

MODULE PRESTASHOP POLARISLINK

Important !

Cette documentation s'adresse uniquement aux professionnels de l'hébergement tels qu'administrateurs et webmestres qui seraient amenés à installer notre module Prestashop PolarisLink dans le cadre de la synchronisation entre un Backoffice Polaris et un site Internet sous Prestashop.

Le module permet, sur injonction du Backoffice de gestion de stocks Polaris lié et selon les options configurées de :

- créer et associer des fiches produits sur Prestashop, accompagnés de leur description et de leurs photos à partir des informations du backoffice Polaris ;
- mettre à jour les stocks et les prix de Polaris vers Prestashop ;
- gérer les promotions, avoirs et chèques cadeaux (vouchers) depuis Polaris vers Prestashop ;
- d'exporter les commandes Prestashop validées en ventes sous Polaris ;
- synchroniser certaines informations clients utiles à la réalisation des commandes entre Prestashop et Polaris ;

A savoir : seule la monnaie principale du site Prestashop est prise en compte, le module ignorant les commandes exprimées dans d'autres monnaies.

SOMMAIRE

1 Mise en place et installation

1.1 Pré-requis

1.2 Mise à jour du module

1.3 Support

2 Paramétrage

2.1 Paramètres globaux

2.1.1 Permettre la connexion de VEGA Informatique à votre Backoffice Prestashop

2.1.2 Nettoyage automatique des tables de statistiques

2.1.3 Ne pas exporter les ventes antérieures au

2.1.4 Mise à jour des données par tâche CRON

2.2 Panneau de commandes

- 2.2.1 Ajouter un lien
- 2.2.2 Archives
- 2.2.3 Nettoyer journaux
- 2.2.4 Nettoyer erreurs
- 2.2.5 Test API
- 2.2.6 Chercher les mises à jour
- 2.2.7 Enregistrer

2.3 Sélecteur de produits connexes (ou coloris liés)

2.4 Association magasins

3 Configuration des liens

3.1 Accès

3.2 Options générales (du lien)

3.3 Stock minimum de sécurité

3.4 Poids de produit par défaut

3.5 Fournisseur associé et fournisseur de récupération

3.6 Classement des références

3.7 Configuration des thèmes Polaris

3.8 Configuration des catégories

3.9 Configuration des tailles

3.9.1 Je ne souhaite pas gérer des grilles de tailles multiples

3.9.2 Je souhaite gérer des grilles de tailles multiples

3.10 Configuration des niveaux

3.11 Options d'import/export

3.11.1 Montrer/cacher les produits selon leur stock

3.11.2 Utiliser le stock avancé si disponible

3.11.3 Utiliser la copie des photos

3.11.4 Importer les promotions en valeur plutôt qu'en pourcentage

4 Utilitaires mis à disposition

4.1 Resynchroniser

4.2 Tri des tailles

4.3 Mappage catégories

5 Fonctionnement

5.1 Correspondance des entités

5.1.1 Notes sur l'identification des modèles et des déclinaisons

5.1.2 Notes sur l'identification des autres données

5.2 Attributs génériques

5.3 Notes sur la méthode de mise à jour des modèles

5.4 Utilisation du groupe de produits

5.4.1 Exemple : choix des coloris disponibles sur la fiche produit.

5.4.2 Exemple : utilisation de \$relatedProductsAttributes

6 Les données concernées

- 6.1 Les données exportées par la passerelle
- 6.2 Les données importées par la passerelle
- 6.3 Choix et restriction des données mises à jour sur Prestashop
- 6.4 Pourquoi les adresses ne sont pas synchronisées ?

7 Documentation technique du module

- 7.1 Variables de configuration
- 7.2 Autres variables
- 7.3 Tables exploitées, champs ajoutés

8 FAQ

- 8.1 La clé est refusée avec le message *401 - Unauthorized*
- 8.2 Erreur SHOW PROCESSLIST

MISE EN PLACE ET INSTALLATION

Le webmestre administrant le site peut télécharger le module à installer ici : <http://vega.link/pl17>

Une fois installé, il est nécessaire d'aller dans la page de configuration du module et :

- faire le tour de la configuration pour s'assurer que les données importées / exportées correspondent bien au souhait de votre client et de votre mise en page ;
- **générer une clé d'accès à remettre au client** afin qu'il puisse lier son module avec son backoffice Polaris présent dans ses magasins ;
- vérifier que l'accès est actif.

Pré-requis

Ce module est uniquement certifié compatible pour les versions de Prestashop à partir de la version 1.7.6.1, sous PHP 7.2.

A savoir :

Le module est normalement compatible pour les versions 1.6 à partir de la version 1.6.1.24, sous PHP 7.0 mais n'est plus maintenu en raison de l'arrêt de support de cette branche de Prestashop.

Les autres pré-requis qui en découlent :

- Base de données MySQL ou MariaDB uniquement ;

- Base de données *ACID* (InnoDB minimum pour MySQL ou MariaDB, pas de MyISAM)
- Commande SHOW PROCESSLIST retournant au moins les threads de l'utilisateur connecté sous MySQL/MariaDB ;

Pour assurer le fonctionnement correct du pont, il est nécessaire d'utiliser quelques données normalement utilisées par Prestashop. Cela garanti un fonctionnement sans mappage (et donc sans source inutile de conflit ou de désynchronisation), et a le mérite de pouvoir être manipulé manuellement et facilement en cas de problème ou de reprise de données.

En effet, l'exploitation du pont en mode "*Exportation catalogue*" et "*Importation des ventes*" nécessite :

- l'utilisation privée d'un ou plusieurs **fournisseurs spécifiés**, pour reconnaître les produits dont le pont s'occupe ;
- l'utilisation privée des **références fournisseurs** (*supplier_reference*), pour reconnaître ces produits sous Polaris ;
- ce fournisseur doit-être le fournisseur par défaut des références pour que l'importation des ventes fonctionnent.

Ces champs ne sont en revanche pas systématiquement requis lorsque nous utilisons uniquement le mode "*Synchronisation du stock et des prix*", qui lui peut se baser uniquement sur les GENCODES des articles, s'ils existent (seules les références possédant un de ces codes-barres seront alors reconnues et mises à jour).

Mise à jour du module

Le module embarque une procédure de téléchargement de mise à jour que vous pouvez utiliser depuis la configuration du module (bouton "chercher les mises à jour").

Une fois la mise à jour téléchargée, **vous devez la mettre à jour depuis le gestionnaire de modules de Prestashop (section mises à jour)**.

Support

En cas de problème avec l'utilisation du module ou d'erreur inexpliquée, le webmestre administrant le site peut obtenir du support en envoyant ses demandes à polarislink@vega-info.fr en précisant les coordonnées du client et l'adresse du site pour lequel il intervient.

En cas de dysfonctionnement, la demande de support devra alors comporter l'erreur exacte rencontrée, le contexte (au moins un exemple de produit impactés) ainsi que les détails de sa reproduction afin de faciliter son éventuelle résolution.

Nous invitons fortement nos clients à ne pas nous contacter directement et de laisser leur webmestre nous contacter : notre support est adapté aux professionnels de l'hébergement et aux concepteurs du site.

Avant de contacter le support, merci de vérifier :

- que le dysfonctionnement n'est pas lié à l'environnement d'exécution de Prestashop ;
- que les fonctionnalités de Prestashop liées au catalogue ne produisent pas d'erreur lorsqu'elles sont utilisées via l'interface du backoffice ;
- que votre version de PHP est à jour et est compatible avec la version de Prestashop que vous utilisez ;
- que le module est lui-même à jour ;

PARAMÉTRAGE

Important !

Quelle que soit votre modification dans les options suivantes, pensez à les enregistrer !

Paramètres globaux

Permettre la connexion de VEGA Informatique à votre Backoffice Prestashop

En activant cette option, vous autorisez nos équipes techniques à rentrer sur votre backoffice pour en vérifier la configuration lors d'une éventuelle opération de maintenance.

Vous ne devez pas créer un accès spécifique pour nos techniciens ou nos développeurs : en passant par cette option, toutes leurs interventions sont journalisées.

Nettoyage automatique des tables de statistiques

Cette option vous permet de nettoyer automatiquement les tables de statistiques de fréquentation de Prestashop qui peuvent devenir très volumineuses et lourdes avec le temps.

Sur la plupart des installations ces données ne servent à rien en plus d'être souvent incompatibles avec le RGPD et sa mise en place dans votre structures. Elles sont en outre aisément remplacées par l'utilisation d'une solution tierce plus performante et plus adaptée (Matomo, GoogleAnalytics, ...).

Ne pas exporter les ventes antérieures au

Paramètre le module afin qu'il n'exporte pas les commandes passées avant cette date. Cela garanti que les commandes antérieures à l'installation du module ne soient réimportées dans le backoffice Polaris du client et fausse ses encaissements.

Ne le modifier que si vous savez ce que vous êtes en train de faire !

Mise à jour des données par tâche CRON

Notre module essaye d'appliquer les modifications dès qu'elles sont envoyées par le backoffice Polaris, mais certaines installations limitent volontairement les ressources accordées à PHP et les scripts s'arrêtent souvent après 30 secondes, avant que leurs travaux ne soient terminés. Ce n'est pas grave en soit car ils reprennent lors de la soumission des travaux ultérieurs mais dans quelques situations, les travaux s'entassent et ne peuvent jamais aboutir ou bien sont très longs.

Vous pouvez activer la mise à jour des données pour que la synchronisation se fasse en deux temps :

1. Polaris exporte le fichier de synchronisation
2. Une tâche CRON système démarre le traitement du fichier à intervalles réguliers ;

Pour connaître l'appel de la tâche CRON à effectuer, activez l'option et enregistrez. Le module vous fournira alors la configuration à positionner sur le gestionnaire de tâches automatisées du système (cron).

Panneau de commandes

Ajouter un lien

Ajoute un lien de synchronisation.

La plupart du temps, cette option ne sera pas utilisées.

Archives

La page des archives vous permet de visualiser ou télécharger tous les fichiers de synchronisation archivés. Les fichiers sont automatiquement archivés en cas d'erreur pour permettre à notre maintenance de les analyser en cas de besoin.

Nettoyer journaux

Toutes les modifications opérées par le module sont journalisées et présentes dans le journal de Prestashop. Le nombre de celles-ci peut devenir conséquent avec le temps.

Cette option vous permet de supprimer uniquement les lignes concernant ces modifications afin d'alléger votre journal.

Nettoyer erreurs

Cette option vous permet de retirer de votre journal Prestashop toutes les lignes relatives à des erreurs de traitement de la synchronisation une fois que vous avez corrigé le problème et que vous n'en avez plus besoin.

Test API

Ce lien vous permet de vous positionner automatiquement sur l'adresse utilisée par notre backoffice Polaris pour atteindre le module. Cela permet de vérifier manuellement si le lien est fonctionnel et qu'il répond correctement.

Chercher les mises à jour

Effectue une recherche de mise à jour sur nos serveurs.

Si la recherche est concluante, le module remplace ses binaires par ceux trouvés.

Attention, même si ce n'est pas systématique, il est quand même nécessaire de mettre à jour le module pour Prestashop

en passant par le gestionnaire de modules.

Enregistrer

Quelle que soit votre modification dans les options suivantes, pensez à les enregistrer !

Sélecteur de produits connexes (ou coloris liés)

Selon la configuration du backoffice Polaris, certains produits sont liés sans être des déclinaisons.

Prestashop gérant assez mal les différents coloris d'un même produit dans les pages produits (impossibilité d'avoir un résultat présentant un produit pour chaque couleur dans les pages catalogue plutôt qu'un simple produit regroupant tous les coloris), c'est souvent le cas pour un même produit existant dans des couleurs différentes. Notre backoffice Polaris se propose ainsi de créer un produit Prestashop pour chaque couleur différente, et de lier tous les produits.

Le sélecteur de produits connexes vous permet d'afficher simplement des liens vers les autres produits liés sur la page produit de chacun d'entre-eux (*aka. "existe aussi en"*).

Vous pouvez activer ou non le sélecteur dans les options de configuration présentes après celles des liens, mais également filtrer les produits avec ou sans stock, cacher le produit actuellement visionné de la liste, le personnaliser visuellement et laisser le module gérer ou non les liens canoniques pour le référencement (pour ne pas subir de pénalité SEO si la description du produit est la même).

Le sélecteur vient se placer tout seul en utilisant le *hook* présent dans le thème par défaut en dessous du prix (`{hook h='displayProductPriceBlock' product=$product type="after_price"}`), à condition que ce dernier n'est pas été supprimé.

Si le concepteur du thème que vous utilisez a enlevé ce *hook* ou que vous souhaitez le positionner autre part, la configuration vous donne toutes les instructions nécessaires pour placer un *hook* spécifique.

Association magasins

[A VENIR - A partir de la version 4.00 de la passerelle // sous réserve de faisabilité]

L'association magasins vous permet de faire correspondre les boutiques renseignées dans Prestashop à vos magasins Polaris.

Cette configuration est nécessaire quand vous souhaitez pouvoir récupérer un stock différencié par boutique dans la table `polarislink_stocks` pour une utilisation dans le cadre d'un développement tiers spécifique ou pour tous les modules click'n'collect déjà compatibles.

L'association magasins vous permet également de pouvoir ventiler vos commandes dans vos différents magasins Polaris directement depuis la gestion des commandes de Prestashop.

CONFIGURATION DES LIENS

Vous pouvez avoir plusieurs liens, indépendants des uns des autres, pour relier plusieurs fois votre backoffice Polaris à votre site avec des configurations différentes (monnaies différentes, produits différents, ...).

Quand vous modifiez une de ces options, il est nécessaire d'exécuter une resynchronisation pour en appliquer les effets sur les anciens produits.

Accès

La rubrique accès vous permet de déterminer un nom pour reconnaître et identifier facilement la liaison mais également de déterminer si l'accès à la synchronisation est actif ou provisoirement suspendu, ainsi que la clé d'accès.

La clé d'accès est celle qui identifie le backoffice Polaris et est donc à recopier dans les options de ce dernier.

Si vous souhaitez régénérer une nouvelle clé, cliquez sur le bouton éponyme et n'oubliez pas de recopier la clé dans les options de votre site Internet côté Polaris.

Nos techniciens Polaris seront là pour vous guider dans cette dernière étape.

Options générales (du lien)

Les options générales vous permettent de spécifier la portée des éléments synchronisés, tels que la langue par défaut (langue), la devise (devise gérée) dans laquelle seront interprétés les prix et montant, l'appartenance des produits créés en cas d'utilisation de multi-boutiques.

Stock minimum de sécurité

Le stock de sécurité est une option qui retranche une quantité aux stocks communiqués par Polaris pour établir le stock afin de pallier au problème de ventes concomitantes sur le site et en boutique.

Par exemple, si vous spécifiez 1 et que vous avez 2 pièces en stock, la passerelle considèrera que vous n'avez plus qu'une pièce en stock ($2 - 1$) sur votre site Internet.

Spécifiez 0 pour désactiver ce mécanisme.

Poids de produit par défaut

Indiquez ici un poids en gramme par défaut pour vos produits provenant du backoffice Polaris. Si Polaris renvoie un poids égal à 0 pour un produit, ce dernier sera remplacé par la valeur spécifiée ici.

Fournisseur associé et fournisseur de récupération

Pour retrouver les produits dont les informations sont modifiables, le module utilise la référence fournisseur indiquée sur chaque produit. Reste à déterminer le fournisseur qui représentera Polaris pour les importations.

Vous pouvez spécifier ici quel est le fournisseur dédié au backoffice Polaris.

Le fournisseur de récupération concerne les produits où seul le stock et les prix doivent être mis à jour et non les informations catalogue. Cette option est notamment utile lorsque vous récupérez un site Internet existant et que vous ne désirez maintenir que le stock des produits, sans pour autant modifier leur information catalogue.

Classement des références

Vous pouvez décider dans cette section de ventiler automatiquement les produits Polaris dans des catégories

correspondant à la classification Polaris :

- par RFS ;
- par 4 type de catégories ;
- par saison ;
- par collection de produits.

Si cette option est activée, les produits seront automatiquement ventilés dans ces catégories dont vous pouvez choisir les catégories parentes comme vous le souhaitez. Au webmestre d'activer ou de renommer les catégories parentes pour montrer ou non celles-ci sur le site.

Vous pouvez décider à quelle catégorie par défaut un produit sera assigné grâce à l'option "racine de la catégorie par défaut". Par exemple, si celle-ci est positionnée sur celle de la collection, les produits auront tous des catégories dont le parent est "collection" en guise de catégorie par défaut.

Configuration des thèmes Polaris

Les thèmes et groupe de thèmes sous Polaris peuvent être ignorés ou constitués :

- des catégories, le nom du groupe étant ignoré et les thèmes des catégories filles d'une catégorie parente à préciser ;
- des caractéristiques, le nom du groupe donnant lieu au nom de la caractéristique et le premier thème affecté dans ce groupe une valeur de cette caractéristique ;
- des mots-clés, le nom du groupe étant ignoré et les thèmes constituant les mots clés

Configuration des catégories

Les catégories Polaris peuvent constituer des catégories Prestashop ou bien des caractéristiques, ou les deux.

Il existe 4 types de catégories Polaris différentes.

Configuration des tailles

Indépendamment de gérer les tailles, vous pouvez choisir de gérer des grilles de tailles multiples ou non.

Je ne souhaite pas gérer des grilles de tailles multiples

Si vous décidez de ne pas gérer plusieurs grilles de tailles, alors vous devez spécifier quel groupe d'attributs gèrera toutes les tailles de produits. A chaque nouvelle taille, un nouvel attribut de ce groupe sera créé et utilisé, mélangeant toutes les tailles au sein du même groupe (aka. S, M, L, XL, 42, 43, 44 ...).

Je souhaite gérer des grilles de tailles multiples

Cette option vous permet de séparer les différentes grilles de tailles dans des groupes d'attributs Prestashop différents.

Si vous décidez de gérer plusieurs grilles de tailles, alors :

- il vous faut créer autant de groupes d'attributs que de grilles que vous avez à gérer ;
- lors de la synchronisation, si le nom de la grille de tailles Polaris correspond au nom d'un groupe d'attributs Prestashop, c'est ce groupe qui sera utilisé pour gérer la taille ;
- sinon, le groupe choisi pour la taille sera utilisé pour accueillir cette taille.

Configuration des niveaux

Les niveaux de déclinaison Polaris peuvent constituer soit des attributs, soit des caractéristiques.

Chaque niveau de déclinaison s'accompagne d'un niveau générique qui peut lui même constituer soit un attribut, soit une caractéristique. Les niveaux génériques sont très utilisés pour gérer les simplifications de coloris et leur nuance (couleur : gold, générique : jaune).

Il existe 4 niveaux différents.

Options d'import/export

Il existe une multitude d'options d'import/export pour contrôler la manière dont le module importe ou exporte les données de Prestashop.

Dans la suite, nous n'allons revenir que sur certaines.

Montrer/cacher les produits selon leur stock

Cette option vous permet de contrôler automatiquement la visibilité de votre produit sur le site dès lors qu'il n'a plus de stock ou qu'il en a de nouveau.

Désactiver un produit lorsqu'il n'a plus de stock peut être préjudiciable pour le référencement si vous comptez réapprovisionner un jour : il vaut mieux conserver la page produit et spécifier que celui-ci est en rupture de stock.

Modifier la visibilité de ce dernier permet en revanche de toujours sortir la page du produit lorsqu'elle est accédée directement mais de contrôler si le produit sort ou non dans les résultats de recherche interne ou sur les pages catalogues du site.

Utiliser le stock avancé si disponible

Prestashop 1.6 disposait d'une gestion d'entrepôts. Sur le backoffice Polaris, plusieurs magasins peuvent se mettre à plusieurs pour fournir un stock commun pour le site Internet. Par défaut, le module agrège donc l'ensemble des stocks pour proposer un stock globalisé à la vente.

Activez cette option pour obtenir un stock au magasin (magasin Polaris = entrepôts Prestashop) quand le système le permet.

(Ce système est désactivé sur Prestashop 1.7 et nous conseillons de laisser cette option désactivée sur ces systèmes.)

Utiliser la copie des photos

Par défaut, le pont fait la liaison de vos photos Prestashop avec celle importées en utilisant des liens symboliques. Si votre système ne les supporte pas, activez cette option pour travailler avec des copies, vos photos occuperont alors le double de stockage mais seront copiées dans le répertoire /img de Prestashop.

Importer les promotions en valeur plutôt qu'en pourcentage

Cette option a été ajoutée en version 3.23 et ne fonctionne qu'avec Polaris version 9 et permet d'importer les promotions en montant de réduction plutôt qu'en pourcentage de réduction.

Quelques marchands pratiquent en effet des promotions de type (-5€, -10€, -15€) et souhaitent afficher le gain rond en numéraire plutôt qu'un pourcentage de réduction non conventionnel (-17,35%, -24,33%, ...).

Lorsque cette option est changée, il est nécessaire de procéder à un renvoi total des données.

UTILITAIRES MIS À DISPOSITION

Resynchroniser

Dans son fonctionnement normal, le backoffice Polaris est configuré pour envoyer au module que les changements intervenus depuis la dernière communication. En sélectionnant cet utilitaire, vous configurerez le prochain envoi pour qu'il communique la totalité des données.

Le bouton passe au jaune lorsque la demande a été prise en compte mais pas encore appliquée.

Une fois les données entièrement envoyées, le fonctionnement normal est rétabli et le bouton perd sa couleur jaune.

Tri des tailles

Cet utilitaire permet de trier les attributs prestashop représentant les tailles dans leur ordre naturel (XS, S, M, L, XL ou 42, 43, 44). Le tri s'effectue au moment de l'exécution de l'utilitaire et ne conditionne pas que les prochaines tailles créées seront triées.

Mappage catégories

Le mappage de catégorie vous permet de faire correspondre les RFS Polaris avec des catégories existantes de Prestashop.

Il vous présente une liste des catégories aspirées de Polaris et vous propose de ventiler chacune de ces catégories dans une ou plusieurs catégories Prestashop.

La modification des produits concernés est instantanée : dès que vous associez une catégorie à une autre, les modifications sont effectuées sur les produits existants et à venir.

? Il est nécessaire que le module ait communiqué au moins une fois totalement avec Polaris pour que les catégories de ce dernier soient exportées. Il est évidemment nécessaire qu'au moins un produit d'une RFS soit publié pour que celle-ci

le soit également sur le site !

FONCTIONNEMENT

Le module d'extension de Prestashop réceptionne des fichiers de modifications au format XML et le traite de manière synchrone (à la requête) ou asynchrone (par tâche CRON système).

Correspondance des entités

Le système fonctionne globalement sans mappage : quand cela est possible, il procède par équivalence d'entité sur la base d'une clé de reconnaissance, choisie pour l'être et ne devant pas être modifiée.

Le tableau suivant décrit l'équivalence choisi par la passerelle sur les deux systèmes.

Si pour un export, un import ou une synchronisation, il y a plus d'une entité qui répond pour une clé, alors c'est toujours la plus ancienne qui est choisie en sélection ou en modification.

Entités		Clé ¹		Notes
Polaris	Prestashop	Polaris	Prestashop	
Rayon, Famille, Sous-Famille	Catégorie	Numéro interne + libellé étendu ²	Référence Polaris cachée, si non trouvé, le nom	<p><i>Ces informations sont synchronisées en tant que sous-catégories de catégories spécifiques sous Prestashop, toutes elles-même membres d'une catégorie parent nommée par défaut "Classification Polaris".</i></p> <p>Ces catégories ne doivent pas être renommées ou déplacées directement, mais la catégorie racine dont elles dépendent elle, peut l'être.</p> <p><i>Voir la section spécifique aux catégories.</i></p>
Catégorie1 à 4				
Collection				
Modèle	Groupe de produits	Numéro interne	Numéro interne Polaris	<p><i>Le groupe de produits est une information ajoutée par le pont pour permettre aux webdesigners de relier plusieurs produits entre-eux au moment de l'affichage de la page produit.</i></p> <p><i>Nous pourrons ainsi indiquer automatiquement qu'un produit Prestashop est décliné en plusieurs couleurs, même si un produit par couleur est créé.</i></p>

Modèle + Niveaux	Produit	EAN Polaris ou GENCODE	Référence du fournisseur Polaris ou EAN13	<p><i>Il est possible de choisir le dénominateur d'un produit Prestashop par rapport à son équivalent Polaris :</i></p> <ul style="list-style-type: none"> • soit le modèle Polaris = Produit prestashop • soit le modèle Polaris + niveaux sélectionnés = Produit prestashop. <p><i>Il est donc normalement possible de choisir qu'un c'est le modèle Polaris accompagné de sa déclinaison couleur qui forme un seul produit Prestashop, décliné par matière et par taille.</i></p>
Déclinaison + Taille	Déclinaison	Composition des libellés	Composition des noms	<p><i>Les tailles sont une déclinaison comme une autre pour Prestashop.</i></p> <p><i>Il est possible de choisir les niveaux envoyés sur la passerelle (Toujours, au besoin, jamais si possible).</i></p>
Taille	Attribut			<p><i>Les tailles et les niveaux sont synchronisés comme des attributs membres des groupes tailles, niveau1 à 4, par rapport à leur libellés.</i></p> <p><i>A partir de la v2 de la passerelle et à condition de position l'équivalence des produits correctement, il est possible de choisir de faire correspondre les niveaux de Polaris à des caractéristiques en sus ou en remplacement des attributs.</i></p> <p><i>Dans cette version, il est également désormais possible d'ignorer certains niveaux.</i></p>
Niveaux 1 à 4	Attribut et/ou caractéristique	Libellé étendu ²	Nom	<p><i>Dans cette version, il est également désormais possible d'ignorer certains niveaux.</i></p>
Magasin	Magasin	-	-	<i>Non gérés</i>
	Dépôt	Code magasin	Référence	<i>Lorsque le stock avancé est activé sous Prestashop, la référence de l'entrepôt correspond au code magasin sous Polaris.</i>
Marque	Marque	Libellé étendu ²	Nom	<i>Les marques Polaris sont strictement équivalentes aux marques de Prestashop.</i>
-	Fournisseur	-	Paramétrage	<i>Un fournisseur de catalogage et un fournisseur de reprise sont automatiquement créés s'ils n'existent pas, mais ils sont réglables dans les options de la passerelle. Du côté Polaris, cette entité ne correspond à rien et n'est pas utilisée.</i>

Multi-thème	<p><i>Au choix :</i></p> <ul style="list-style-type: none"> • Mot-clé • Valeur de caractéristique • Catégorie 	Libellé étendu ²	Nom (et selon, catégorie parente ou caractéristique)	<p>Chaque mot clé Prestashop correspond à un thème Polaris, les groupes de multi-thèmes ne sont pas gérés et sont simplement ignorés. OU</p> <p>Chaque thème Polaris correspond à une valeur de caractéristique Prestashop, celle-ci étant géré par le groupe multi-thème correspondant. Dans ce mode, il ne peut y avoir qu'une caractéristique : c'est la première qui est prise en compte en cas de duplicité.</p> <p>OU</p> <p>Chaque thème Polaris correspond à une catégorie Prestashop fille d'une catégorie correspondant au groupe de thème.</p>
Solde et promotion	Prix spécifique	EAN Polaris + dates	Référence du fournisseur Polaris + dates	Prix spécifiques valables pour toutes les devises, pays, groupe et groupe de clients.
Membre	Client	Adresse mail ou <i>fallback (cf notes)</i>	Adresse mail ou <i>fallback (cf notes)</i>	<p>S'il n'y a pas de correspondance avec l'adresse mail, les deux parties recherchent une ancienne adresse mail.</p> <p>Les informations du site Internet sont prioritaires : si deux modifications ont lieu sur un même client entre deux synchronisations, ce sont les informations de Prestashop qui sont gardées.</p>
Adresse	Adresse	-	-	<p>Les adresses ne sont ni envoyées, ni reçues.</p> <p>Selon le RGPD, les coordonnées ne sont utilisables que pour assurer la livraison et la facturation. Puisque le traitement de la commande a lieu sur Prestashop, elles n'ont pas vocation à remonter sur Polaris pour alimenter un fichier qui aurait d'autres finalités. En vertu du règlement, il doit s'agir d'un traitement séparé pour lequel l'internaute doit donner un accord explicite.</p> <p>Ce traitement n'est pas géré par cette passerelle.</p>
Débit	Commande (validée)	Tag de vente extérieure	Numéro de commande	<p>Par défaut, seules les ventes étant passées par le statut "paiement accepté" sont envoyées à Polaris. Ce comportement est configurable mais nous déconseillons de le changer pour des raisons de cohérence de stock (Polaris décrémente le stock quand la vente lui est signifiée).</p> <p>Le débit est créé si possible à la date de création de la commande Prestashop, si impossible car clôturé, alors, le système choisi la première date disponible.</p> <p>Le système ne gère pour l'instant pas les retours, ni les annulations, qui doivent être répercutées manuellement sur Polaris après importation de la commande.</p>

Bon	Règle panier	EAN	Code	<p>Les bons créent des règles paniers dont le nom correspond à leur libellé, avec un code identique au code barre, limité à une utilisation.</p> <p>Les données prises en charges sont identiques de chaque côté : limité ou non à un client, valide de date à date, pour une réduction en montant ou en pourcentage.</p>
-----	--------------	-----	------	---

¹ : la clé est l'élément sur lequel la passerelle se base pour identifier à quoi correspond les entités d'un côté, comme de l'autre.

² : le libellé étendu correspond au libellé internet de l'entité sous Polaris s'il existe ou s'il n'est pas vide, et dans le cas contraire, au libellé classique. Si ce dernier est également non renseigné, un texte indéfini est choisi, comme par exemple : «inconnu».

Notes sur l'identification des modèles et des déclinaisons

Afin d'opérer les modifications sur les modèles ou leurs déclinaisons et de faire correspondre les produits *Prestashop* aux modèles *Polaris*, il est nécessaire de les reconnaître.

Deux méthodes permettent d'identifier à quel produit se rapporte les modifications apportées, elles sont essayées dans l'ordre par le module d'extension :

1. un des code-barres EAN *Polaris* doit correspondre à la référence fournisseur du fournisseur «*Polaris*» ou «*PolarisReprise*» sous *Prestashop* ;
2. si non concluant et qu'il s'agit d'une mise à jour du stock (ne fonctionne pas en mode catalogue), la référence *gencode* sous *Polaris* doit correspondre au champ *EAN13* sous *Prestashop*.

Si aucune des deux méthodes n'est concluante, le système considère que la référence n'existe pas sous *Prestashop*.

Important !

Pour un modèle, il faut comprendre : n'importe lequel de tous les codes barres Polaris des articles de ce modèle, qu'ils soient publiés ou non, en quarantaine ou non.

Notes sur l'identification des autres données

Toutes les données de catégorisation (rfs, catégories, saisons, collections, etc...) sont identifiées par leur numéro interne Polaris.

En cas de changement de libellé, c'est bien la même catégorie qui est réutilisée si elle a précédemment été créée par le pont. Si une catégorie n'a pas été créée par le pont mais porte le même nom que celle que le pont doit créer, alors ce dernier essaye de la réutiliser quand c'est possible.

Toutes les données d'attributs (niveaux, tailles, etc...) sont généralement identifiées par leur libellé.

En cas de changement de libellé, une nouvelle donnée est créée, l'ancienne n'est pas modifiée, ni supprimée. Dans la mesure du possible, le pont essaye de réutiliser un attribut qui a le même nom que le nouvel attribut à utiliser.

Attributs génériques

Pour chaque niveau de déclinaison, il est possible de définir un nom générique.

Le pont générera alors automatiquement un deuxième groupe d'attributs quand ce nom est spécifié.

Cela est notamment utile pour gérer les regroupements de couleurs afin de pouvoir proposer des sélecteurs plus ergonomiques sur les sites.

Exemple:

- J'ai quatre couleurs :
 - "rouge", avec un nom générique "rouge",
 - "bordeaux", avec un nom générique "rouge",
 - "sable", avec un nom générique "jaune",
 - "moutarde", avec un nom générique "jaune",
- Si j'ai un produit qui se fait dans les 4 couleurs en taille TU, alors le pont créera les déclinaisons suivantes :
 - Produit - Taille: TU - Couleur: rouge - Couleur générique: rouge
 - Produit - Taille: TU - Couleur: bordeaux - Couleur générique: rouge
 - Produit - Taille: TU - Couleur: sable - Couleur générique: jaune
 - Produit - Taille: TU - Couleur: moutarde - Couleur générique: jaune

Le webconcepteur pourra alors prévoir un sélecteur "Couleur générique" dans la navigation à facette, et simplement afficher l'attribut "couleur" dans la page produit.

Notes sur la méthode de mise à jour des modèles

En cas de modification de l'une de ses composantes (classification, taille, prix, déclinaisons, niveaux, thèmes, ...), c'est **le modèle entier qui est republié** sur *Prestashop*, mis à jour dans sa globalité et non simplement la donnée qui a changée.

Par exemple, si vous changez la taille "42/44" par "42-44", la passerelle va renvoyer toutes les informations de tous modèles qui ont un article dans cette taille.

Le processus est plus long, mais cela permet d'éviter les incohérences et garanti une atomicité au niveau du produit et non de la taille : il est ainsi impossible d'obtenir des tailles ou déclinaisons manquantes ou désynchronisées pour les modèles. Soit le modèle est synchronisé et il l'est totalement (déclinaisons incluses), soit il ne l'est pas.

Utilisation du groupe de produits

Si l'équivalence "*modèle Polaris = produit Prestashop*" n'a pas été choisie pour la synchronisation, il est normalement impossible de savoir que tel ou tel produit existe en telle ou telle couleur avec les moyens traditionnels. En effet, si le découpage à lieu sur la couleur, il n'y aura jamais plus d'un attribut couleur dans un même produit Prestashop.

Pour palier à ce problème, la passerelle intègre un mécanisme permettant de retrouver cette information et de lier/connecter entre-eux les produits qui ne diffèrent que par des attributs qui ont été segmentés en les rendant connexes.

Les intégrateurs du site peuvent retrouver ce regroupement "modèle Polaris" à travers les groupes de produits.

Depuis la version 2 de la passerelle, il est possible de simplement générer un sélecteur sur les pages produits, en dessous du prix de l'article, ou bien où l'intégrateur souhaite en insérant le code suivant là où il souhaite générer le sélecteur : `{hook h="actionPolarisLinkProductSelector" product_id=$product->id}`.

Dans le cas où l'intégrateur souhaiterait un contrôle plus précis du système, il peut utiliser la méthode de la version 1, toujours d'actualité en manipulant directement le groupe de produits, comme indiqué ci-après.

Le groupe de produits s'obtient en appelant le hook `actionPolarisLinkLoadRelatedProducts` qui va peupler le template avec des informations supplémentaires liant les produits entre-eux.

Appel du hook	Documentation	Contenu des variables SMARTY créées	
		\$relatedProducts	\$relatedProductsAttributes
{hook h='actionPolarisLinkLoadRelatedProducts'}	<p>Sans argument, le hook fonctionne en automatique, et cela suffira dans la plupart des cas.</p> <p>Selon son placement dans le squelette, il autodétecte le ou les produits en cherchant respectivement dans les variables <code>\$product</code> et <code>\$products</code> de la portée en cours.</p>	Ensemble des produits Prestashop composant un même modèle Polaris (en règle général, un produits Prestashop par coloris)	
		Si <code>\$product</code> est détecté :	Product[] : tableau des produits connexes
		Si <code>\$products</code> est détecté:	Array[X=>Product[]] : Tableau associant chaque produit à ses produits connexes
{hook h='actionPolarisLinkLoadRelatedProducts' id_product=X}	Le hook calcule les produits connexes du produit dont le numéro est spécifié dans <code>id_product</code> .	Product[] : tableau des produits connexes	
			<p>Structure de données exposant les différents attributs existants pour chaque produits (voir exemple) :</p> <p><code>products</code> : liste, par produit (clé), des variations existantes. Cette liste est peuplée avec TOUS les attributs possibles pour CHAQUE produit (le produit rouge X existe également en vert, c'est le produit Y ; ou en bleu, c'est le produit Z).</p> <p><code>details</code> : détails des attributs par produit</p>

<pre>{hook h='actionPolarisLinkLoadRelatedProducts' id_products=[X, Y, Z]}</pre>	<p>Le hook calcule les produits connexes de chaque produit dont le numéro est spécifié dans le tableau <i>id_products</i>.</p>	<p>Array[X=>Product[]] :</p> <p>Tableau associant chaque produit à ses produits connexes</p>
--	--	---

Exemple : choix des coloris disponibles sur la fiche produit.

Dans l'exemple suivant, il s'agit, dans la fiche produit, de proposer à l'internaute une liste visuelle du même produit mais dans des coloris différents.

*Nota : si vous recopiez l'exemple tel quel, n'oubliez pas de créer le format d'image 'selection' avec une taille de 48px * 48px pour avoir un rendu confortable.*

Pour prestashop 1.6, la modification est à apporter sur la page `/themes/xxx/product.tpl`.

```
{*} remplacer id_attribute_color avec l'id du groupe d'attribut correspondant à votre
couleur {*}
{$id_attribute_color = 2}
{*} demande de peuplement de $relatedProducts {*}{hook
h='actionPolarisLinkLoadRelatedProducts'}
{*} s'il y a plusieurs produits connexes, affichage du choix à l'internaute {*}{if
count($relatedProducts) > 1}
<style>
.related-products-selector {
}
.related-products-selector li {
float: left;
font-size: 8pt;
margin-right: 4px;
}
.related-products-selector .active {
background: darkorange;
}
.related-products-selector .active a {
border: 1px solid darkorange;
}
.related-products-selector .active img {
opacity: 0.7;
}
.related-products-selector a {
display: block;
border: 1px solid #ccc;
border-radius: 24px;
width: 48px;
height: 48px;
line-height: 24px;
overflow: hidden;
text-align: center;
background: white;
text-decoration: none;
}
.related-products-selector a:hover,
```

```

.related-products-selector a:focus
{
border: 1px solid darkorange;
text-decoration: none;
outline:none;
}
</style>
{*} Marquer le nom de la couleur {*}{if $relatedProductsAttributes.details[$product-
>id][$id_attribute_color]}
<p>Couleur : <strong> {foreach from=$relatedProductsAttributes.details[$product-
>id][$id_attribute_color] item=attr}
    {$attr.name}
  /foreach}
</strong></p>
{/if}
{*} Positionner un sélecteur de variante {*}<p style="font-weight: bold;">Existe en :
</p>
<ul class="related-products-selector">{foreach from=$relatedProducts item=r_product}
{if $r_product->active} <li><a {if $r_product->id == $product->id}class="active" {/if}
href="{ $r_product->getLink()}" title="{ $r_product->name|capitalize|escape:'html':'UTF-
8'}">
    {assign var="id_cover" value=$r_product->getCoverWs()}getImageLink($r_product-
>link_rewrite, $id_cover, 'selection')|escape:'html':'UTF-8'}{/if}"
title="{ $r_product->name|capitalize|escape:'html':'UTF-8'}" alt="{ $r_product-
>name|capitalize|escape:'html':'UTF-8'}" />
  </a></li>
{/if}
{/foreach}
</ul>
<div style="clear: both; margin-bottom: 2ex;"></div>
{/if}

```

De même, si vous reprenez ce code source, il sera judicieux de cacher l'attribut couleur du sélecteur traditionnel, puisqu'il fait désormais double emploi. Pour se faire, insérer le bloc suivant au début de votre page *product.tpl*, ou dans les CSS de votre thème (remplacez le 6 par l'id de votre groupe d'attributs représentant la couleur) :

```

<style>
#product #group_6 { display: none; } #product label[for=group_6] { display: none
!important; }
</style>

```

En version 1.7 de Prestashop, le système de *template* a changé sous la version 1.7 : les objets de manipulation ne sont plus accessibles directement et un tableau avec les informations pertinentes a été généré pour les remplacer.

Le module suit cette méthodologie et expose le maximum d'informations disponibles et de la même manière que *\$product* (quasiment les mêmes que celles du produit).

La logique reste la même, on appelle un *hook* qui génère les informations sur les modèles connexes dans *\$relatedProducts*, puis on les utilise :

```

{hook h='actionPolarisLinkLoadRelatedProducts'}
{if count($relatedProducts) > 1}
<style>
.related-products-selector {

```

```

}
.related-products-selector li {
float: left;
font-size: 8pt;
margin-right: 4px;
}
.related-products-selector .active {
background: darkorange;
}
.related-products-selector .active a {
border: 1px solid darkorange;
}
.related-products-selector .active img {
opacity: 0.7;
}
.related-products-selector a {
display: block;
border: 1px solid #ccc;
border-radius: 24px;
width: 48px;
height: 48px;
line-height: 24px;
overflow: hidden;
text-align: center;
background: white;
text-decoration: none;
}
.related-products-selector a:hover,
.related-products-selector a:focus
{
border: 1px solid darkorange;
text-decoration: none;
outline:none;
}
</style>
<p style="font-weight: bold;">Existe en : </p><ul class="related-products-selector">
{foreach from=$relatedProducts item=r_product}
<li> <a {if $r_product.id_product == $product.id_product}class="active"{/if}
href="{ $r_product.url}" title="{ $r_product.name}">
{if empty($r_product.cover)}
{else}
{/if}
</a>
</li>
{/foreach}
</ul>
<div style="clear: both; margin-bottom: 2ex;"></div>
{/if}

```

Exemple : utilisation de `$relatedProductsAttributes`

`$relatedProductsAttributes` vous permet de lister tous les attributs disponibles sur des produits connexes.

L'exemple ci-dessous vous permet par exemple de générer une puce de couleur pour indiquer que chaque produit existe

également en telle ou telle couleur.

Si vous recopiez cet exemple, prenez garde à modifier la valeur ZZZ par le numéro de groupe représentant les couleurs.

La modification s'effectue dans le fichier /themes/xxx/product-list.tpl pour la version 1.6.

```
{*} à insérer au début du fichier {*}
{assign var='ZZZ' value='2'}
{hook h='actionPolarisLinkLoadRelatedProducts'}
{*} fin d'insertion {*}
{*} à insérer sous la photo d'un produit {*}{*} début : affichage pastilles de
couleurs disponibles avec PolarisLink {*} {assign var='attrsList'
value=$relatedProductsAttributes.products[$product.id_product]} {if
isset($attrsList[$ZZZ])}
<div class="color-list-container">
<ul class="color_to_pick_list clearfix">
{foreach from=$attrsList[$ZZZ] item=r_attr} {assign var='img_color_exists'
value=file_exists($relatedProductsAttributes.img_col_dir|cat:$r_attr.id_attribute|cat:
'.jpg')}
{if $img_color_exists || !empty($r_attr.color)}
<li> <a href="{ $r_attr.linkUrl|escape:'html':'UTF-8'}" class="color_pick" {if
!$img_color_exists && isset($r_attr.color) && $r_attr.color}
style="background: { $r_attr.color }; " {/if}>
{if $img_color_exists} 
{/if}
</a>
</li>
{/if}
{/foreach}
</ul>
</div>
{/if}
{*} fin : affichage pastilles de couleurs disponibles avec PolarisLink {*
```

Le thème par défaut de la version 1.7 de Prestashop ne s'y prête pas, mais un début pourrait-être :

Dans le fichier /themes/classic/templates/catalog/listing/product-list.tpl:

```
...
{block name='content'}
{hook h='actionPolarisLinkLoadRelatedProducts'}
<style>
.color-list-container ul li a {
display: block;
width: 18px;
height: 18px;
margin: 1px;
border-radius: 9px;
}
.color-list-container ul li {
display: inline-block;
border: 1px solid #d6d4d4;
border-radius: 11px;
width: 22px;
height: 22px;
```

```
margin: 1px;
}
</style>
...
```

Et dans le fichier /themes/classic/templates/catalog/_partials/miniatures/product.tpl:

```
{assign var='ZZZ' value='2'}{assign var='attrsList'
value=$relatedProductsAttributes.products[$product.id_product]}{if
isset($attrsList[$ZZZ])}
<div class="color-list-container">
  <ul class="color_to_pick_list clearfix">
    {foreach from=$attrsList[$ZZZ] item=r_attr} {assign var='img_color_exists'
value=file_exists($relatedProductsAttributes.img_col_dir|cat:$r_attr.id_attribute|cat:
'.jpg')}
    {if $img_color_exists || !empty($r_attr.color)}
      <li> <a href="{ $r_attr.linkUrl}" class="color_pick"{if !$img_color_exists &&
isset($r_attr.color) && $r_attr.color} style="background: $r_attr.color;"{/if}> {if
$img_color_exists} 
      {/if}
    </a>
    </li>
    {/if}
  {/foreach}
</ul>
</div>
{/if}
```

LES DONNÉES CONCERNÉES

Les données exportées par la passerelle

A minima, chaque entité prend en charge les données composantes de sa clé lorsqu'elle est exporté.

Les données prises en charge par la passerelle sont synthétisées dans le tableau ci-dessous.

#	Depuis la donnée sous Polaris	V1	Vers la donnée sous Prestashop	Notes
---	-------------------------------	----	--------------------------------	-------

1	Quantité en stock	1.10	<ul style="list-style-type: none"> Quantité manuelle ou Quantité de stock dans les dépôts en gestion avancée du stock (version 1.6 de Prestashop uniquement) 	<p>La quantité en stock peut-être :</p> <ol style="list-style-type: none"> la totalité du stock agrégé seul le stock des magasins publiés seul le stock des magasins non publiés seul le stock du magasin de référence² les options 2 à 4 sous forme de dépôt en gestion avancée du stock sous Prestashop <p>L'option 5 est automatiquement utilisée quand la gestion de stocks avancées a été activée sous Prestashop.</p> <p>quantité stock exportée = Stocks + (Transit +) - (Commandes client en cours)</p>
2	Prix de vente TTC	1.10	<p>La donnée est stockée dans deux champs :</p> <ul style="list-style-type: none"> une base en H.T. dans le prix du produit, un impact sur le prix dans la déclinaison. 	<p>Prix arrondi mathématiquement à deux décimales.</p> <p>Le prix de vente est converti en HT à la volée et un prix minimum est calculé pour le PV du produit.</p> <p>Les prix des déclinaisons sont ajustés en relatif autour de ce prix minimum.</p>
3	Prix d'achat HT	1.10	<p>La donnée est stockée dans deux champs :</p> <ul style="list-style-type: none"> une base dans le prix du produit, un impact sur le prix dans la déclinaison. 	<p>Prix arrondi mathématiquement à deux décimales.</p> <p>Un prix d'achat minimum est calculé pour le PA du produit.</p> <p>Les prix d'achat des déclinaisons sont ajustés en relatif autour de ce prix minimum.</p>
4	Prix TTC de soldes et promotions, par produit et par taille, du magasin de référence ²	1.00	Impact sur le prix, en %, dans les prix spécifiques des produits	<p>A raison d'un prix spécifique par taille, date et plan de soldes. Il n'y a pour le moment aucune différence entre soldes et promotions car Prestashop ne permet pas de typer les prix spécifiques.</p>
5	Code-barres EAN Polaris	1.20	Référence fournisseur <i>Polaris</i> ou <i>PolarisReprise</i>	
6	GENCODE	1.20	Champ EAN13 du modèle et des déclinaisons concernées	
7	Code modèle	1.20	Référence produit	
8	Code-barres EAN Polaris	1.20	Référence déclinaison	
9	Libellé modèle	1.20	Nom du produit	<p><i>Libellé Internet du modèle en priorité, si inexistant, alors libellé du modèle, si inexistant, en dernier recours, le code modèle. Les noms et libellés sont convertis en minuscules.</i></p> <p><i>Les caractères interdits sous Prestashop sont filtrés et remplacés par des équivalents valides.</i></p>

10	Photos produits, marques, & catégories.	1.20	Images correspondantes.	
12	<p>Membre :</p> <ol style="list-style-type: none"> civilité, nom & prénom, adresse mail, critère 1, "souhaite recevoir mailings", quarantaine 	1.20	<p>Client :</p> <ol style="list-style-type: none"> civilité, nom & prénom, adresse mail, date de naissance, opt-in, activé 	<p>Les caractères interdits sous Prestashop sont filtrés et remplacés par des équivalents valides.</p> <p>Le groupe de clients auxquels les clients créés sous Prestashop se rattache est configurable dans les options du module. Par défaut, il s'agit du groupe "Clients Polaris".</p>

¹Version de début de prise en charge : la donnée est prise en charge par la version du module d'extension spécifié et sous la dernière version de Polaris disponible.

Les données importées par la passerelle

#	Depuis la données sous Prestashop	V ¹	Vers la donnée Polaris	Notes
1001	<p>Clients</p> <ol style="list-style-type: none"> civilité, nom & prénom, adresse mail, date de naissance, opt-in 	1.20	<p>Membre :</p> <ol style="list-style-type: none"> civilité nom & prénom, adresse mail, "né le", "souhaite recevoir mailings" 	
1003	<p>Commande :</p> <ol style="list-style-type: none"> date de la commande client, détail des produits commandés et non annulés, montant des articles, réductions à l'article, réductions au panier, règles paniers utilisées 	1.20	<p>Débit :</p> <ol style="list-style-type: none"> date du débit, membre, détail de la vente, montant, remise article, remise caisse, bon utilisé, 	<p>La commande est importée sous Polaris dans la devise gérée par le lien ou ignorée si elle n'est pas de la même devise.</p> <p><i>Si des articles ne sont pas reconnus, Polaris importe la vente en utilisant une RFS d'équilibrage définie dans la configuration du site.</i></p> <p><i>Les remises panier Prestashop par code inconnues sur Polaris sont importées en tant que règlement de type "voucher".</i></p> <p><i>Celle sans code sont importées comme des règlements de type "remise_caisse".</i></p> <p><i>Par défaut, les règlements "voucher" sont importés comme des RB, les remises caisses comme des RE</i></p>

¹ Version de début de prise en charge : la donnée est prise en charge par la version du module d'extension spécifié et sous la dernière version de Polaris disponible.

Choix et restriction des données mises à jour sur Prestashop

Pour des raisons de sécurité ou de performance, les données suivantes peuvent ne pas être envoyées / rétribuées de Prestashop par configuration, soit sur Polaris, soit sur du module Prestashop.

Si l'envoi d'une donnée est activée sous Polaris mais pas sa réception sous Prestashop alors la donnée est présente dans le fichier de modifications, mais ne sera pas exploitée et modifiée sous Prestashop. A l'inverse, si la donnée a été désactivée sous Polaris et sa réception autorisée sous Prestashop, alors la donnée n'est pas présente dans le fichier de modifications et ne pourra pas être mise à jour par la passerelle.

Donnée concernée	Sens du flux	Nom de l'option		Notes
		Polaris	Prestashop	
Photos	Po ? Pr	-	Permettre l'importation des photos	<i>Si cette option est désactivée sur Prestashop, alors les photos ne seront ni envoyées, ni utilisées.</i>
Descriptions	Po ? Pr	-	Permettre la modification de la description des entités	<i>Quand cette option est désactivée, alors la description sera utilisée lors de la création de l'entité, mais plus mise à jour par la suite.</i>
Client (toutes données)	Po ? Pr	-	Permettre l'importation des clients	<i>Quand cette option est désactivée, le module ignore toutes les demandes de modification des entités client.</i>
Client (toutes données)	Pr ? Po	-	Permettre l'exportation des clients	<i>Quand cette option est désactivée, le module refuse de livrer les données clients et termine toute demande en ce sens avec une erreur de type 403 (forbidden). Les commandes exportées ne font alors pas référence à des clients et sont importées dans Polaris sans être attachées à un client.</i>
Commandes (toutes données)	Pr ? Po	-	Permettre l'exportation des ventes	<i>Quand cette option est désactivée, le module refuse de livrer les ventes et termine toute demande en ce sens avec une erreur de type 403 (forbidden).</i>
Quantité en stock	Po ? Pr	Synchroniser le stock	Permettre la mise à jour du stock	
Prix de vente	Po ? Pr	Synchroniser les prix de vente	Permettre la modification des prix	<i>Quand cette option est désactivée sur Prestashop, n'importe les prix qu'en création de l'entité.</i>
Prix d'achat	Po ? Pr	Exporter les PAHT	-	
Promotions	Po ? Pr	Exporter les soldes et promotions	Permettre l'importation des promotions	
Mots-clés	Po ? Pr	Gérer les mots-clés	-	<i>Si l'option est désactivée, alors n'envoie pas les entités mots-clés.</i>

Pourquoi les adresses ne sont pas synchronisées ?

Le règlement européen sur la protection des données personnelles (RGPD) guidant le développement de cette nouvelle passerelle, **les adresses ainsi que le numéro de téléphone portable ne sont pas synchronisées ni dans un sens, ni dans l'autre**, mais pourquoi ?

Avec le RGPD, la collecte des données doit répondre à une justification claire et exposée à l'internaute. La collecte ne doit avoir aucune autre raison que celles exposées à ce dernier. L'internaute doit en outre pouvoir refuser ou restreindre la collecte tout en continuant à bénéficier du service si aucune contrainte technique n'est levée.

En clair, pour pouvoir utiliser l'adresse postale d'un internaute commandant une paire de chaussures pour lui envoyer un mailing par la suite n'est autorisé par la loi que si :

1. il était informé de l'utilisation qui allait être faite de ses données lors de la collecte de l'adresse,
2. qu'il ait donné explicitement son accord dès la collecte,
3. et qu'il puisse refuser tout en entrant son adresse pour bénéficier de la livraison.

A ce jour, ces trois conditions ne peuvent pas être mis en place sous Prestashop sans module d'extension car il n'est pas possible de distinguer quel internaute a donné ou non son accord et pour quoi.

En absence de consentement explicite, il est réputé que l'internaute a saisi ses informations strictement pour voir le bon déroulement de sa livraison, et rien d'autre.

Hors, le traitement de la commande intervenant sous Prestashop, il n'y a aucune justification de rapatrier les coordonnées postales de l'internaute sous Polaris à des fins de traitement de service autrement que pour exploiter ces données différemment, ce qui n'est pas autorisé.

Prendre en charge la recopie de ces informations exposerait donc le marchand à de sévères réprimandes.

Il est également inutile de prendre en charge la recopie des adresses de Polaris vers le site car dans la majorité des cas si ce n'est dans tous, et en partant du fait où le client est également inscrit sur le site, c'est le site Internet qui dispose le plus rapidement de la bonne information. Envoyer les adresses reviendrait à prendre le risque d'écraser une bonne adresse avec une obsolète.

DOCUMENTATION TECHNIQUE DU MODULE

Variables de configuration

La configuration du module est sérialisée en *BinaryPHP* et stockée dans la table `polarislink_sites`.

Elle est composée d'un objets aux propriétés suivantes et chaque tuple de la table représente une instance de la passerelle.

Propriété	Type	Usage
LinkID	Entier	Numéro interne de la passerelle
LinkName	Chaîne	Nom de la passerelle configurée (pour la reconnaître)
Resync	Entier	1 pour transmettre une demande de resynchronisation générale à la prochaine connexion du client de la passerelle
SupplierID	Entier	Identifiant interne du fournisseur utilisé pour stocker les références
SupplierBackupID	Entier	Identifiant interne du fournisseur utilisé pour retrouver les références dont il faut récupérer le stock
LangID	Entier	Identifiant de la langue utilisée (<i>pas encore géré</i>)

ShopID	Entier	Identifiant du magasin géré (<i>pas encore géré</i>)
CurrencyID	Entier	Identifiant de la devise
ShopGroupID	Entier	Identifiant du groupe de magasins géré
MaxExecutionTime	Entier	Temps maximum, en seconde, d'exécution du script de traitement
AccessKey	Chaîne	Clé d'accès
AccessActive	Booléen	Clé active ou non ?
StockMini	Entier	Nombre de pièces à retirer de chaque article en guise de stock de sécurité
CustomersGroupID	Entier	Groupe de clients dans lequel ranger les clients créés par importation
OrderExportStateID	Tableau d'entiers	Identifiant des états de commandes à exporter vers Polaris. Lorsqu'une commande a déjà atteint au moins une fois un état présent dans cette liste, alors elle est exportée.
MethodTags	Entier	Méthode pour gérer les thèmes Polaris : 0 -> non gérés, 1 -> comme des catégories, 2 -> comme des caractéristiques, 3 -> comme des mots clés.
CategoryForTags	Entier	Si les thèmes sont gérés comme des catégories, catégorie parente des thèmes
DefaultCategoryRoot	Entier	Précise dans quelle catégorie racine de Prestashop est sélectionnée la catégorie par défaut du produit
CategoryForRFS	Entier	Catégorie parente des RFS
CategoryForCollection	Entier	Catégorie parente des collections
CategoryForSeason	Entier	Catégorie parente des saisons
CategoryForCategory1	Entier	Catégorie parente des catégories de type 1
CategoryForCategory2	Entier	Catégorie parente des catégories de type 2
CategoryForCategory3	Entier	Catégorie parente des catégories de type 3
CategoryForCategory4	Entier	Catégorie parente des catégories de type 4
AttributeForAttr0	Entier	Groupe d'attributs pour la taille
AttributeForAttr1	Entier	Groupe d'attributs pour les niveaux 1, 0 si non utilisé
AttributeForAttr2	Entier	Groupe d'attributs pour les niveaux 2, 0 si non utilisé
AttributeForAttr3	Entier	Groupe d'attributs pour les niveaux 3, 0 si non utilisé
AttributeForAttr4	Entier	Groupe d'attributs pour les niveaux 4, 0 si non utilisé
AttributeForAttrG1	Entier	Groupe d'attributs pour les niveaux 1, 0 si non utilisé
AttributeForAttrG2	Entier	Groupe d'attributs pour les niveaux génériques 2, 0 si non utilisé
AttributeForAttrG3	Entier	Groupe d'attributs pour les niveaux génériques 3, 0 si non utilisé

AttributeForAttrG4	Entier	Groupe d'attributs pour les niveaux génériques 4, 0 si non utilisé
FeatureForCollection	Entier	Caractéristique pour les collections, 0 si non utilisé
FeatureForSeason	Entier	Caractéristique pour les saisons, 0 si non utilisé
FeatureForAttr1	Entier	Caractéristique pour les niveaux 1, 0 si non utilisé
FeatureForAttr2	Entier	Caractéristique pour les niveaux 2, 0 si non utilisé
FeatureForAttr3	Entier	Caractéristique pour les niveaux 3, 0 si non utilisé
FeatureForAttr4	Entier	Caractéristique pour les niveaux 4, 0 si non utilisé
FeatureForAttrG1	Entier	Caractéristique pour les niveaux 1, 0 si non utilisé
FeatureForAttrG2	Entier	Caractéristique pour les niveaux 2, 0 si non utilisé
FeatureForAttrG3	Entier	Caractéristique pour les niveaux 3, 0 si non utilisé
FeatureForAttrG4	Entier	Caractéristique pour les niveaux 4, 0 si non utilisé
OPT_ACTIVE_PUBLISHED	Booléen	Activer automatiquement les produits publiés
OPT_AUTO_ACTIVE_ON_STOCK	Booléen	Paramètre la passerelle pour que la visibilité de chaque article soit modifiée en fonction de son stock
OPT_DELETE_WHEN_UNPUBLISHED	Entier	Supprimer les produits dépubliés au lieu de les désactiver
OPT_USE_ARCHIVES	Booléen	Conserve les fichiers d'importation envoyés au lieu de les détruire
OPT_USE_ADV_STOCK	Booléen	Utiliser l'option des stocks avancés
OPT_USE_EAN_STOCK	Booléen	Utiliser l'option d'identification des articles par leur gencode si la référence Polaris n'a rien donné
OPT_ALLOW_I_PHOTOS	Booléen	Autoriser l'importation des photos
OPT_ALLOW_I_DESCRIPTIONS	Booléen	Autoriser les modifications des descriptions par l'importateur
OPT_ALLOW_I_MDESCRIPTIONS	Booléen	Autoriser les modifications des méta-descriptions par l'importateur
OPT_ALLOW_I_CUSTOMERS	Booléen	Autoriser l'importation des clients
OPT_ALLOW_E_CUSTOMERS	Booléen	Autoriser l'exportation des clients
OPT_ALLOW_E_SALES	Booléen	Autoriser l'exportation des ventes
OPT_ALLOW_I_STOCKS	Booléen	Autoriser l'importation des stocks
OPT_ALLOW_I_SALE_PRICES	Booléen	Autoriser la mise à jour des prix de vente
OPT_ALLOW_I_PROMOS	Booléen	Autoriser l'importation des promos
OPT_ALLOW_I_CATALOG	Booléen	Autoriser la modification du catalogue produit
OPT_ALLOW_I_VOUCHERS_ASSETS	Booléen	Autoriser l'importation des bons d'avoir
OPT_ALLOW_I_VOUCHERS_OFFERS	Booléen	Autoriser l'importation des bons cadeaux, des bons de remises et chèques cadeaux
OPT_PROMO_AMOUNT	Booléen	Les promotions sont importées en montant de réduction et non en pourcentage lorsque vrai
OPT_NO_SYMLNK_IMG	Booléen	Faire une copie des photos dans le répertoire image de Prestashop en lieu et place d'un lien symbolique

Autres variables

Les variables de configuration standard suivantes sont créées et maintenues lors de l'exploitation du module :

Variable de configuration	Type	Usage
---------------------------	------	-------

VEGA_POLARIS_LINK_LAST_SYNC	Timestamp	Date de la dernière synchronisation (réussie ou échouée)
VEGA_POLARIS_LINK_LAST_ERRS	Entier	Nombre d'erreurs depuis la dernière remise à zéro des erreurs
POLARISLINK_ALLOW_MAINTENANCE_ACCESS	Booléen	Autorise l'accès à distance au backoffice Prestashop par les équipes de Polaris

Tables exploitées, champs ajoutés

Le module ajoute les tables suivantes :

Nom de la table	Champ	Usage
polarislink_jobs	Travaux de synchronisation en cours	
	id_link	Identifiant de la passerelle pour lequel ce mappage a été établi
	id_job	Identifiant interne du travail
	job	Nom du fichier contenant les instructions à propager
	priority	<i>Non utilisé</i>
	creation_date	Date de création du fichier de modifications
	rows_done	lignes déjà effectuées
	total_rows	nombre de ligne à faire
	err	Nombre d'erreurs générées lors du traitement de ce travail
polarislink_mails	Historique des adresses mails des internautes	
	id_mail	Identifiant interne de l'historique de mail
	id_customer	Identifiant Prestashop du client
	mail	adresse mail chiffrée
	mail_len	taille de l'adresse mail en nombre de caractères
polarislink_photos	Image miroir des photos Polaris envoyées par ce dernier	
	id_link	Identifiant de la passerelle pour lequel ce mappage a été établi
	id_photos	Identifiant interne de la photo sous Prestashop
	id_polaris	Numéro interne de la photo sous Polaris
	etag	Empreinte du fichier communiqué par Polaris
polarislink_products_group	Relation entre les produits prestashop et les modèles Polaris	
	id_group	Numéro interne du modèle Polaris représenté
	id_product	Identifiant du produit Prestashop concerné
polarislink_ref_category	Mappage entre les catégories Prestashop et les catégories Polaris	
	id_link	Identifiant de la passerelle pour lequel ce mappage a été établi
	id_ps	Identifiant Prestashop
	id_polaris	Équivalent Polaris
polarislink_ref_grtag_features	Mappage entre les groupes de mots-clé et les groupes de thèmes Polaris	
	id_link	Identifiant de la passerelle pour lequel ce mappage a été établi
	id_ps	Identifiant Prestashop

id_polaris	Équivalent Polaris	
polarislink_ref_manufacturer	Mappage entre les marques Prestashop et les marques Polaris	
	id_link	Identifiant de la passerelle pour lequel ce mappage a été établi
	id_ps	Identifiant Prestashop
	id_polaris	Équivalent Polaris
polarislink_ref_tag_features	Mappage entre les mots-clé Polaris et les mots-clé Prestashop	
	id_link	Identifiant de la passerelle pour lequel ce mappage a été établi
	id_ps	Identifiant Prestashop
	id_polaris	Équivalent Polaris
polarislink_sites	Configuration des instances de passerelle	
	id_link	Numéro de l'instance
	cfg	Configuration de la passerelle
	lockname	<p>Numéro de la connexion SQL.</p> <p>Si vide = aucun traitement en cours, sinon, un traitement est normalement en cours (la connexion SQL existe toujours) ou bien le traitement a échoué sans possibilité de terminer proprement (la connexion n'existe plus).</p> <p><i>Ce champ est utilisé pour fournir des fonctions d'exclusion de multi-exécution.</i></p>
polarislink_stocks	Quantités par magasin / boutique	
	id_product	Identifiant du produit concerné par cette référence de stock
	id_product_attribute	Identifiant de la déclinaison concernée
	id_store	Identifiant de la boutique concernée
	stock	Nombre de pièces en stock de la boutique
	date_upd	Date de la mise à jour de ce stock

FAQ

La clé est refusée avec le message **401 - Unauthorized**

La plupart du temps, cette erreur ne provient pas du module mais du serveur Web qui ne transmet pas l'identification HTTP.

En effet, PHP ne reçoit pas l'identification HTTP correctement et il s'agit probablement d'une mauvaise configuration du serveur. Les variables d'environnement `HTTP_AUTHORIZATION`, `REDIRECT_HTTP_AUTHORIZATION`, ou `PHP_AUTH_USER` et `PHP_AUTH_PW` doivent être définies.

Pour cela :

1. Vérifier qu'un espace n'a pas été ajouté à la clé lors du copier/coller de cette dernière ;
2. si le serveur utilise la technologie PHP par CGI (mod-cgi / php-cgi / fast-cgi ou php-fpm), vérifier que le mode "**Activer le mode CGI de PHP**" est bien coché dans les options Paramètres avancés > Webservice.

Si cela n'est pas suffisant, **pour contrôler que le système d'identification est correctement configuré, le webmestre est invité à mettre en place un accès au Webservice de Prestashop et contrôler qu'il fonctionne bien.** En effet, les erreurs et les correctifs à appliquer en cas d'échec sont très bien documentés par la communauté de Prestashop et accessible par simple recherche.

Si l'accès au Webservice fonctionne, mais pas celui du module, alors il s'agit très probablement d'un problème imputable au module et il est alors nécessaire de consulter le développeur.

Erreur SHOW PROCESSLIST

Pour éviter l'exécution simultanée de plusieurs processus de mise à jour, notre système verrou nécessite le bon fonctionnement de la commande mysql *SHOW PROCESSLIST*.

Vous obtenez cette erreur quand cette commande semble ne pas fonctionner correctement : il semblerait que certains hébergements tels que *1and1* aient modifié le comportement de certains de leur systèmes de base de données.

Nous ne sommes pas compatibles avec de tels hébergements.

Contactez votre hébergeur pour obtenir plus de renseignements ou un hébergement conforme aux spécifications annoncées.