-lazy` tant que vous n'avez pas défilé jusqu'à elle. Dans l'onglet **Réseau** des outils du navigateur, filtré sur les images, de nouveaux fichiers apparaissent au fil du défilement.

PageSpeed indique « L'image Largest Contentful Paint a été chargée en différé ». Augmentez **Images chargées tout de suite** de façon à couvrir la grande image en haut de cette page, et laissez **Priorité haute pour la première grande image** activée.

Un diaporama, un zoom ou une galerie de mon thème affiche une image vide ou erronée. Son script manipule lui-même les adresses des images. Lancez un enregistrement, trouvez son gabarit dans le tableau **Gabarits avec images** et ajoutez-le aux **Gabarits exclus**, ou donnez à l'image la classe `no-lazy`. La technique native pose rarement ce problème ; envisagez-la si votre thème compte beaucoup de scripts de ce genre.

Une image d'une description de produit ne doit pas être différée. Ajoutez-lui `class="no-lazy"` dans la vue source de l'éditeur.

J'ai modifié un réglage et la page reste identique. L'enregistrement vide le cache des gabarits de PrestaShop, mais un module de cache de pages entières, un CDN ou le cache de votre navigateur peut encore servir l'ancienne page. Rafraîchissez ce cache, ou videz le cache dans **Paramètres avancés → Performances**.

Les moteurs de recherche voient-ils mes images ? Oui. En technique native, la page contient des images ordinaires. En technique « Script », Google exécute le chargeur ; si des robots n'exécutent pas JavaScript, activez **Copie de chaque image dans** `<noscript>`.

Mon thème diffère déjà les images avec son propre script. Le module laisse ces images au thème et couvre le reste : gabarits des modules, contenu de l'éditeur et iframes.

Le tableau affiche « illisibles en toute sécurité » en face d'un gabarit. Rien n'est cassé : l'image de ce gabarit se charge comme auparavant. Si vous souhaitez la différer, ajoutez vous-même `loading="lazy"` à la balise du gabarit.

Factures et e-mails ne sont jamais modifiés, quels que soient les réglages.

8. Mises à jour et assistance

Les mises à jour sont gratuites à vie : téléchargez les nouvelles versions depuis **Mon compte → Historique des commandes** sur addonspresta.com et installez-les comme un nouveau module. Vos réglages sont conservés.

L'assistance est incluse pendant 12 mois à compter de la commande : écrivez à **mail@addonspresta.com** ou utilisez le formulaire de contact d'addonspresta.com (du lundi au vendredi, de 10 h à 16 h ; réponse généralement sous 2 jours ouvrés). Indiquez votre version de

PrestaShop, la version du module, le nom de votre thème et l’adresse d’une page qui montre le problème.