

Descripción Interface WMS con diversos eCommerce

Documento elaborado por Altanet S.A.

Abril de 2024

1. Interacción con Prestashop

En esta sección se define la integración que se realiza entre el sistema Prestashop y el WMS Altanet Logística.

1.1. Requisitos previos

Para realizar la integración con Prestashop debemos solicitar lo siguiente:

- URL base de la API de Prestashop. Por ejemplo, <https://altanet.cl/api>.
- Token que debe ser generado directamente en la plataforma de Prestashop.

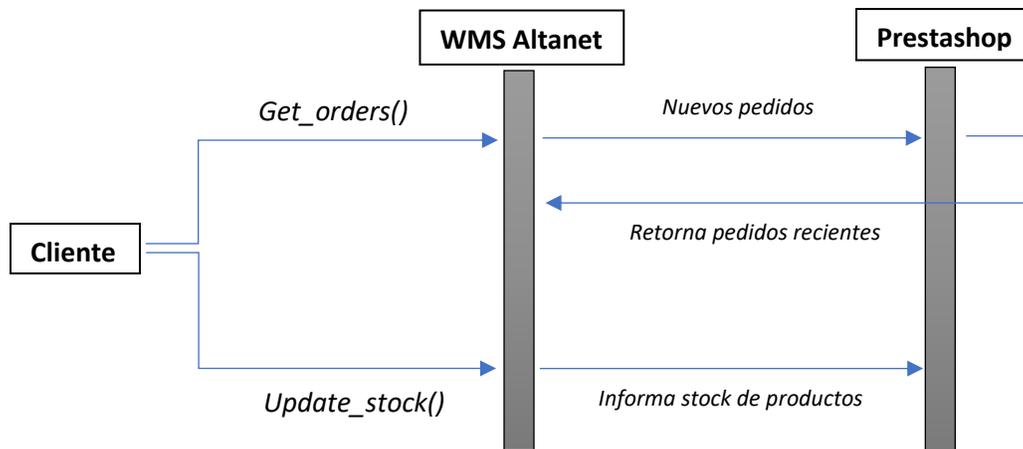
Es necesario ingresar los datos anteriormente solicitados en el código fuente de la API que existe para la comunicación entre el WMS y Prestashop. Se debe buscar las variables en el código fuente llamadas “**baseUrl**” y “**account**”. Donde:

- **baseUrl**: Corresponde a la URL que tiene el comercio asignado por Prestashop.
- **account**: Corresponde al Token generado en Prestashop para conexiones con externos.

1.2. Modo de Operación

Para que exista una comunicación entre WMS Altanet y Prestashop existe dos endpoints que permite la interacción entre ambos sistemas. Estos endpoints se detallan de la siguiente manera:

- **Get_orders**: Este endpoint recibe como parámetros la fecha de inicio y termino de la búsqueda de pedidos recientes en Prestashop, la empresa y la bodega en la cual se van a generar los pedidos en el WMS (las ordenes de despacho generadas).
- **Update_stock**: Este endpoint recibe como parámetros el id único del producto que se quiere actualizar su stock y por otra parte recibe el stock actual que existe en el WMS.



1.3. Automatización

Para que la interacción entre WMS Altanet y Prestashop ocurra de manera autónoma nos apoyaremos de una herramienta para planificación y/o programación de tareas que tiene SQL, como lo son los JOBS. En este caso vamos a generar un JOB que nos permita consultar periódicamente los Endpoints que nos permiten la interacción entre ambos sistemas. Es necesario tener conocimientos en programación SQL para realizar este proceso.

Lo primero será generar el STEP del JOB. En este apartado vamos a generar el código SQL que nos permite realizar el llamado a cualquiera de los Endpoints que necesitemos automatizar. Vamos a definir las siguientes variables:

- **@fecha_inicio varchar(20) = convert(varchar(20), DATEADD(day, -5, getdate()),120)**
Fecha de inicio de búsqueda de pedidos.
- **@fecha_termino varchar(20) = convert(varchar(20),getdate(),120)**
Fecha de término de búsqueda de pedidos.
- **@Url varchar(1024)**
Ruta donde se encuentra nuestro llamado al endpoint.
- **@HTTPConeccion int**
Variable que ocuparemos para realizar el proceso de llamado HTTP.

Posteriormente configuramos el SCHEDULES del JOB que básicamente será el tiempo en el que se ejecutara cada llamado al endpoint con la finalidad que sea un proceso totalmente automático. Para este caso quisimos que el llamado se realizara cada 10 minutos todos los días hasta que decidamos dar termino al proceso automatizado. Lo anterior se debería ver como:

The screenshot displays the configuration for a SQL Server Job Schedule. It is divided into several sections:

- Frequency:** Occurs: Daily (dropdown), Recurs every: 1 day(s) (spinners).
- Daily frequency:** Occurs once at: 0:00:00 (spinners), Occurs every: 10 minute(s) (spinners and dropdown), Starting at: 0:00:00 (spinners), Ending at: 23:59:59 (spinners).
- Duration:** Start date: 11-04-2023 (calendar icon), End date: 11-04-2024 (calendar icon), No end date: (selected radio button).
- Summary:** Description: Occurs every day every 10 minute(s) between 0:00:00 and 23:59:59. Schedule will be used starting on 11-04-2023. (text area with scrollbars).

2. Interacción con Shopify

En esta sección se define la integración que se realiza entre el sistema Shopify y el WMS Altanet Logística.

2.1. Requisitos previos

Para realizar la integración con Shopify debemos solicitar lo siguiente:

- URL base de la API de Shopify. Por ejemplo, altanet.myshopify.com.
- Token que debe ser generado directamente en la plataforma de Shopify.

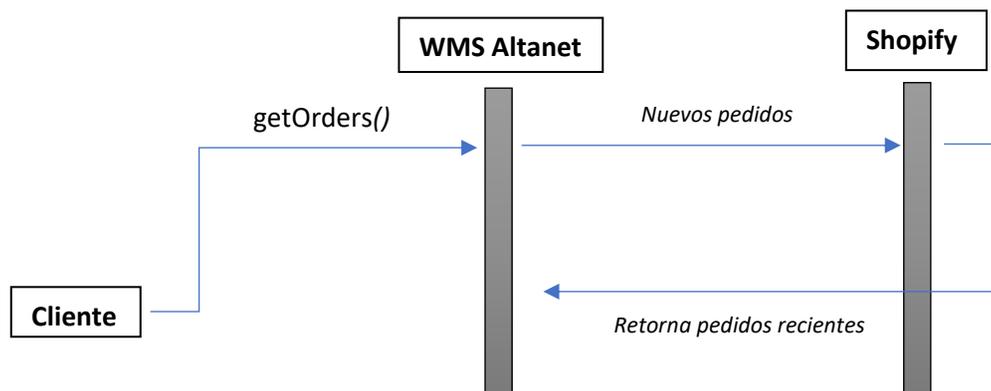
Es necesario ingresar los datos anteriormente solicitados en el código fuente de la API que existe para la comunicación entre el WMS y Shopify. Se debe buscar la función `getURLShop()` y se debe generar un parámetro en el sistema WMS llamado **PASS_SHOPIFY**. Donde:

- **getURLShop()**: Corresponde a la función que nos trae la URL asignada para la tienda en Shopify.
- **PASS_SHOPIFY**: Corresponde al Token generado en Shopify para conexiones con externos.

2.2. Modo de Operación

Para que exista una comunicación entre WMS Altanet y Shopify existe un endpoints que permite la interacción entre ambos sistemas. Este endpoint se detalla de la siguiente manera:

- **getOrders**: Este endpoint recibe como parámetros la empresa y la bodega del WMS en la cual se van a generar las ordenes de despacho de la búsqueda de pedidos recientes en Shopify.



2.3. Automatización

Para que la interacción entre WMS Altanet y Shopify ocurra de manera autónoma nos apoyaremos de una herramienta para planificación y/o programación de tareas que tiene SQL, como lo son los JOBS. En este caso vamos a generar un JOB que nos permita consultar periódicamente los Endpoints que nos permiten la interacción entre ambos sistemas. Es necesario tener conocimientos en programación SQL para realizar este proceso.

Lo primero será generar el STEP del JOB. En este apartado vamos a generar el código SQL que nos permite realizar el llamado a cualquiera de los Endpoints que necesitemos automatizar. Vamos a definir las siguientes variables:

- **@fecha_inicio varchar(20) = convert(varchar(20), DATEADD(day, -5, getdate()),120)**
Fecha de inicio de búsqueda de pedidos.
- **@fecha_termino varchar(20) = convert(varchar(20),getdate(),120)**
Fecha de término de búsqueda de pedidos.
- **@Url varchar(1024)**
Ruta donde se encuentra nuestro llamado al endpoint
- **@HTTPConeccion int**
Variable que ocuparemos para realizar el proceso de llamado HTTP.

Posteriormente configuramos el SCHEDULES del JOB que básicamente será el tiempo en el que se ejecutara cada llamado al endpoint con la finalidad que sea un proceso totalmente automático. Para este caso quisimos que el llamado se realizara cada 10 minutos todos los días hasta que decidamos dar termino al proceso automatizado. Lo anterior se debería ver como:

The screenshot shows the configuration for a job schedule in SQL Server Enterprise Manager. It is divided into several sections:

- Frequency:** Occurs: Daily; Recurs every: 1 day(s).
- Daily frequency:** Occurs every: 10 minute(s); Starting at: 0:00:00; Ending at: 23:59:59.
- Duration:** Start date: 11-04-2023; End date: 11-04-2024; No end date: selected.
- Summary:** Description: Occurs every day every 10 minute(s) between 0:00:00 and 23:59:59. Schedule will be used starting on 11-04-2023.

3. Interacción con Multivende

En esta sección se define la integración que se realiza entre el sistema Multivende y el WMS Altanet Logística.

3.1. Requisitos previos

Para realizar la integración con Multivende necesitamos lo siguiente:

- **Client Id:** Corresponde al id de cliente que otorga Multivende.
- **Merchant id:** Corresponde al id de la tienda que otorga Multivende.

Posteriormente se debe realizar una capacitación y certificación por Multivende, la cual al momento de aprobar nos entregaran las siguientes credenciales:

- **Client Secret.**
- **Code.**

Con las credenciales previamente obtenidas, podemos realizar solicitud de **Token** cada vez que lo necesitemos. Con el **Token** y **el Merchant id** podemos realizar consultas sobre nuevos pedidos, por eso es fundamental obtener las credenciales necesarias para realizar la interacción con Multivende.

3.2. Modo de Operación

Para que exista una comunicación entre WMS Altanet y Multivende existe dos endpoints que permite la interacción entre ambos sistemas. Estos endpoints se detallan de la siguiente manera:

- **ObtieneTokenMV():** Este endpoint nos permite obtener Token de Multivende. Con este token podemos realizar consultas de pedidos. Este token que obtenemos tiene fecha de caducidad por lo tanto debemos volver a pedir token. Sin embargo, existe una función que nos permite refrescar dicho token, este proceso lo hace de manera automática el sistema cuando sea necesario.

La ruta a la cual se solicita token corresponde a la siguiente:

<https://app.multivende.com/oauth/access-token>

- **TraeVentas():** Este endpoint nos permite realizar consulta de nuevos pedidos en un rango de fecha. Para realizar este proceso es requerido que tengamos un token activo y además el Merchant id, sin estas credenciales Multivende no nos permite realizar la consulta de datos.

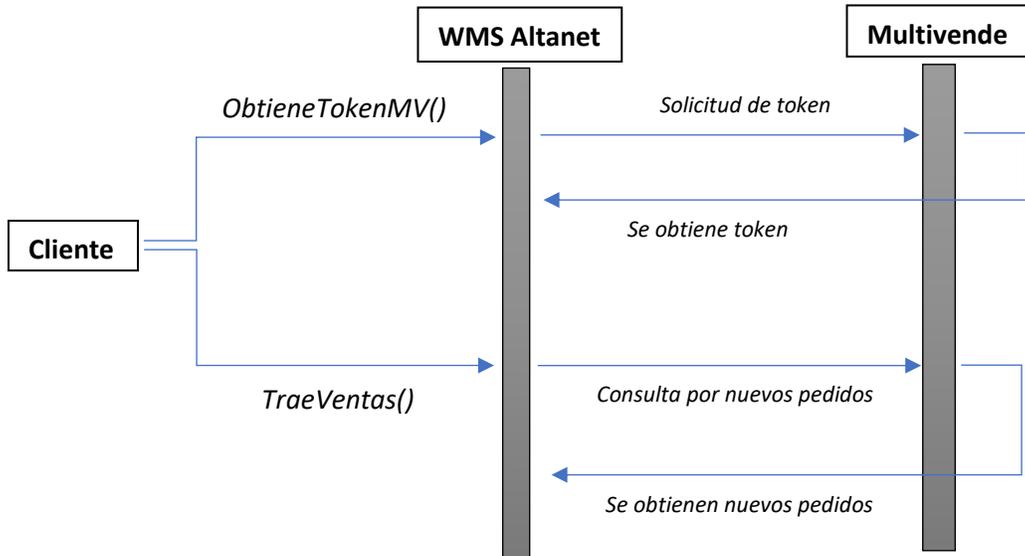
La ruta en Multivende para realizar la consulta de pedidos es la siguiente:

[https://app.multivende.com/api/m/Merchant id/checkouts/light/p/1?sold at from=Fecha desde& sold at to=Fecha hasta](https://app.multivende.com/api/m/Merchant_id/checkouts/light/p/1?sold_at from=Fecha desde& sold_at to=Fecha hasta)

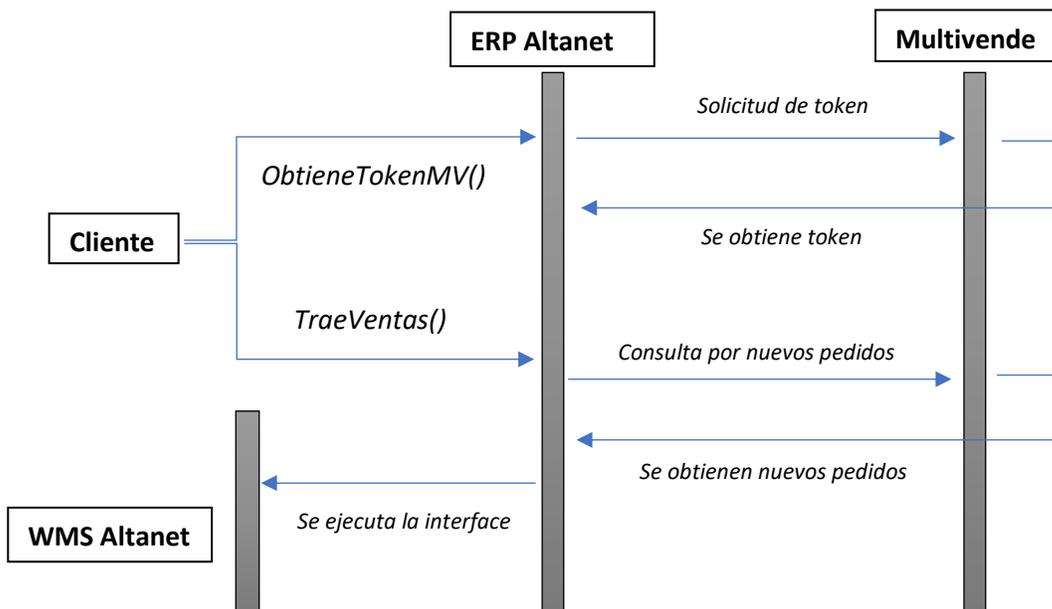
En la ruta anterior obtenemos el id único que asigna Multivende a los pedidos. Con los ids de los pedidos podemos proceder a llamar a la función que cargará las ventas en el WMS, esta función se llama **CargaVentasMV()**. Esta función consulta el detalle de cada pedido en la siguiente ruta de Multivende:

https://app.multivende.com/api/checkouts/id_pedido

Luego que obtenemos el detalle de los pedidos, estos se van grabando en el WMS como orden de despacho.



También existe la posibilidad que los pedidos caigan al ERP Altanet. Basicamente es el mismo proceso explicado anteriormente pero los pedidos caen en primera instancia al ERP y luego por la ejecución de la interface se generan en el WMS. Este proceso lo explicamos con el siguiente diagrama:



3.3. Automatización

Para que la interacción entre WMS Altanet y Shopify ocurra de manera autónoma nos apoyaremos de una herramienta para planificación y/o programación de tareas que tiene Windows, como lo son los Task Schedule. En este caso vamos a generar una tarea programada que nos permita consultar periódicamente los Endpoints que nos permiten la interacción entre ambos sistemas. Es necesario tener conocimientos sobre este proceso.

La programación y tiempo de ejecución se debería ver así:

Trigger	Details	Status
Daily	At 1:00 PM every day - After triggered, repeat every 10 minutes i...	Enabled

Por otra parte para ejecutar el llamado y consultar sobre nuevos pedidos disponibles, vamos a generar un archivo **.bat** con la ruta al endpoint que realiza la consulta.

Este archivo debería ser como:

```
1 start chrome.exe
2 http://altanet/usuario/WSMultivende/Service1.svc/TraeVentas
3 TIMEOUT /T 420>NUL
4 TASKKILL /IM chrome.exe /F
```

El código anterior se ejecutara periódicamente consultando sobre nuevo pedidos, esto se ejecutara con el apoyo del navegador Chrome.

4. Interacción con Woocommerce

En esta sección se define la integración que se realiza entre el sistema Woocommerce y el WMS Altanet Logística.

4.1. Requisitos previos

Para realizar la integración con Woocommerce necesitamos lo siguiente:

- **Link acceso:** Corresponde al link de acceso de cliente que otorga Woocommerce.
- **Costum Key:** Corresponde al key de cliente que otorga Woocommerce.
- **Costum Secret:** Corresponde a la clave secreta de cliente que otorga Woocommerce.

Con las credenciales previamente señaladas podemos consultar el endpoint que nos traerá los pedidos nuevos.

4.2. Modo de Operación

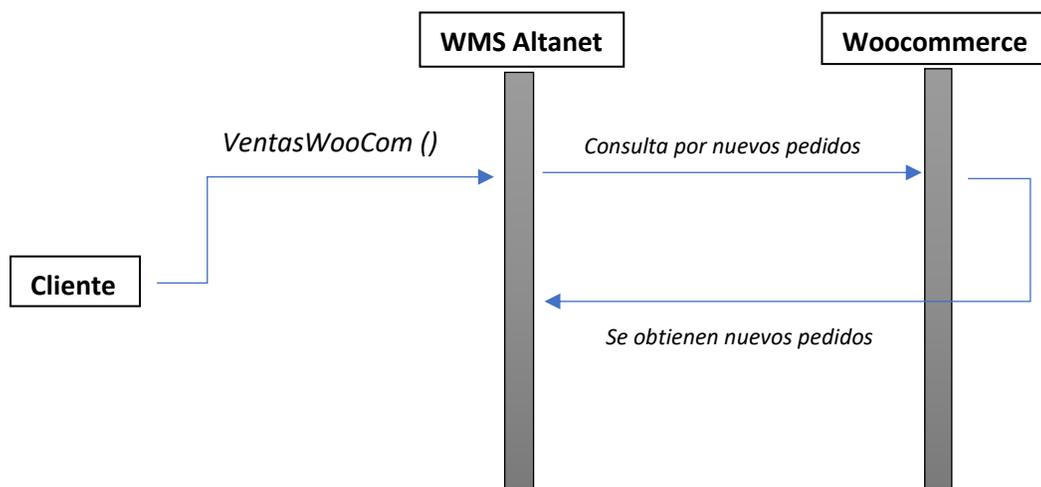
Para que exista una comunicación entre WMS Altanet y Woocommerce existe un endpoint que permite la interacción entre ambos sistemas. Este endpoint se detalla de la siguiente manera:

- **VentasWooCom():** Este endpoint nos permite realizar consulta de nuevos pedidos en un rango de fecha. Para realizar este proceso es requerido que tengamos Link de acceso, Costum Key y Costum Secret. Sin estas credenciales Woocommerce no nos permite realizar la consulta de datos.

La ruta en Woocommerce para realizar la consulta de pedidos es la siguiente:

[LinkAcceso/wp-json/wc/v3/](#)

Luego que obtenemos el detalle de los pedidos, estos se van grabando en el WMS como orden de despacho.



4.3. Automatización

Para que la interacción entre WMS Altanet y Shopify ocurra de manera autónoma nos apoyaremos de una herramienta para planificación y/o programación de tareas que tiene Windows, como lo son los Task Schedule. En este caso vamos a generar una tarea programada que nos permita consultar periódicamente los Endpoints que nos permiten la interacción entre ambos sistemas. Es necesario tener conocimientos sobre este proceso.

La programación y tiempo de ejecución se debería ver así:

Trigger	Details	Status
Daily	At 1:00 PM every day - After triggered, repeat every 10 minutes i...	Enabled

Por otra parte para ejecutar el llamado y consultar sobre nuevos pedidos disponibles, vamos a generar un archivo **.bat** con la ruta al endpoint que realiza la consulta.

Este archivo debería ser como:

```
1 start chrome.exe
2 http://altanet/usuario/WSMultivende/Service1.svc/TraeVentas
3 TIMEOUT /T 420>NUL
4 TASKKILL /IM chrome.exe /F
```

El código anterior se ejecutara periódicamente consultando sobre nuevo pedidos, esto se ejecutara con el apoyo del navegador Chrome.