**¿Cómo integrar el ERP con Magento?**

MAGENTO

ERP

MULESOFT

(Middleware)

La generación de tokens es el **primer paso** para obtener acceso a las API de magento. El token debe pasarse con cada llamada a la API como parte de la información del encabezado.

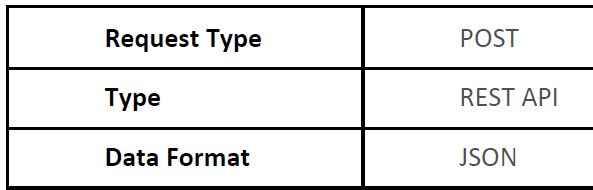
URL:

**Ambiente Desarrollo:**

<http://development-argentina-a6vkxsy-aba4j235kr5wu.us-3.magentosite.cloud/index.php/rest/V1/integration/admin/token>

**Ambiente Pruebas:**

<https://mcstaging.unlb2bcomm1.unileverservices.com/index.php/rest/V1/integration/admin/token>



Username: sigmaadmin

Password: s1gma3rp

**Sample Request**

{

“username” : “sigmaadmin”,

“password” : “s1gma3rp”

}

**Sample Response**

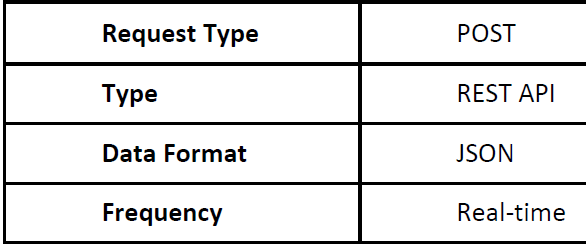
“Okebozau21af94c2g4l1cyo4b9deauet”

1. **Retailer account Interface - Interfaz de registro del retailer:**

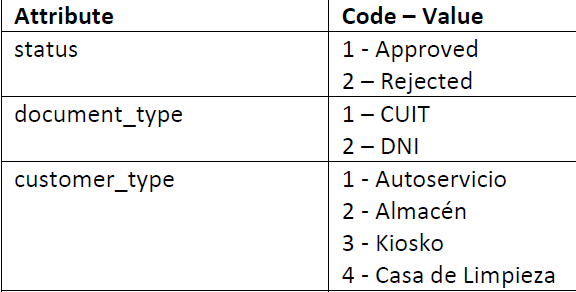
Esta API consiste en obtener las cuentas de los retailer que se dan de alta en Magento. Magento compartirá estos datos de registro y el ERP deberá devolver un codigo y un retailer ID en el momento que se apruebe la cuenta.

En caso de que el ERP rechace la cuenta, deberá devolver a Magento un código y motivo de rechazo.

**URL:** Magento is going to consume the API and URL needs to be come from Mulesoft team.



Los campos por devolver del ERP a Magento:







**2.1- Retailer Account Status Interface – (Approve / Reject)**

Esta API es para enviar el estado de la cuenta del minorista (ya registrado en el Distribuidor) desde el ERP hasta Magento.

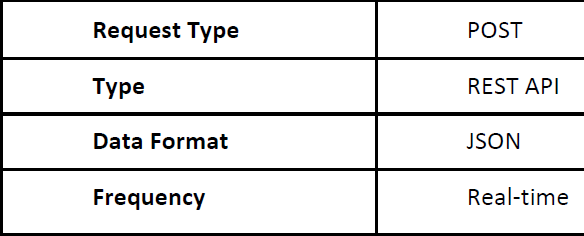
En caso de aprobación, el ERP necesita compartir el estado y la identificación del minorista en magento, que será almacenado para respaldar cualquier transacción futura. Opcionalmente, el ERP compartirá también los siguientes detalles:

* Crédito de cliente
* Crédito total disponible del cliente
* Pago pendiente del cliente

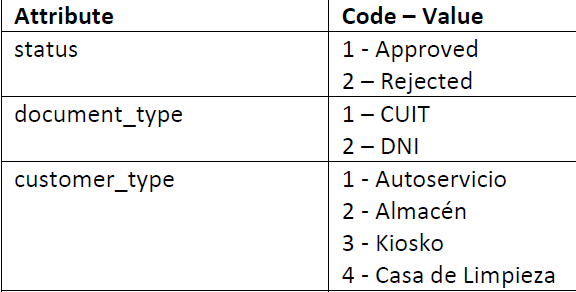
En caso de rechazo, ERP necesita compartir el estado y el motivo de rechazo a Magento que eliminara la asignación entre el minorista y el distribuidor. El minorista rechazado verá el catálogo global sin detalles de precios y no podrá realizar ninguna transacción en el sitio web.

Vamos a obtener los siguientes campos en la respuesta que envie el ERP:

* Retailer\_id
* Estado
* Motivo de rechazo



**URL:** <http://development-argentina-a6vkxsy-aba4j235kr5wu.us-3.magentosite.cloud/index.php/rest/V1/customers/:customerId>



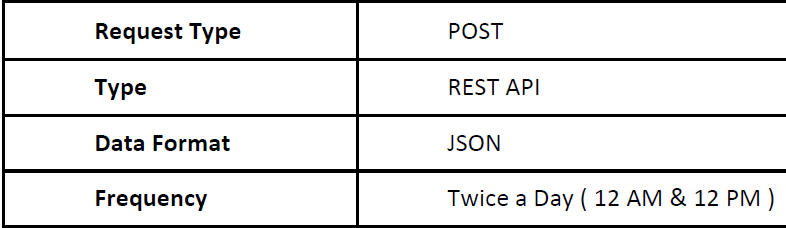


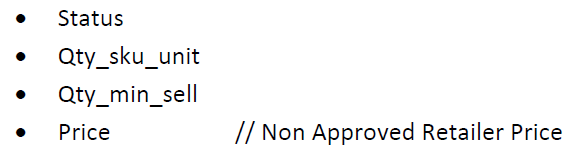
1. **Catalog Interface - Interfaz de productos:**

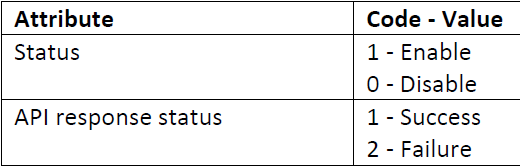
Cada distribuidor administrara su propio catálogo de productos. La selección de productos se realizará en el ERP del Distribuidor. El distribuidor puede habilitar/deshabilitar un conjunto de productos de su catálogo. En caso de que haya un EAN invalido, el administrador de Unilever accionara sobre el mismo.

La interfaz del catálogo tiene precios predeterminados para "Precio del retailer no aprobado"

**URL:** http://development-argentina-a6vkxsy-aba4j235kr5wu.us-3.magentosite.cloud/index.php/rest/ V1/products









1. **Price Interface - Interfaz de precio:**

La API de precio va a actualizar el precio base y el precio especial desde el ERP hacia Magento. En caso de que un sku no exista, el Admin lo revisara offline.

**Hay 4 configuraciones definidas para actualizar precio:**

* Precio base por distribuidor
* Precio base por grupo de distribuidor
* Precio global para todos los minoristas registrados
* Precio global para minorista no aprobado

**Configuración de precios de distribuidores aprobados:**

* Configurar precio por distribuidor
* Configurar precio por grupo de distribuidor
* Configurar precio para todos los distribuidores

El sistema tiene que elegir el precio en la siguiente secuencia, independientemente del precio mínimo.

1. Verifica el precio específico del distribuidor (consulta también los días de semana)

2. Verifica el precio por grupo de distribuidor (consulta también los días de semana)

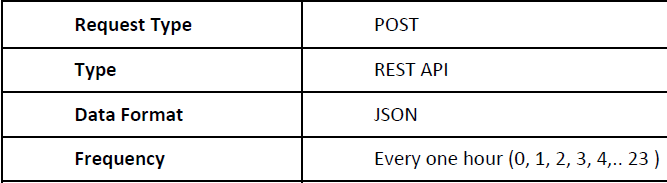
3. Finalmente verifica el precio global (consulta también los días de semana)

**NOTA:** Los comercios no aprobados solo tendrán precio base (sin precio especial). Se puede manejar un precio base diferente para comercios aprobados y no aprobados

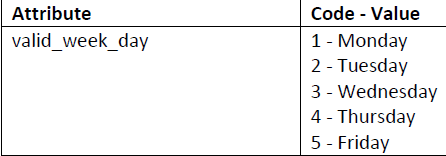
**Cada vez que el ERP impulse nuevos precios, Magento eliminará los precios anteriores para esos SKU y configurará un nuevo conjunto.**

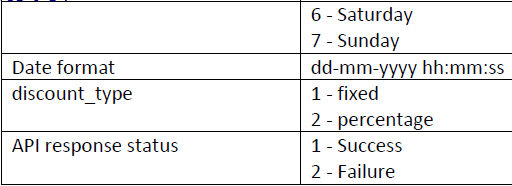
**El ERP necesita compartir TODA la configuración de precios para los SKU en un mismo envío**. No admitimos ninguna actualización de precios delta para un solo sku. El sistema no admite múltiples solicitudes para configurar el precio y el precio especial, todo debe pasar en el mismo envío de la interfaz.

**URL**: http://development-argentina-a6vkxsy-aba4j235kr5wu.us-3.magentosite.cloud/index.php/rest/ V1/products/distributor-prices



**Mapeo de datos:**





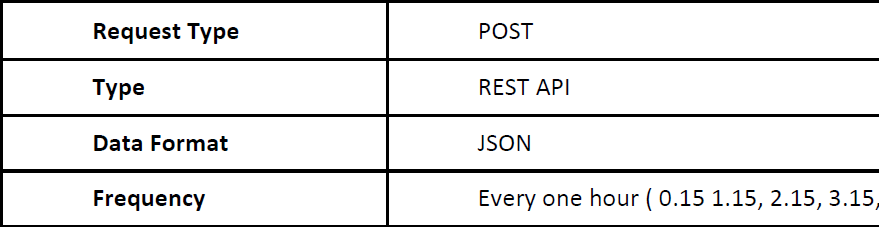


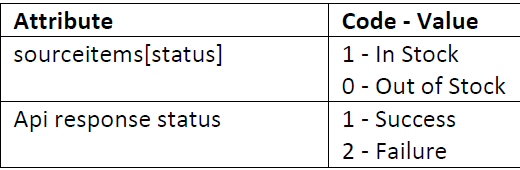
1. **Catalog Inventory Updates – Inventario**

Esta API de actualizaciones de inventario se utilizará para actualizar el inventario de múltiples SKU del sistema ERP a Magento de una sola vez.

Los datos de cada solicitud reemplazarán la información de inventario anterior de sku específico en magento.

**URL**: http://development-argentina-a6vkxsy-aba4j235kr5wu.us-3.magentosite.cloud/index.php/rest/ V1/inventory/source-items



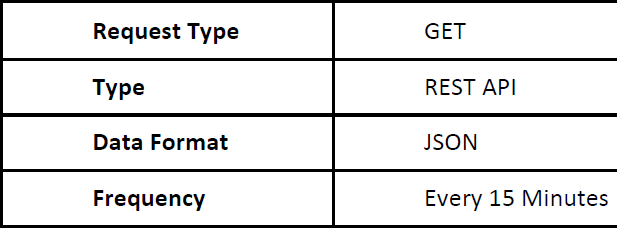


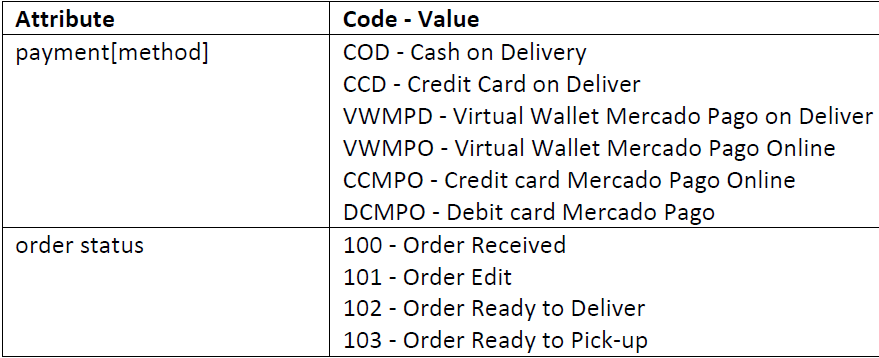
********

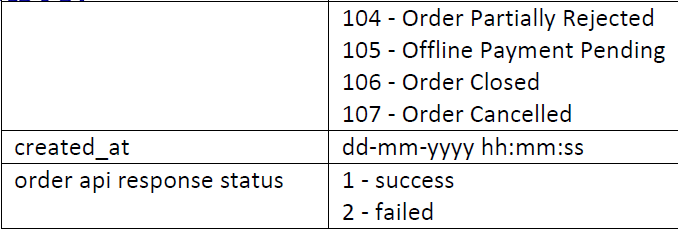
1. **Sales Order - Pedidos**

Esta API de pedidos de ventas se utilizará para enviar todos los pedidos nuevos de magento al ERP del distribuidor. Magento exportara los pedidos con un valor predeterminado de 1 siempre que sea nuevo. El sistema Magento capturará todos los pedidos con flag = 1 y enviará los detalles al ERP y actualizará el pedido con flag = 2 en caso de transferencia exitosa. El cron programado de magento recopilará y enviará todos los pedidos cada 15 minutos. En caso de fallo (ERP inactivo) el cron de magento recopilará todos los pedidos anteriores y reintentará enviar al ERP en la próxima llamada.

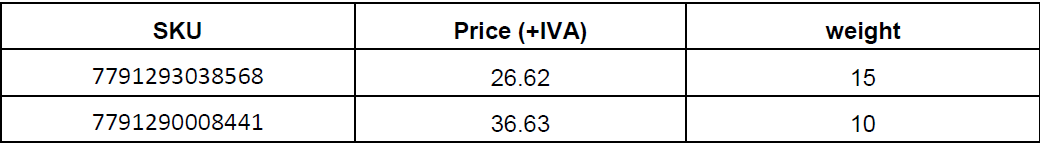
URL: URL should come from Mulesoft.



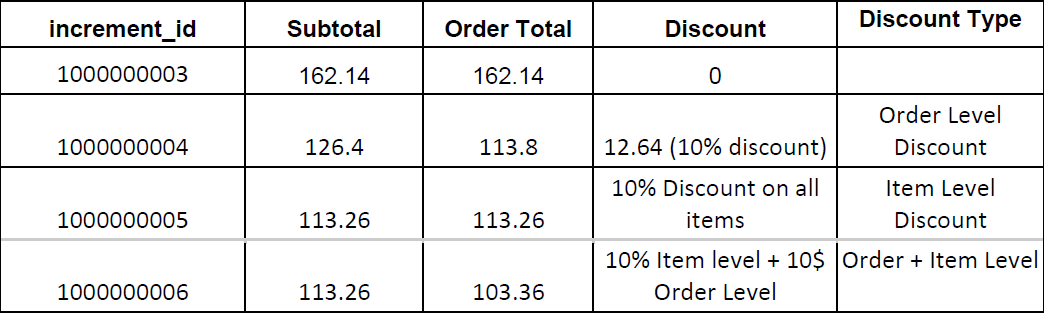




Ejemplo de producto



Ejemplo de pedido





1. **Sales Order Status Update – Actualización del estado del pedido (ERP a Magento)**

Esta API se utilizará para actualizar el estado del pedido desde el ERP hacia Magento.

Cada pedido tiene campos obligatorios que deben pasarse a Magento para actualizar con éxito. La actualización es en tiempo real cada vez que el distribuidor actualiza el estado del pedido en el ERP.

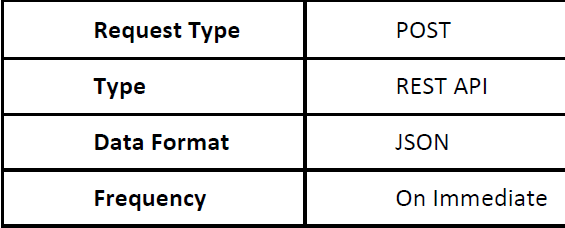
Vamos a gestionar también todos los pedidos sin conexión a través de esta interfaz. A continuación, se detallan las condiciones previas a cumplir para administrar la importación de pedidos offline en Magento

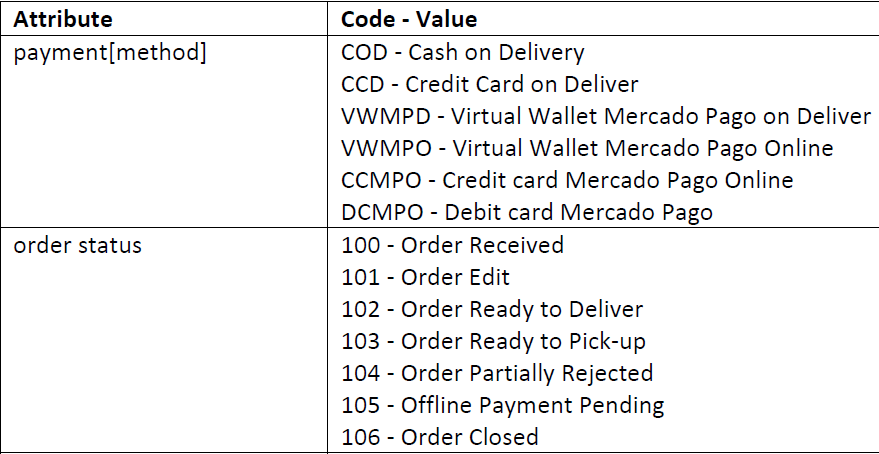
* El ERP enviará pedidos fuera de línea a Magento con todos los detalles del pedido (sku, qty, precio, precio total, etc.), retailer\_id, order\_source {smarttrader}, {PrestaShop}, {Axxum}, {otros} e increment\_id va a estar vacío

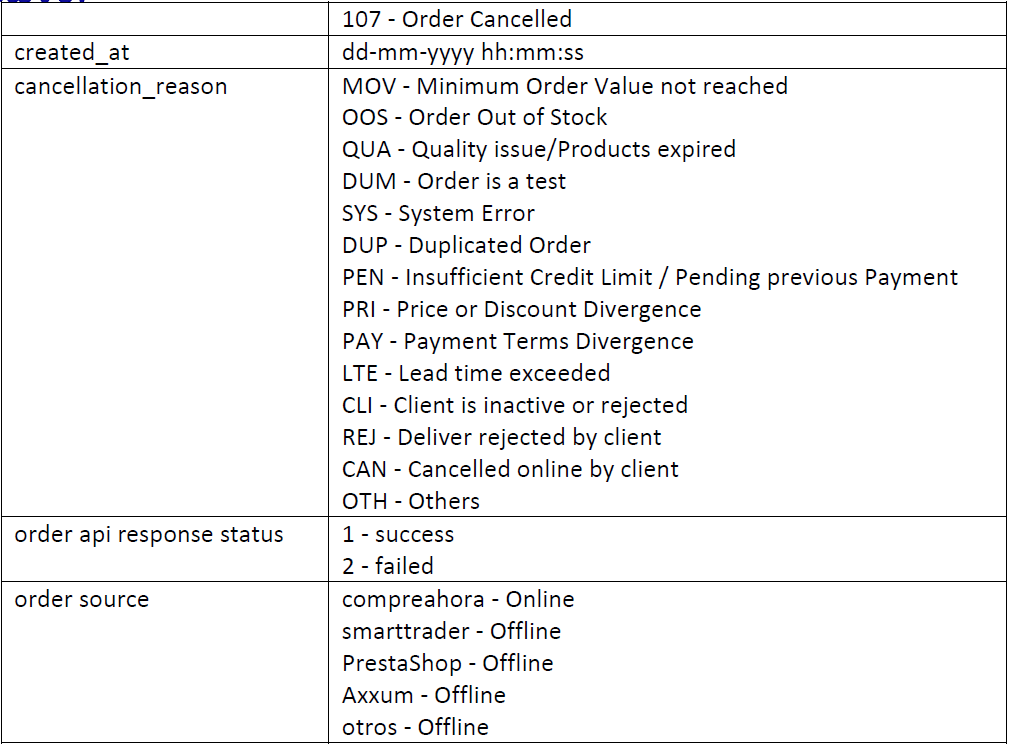
Magento importara las ordenes offline solo si:

* Existe el Retailer Id en Magento
* Existe el sku
* El producto debe tener la cantidad suficiente ya que al importar va a reducir esa cantidad del stock
* Magento enviara “increment\_Id” como respuesta al ERP con el detalle de todo el pedido

**URL:** <http://development-argentina-a6vkxsy-aba4j235kr5wu.us-3.magentosite.cloud/index.php/rest/V1/orders>









1. **Sales Order Status Update – Actualización del estado del pedido (Magento al ERP)**

Esta API es para actualizar el estado del pedido desde Magento al ERP siempre que el minorista cancele el pedido desde la sección Mis pedidos o el Distribuidor gestione el pedido

de magento en caso de falla de la interfaz. El minorista puede cancelar el pedido en caso de que el pedido esté en estado de "pedido recibido". El sistema no permite cancelar el pedido si el estado es diferente al de pedido recibido.

URL: URL should come from Mulesoft.

