GPI SOFTWARE · SUPPLY SYNCRO

Documentación API Supply Syncro

para clientes

Versión: 1.12

Fecha de publicación: Octubre 2026

Formato base: https://api.<CLIENTE>.supply-syncro.cl

Tabla de contenidos

1. Descripción API 4
1.1. Trabajar con Swagger 4
1.1.1. Autenticación 4
1.1.2. Ejemplo de llamada 5
1.1.3. Códigos de salidas 7
1.2. Formato de respuestas 7
1.2.1. Respuesta exitosa de POST 7
1.2.2. Respuesta exitosa de DELETE 8
1.2.3. Respuestas de error 8
1.3. Carga masiva (bulk) en POST 9
2. Métodos de carga de datos 11
2.1. Demanda Despacho 11
2.2. Inventario Onhand 14
2.3. Inventario tránsito desde proveedores 17
2.3.6. Truncar (operación destructiva) 20
2.4. Inventario tránsito entre sucursales 21
2.4.6. Truncar (operación destructiva) 25
2.5. Lead Times Internos 25
2.5.6. Truncar (operación destructiva) 27
2.6. Relación productos-proveedores 28
2.6.6. Truncar (operación destructiva) 30
2.7. Maestro Productos 31
2.7.6. Truncar (operación destructiva) 34
2.8. Maestro Proveedor 34
2.8.6. Truncar (operación destructiva) 36
2.9. Maestro Cliente 37
2.9.6. Truncar (operación destructiva) 39
2.10. Maestro Sucursal 39
2.10.6. Truncar (operación destructiva) 41
2.11. Configuración SKU por sucursal 42
2.11.6. Truncar (operación destructiva)

44
2.12. Descuento 45
2.12.6. Truncar (operación destructiva) 47
2.13. Jerarquía 47
2.13.6. Truncar (operación destructiva) 49
2.14. Conversión de unidades 49
2.14.6. Truncar (operación destructiva) 51
3. Métodos de consulta (solo lectura) 52
3.1. Resultado Abastecimiento 52
3.2. Indicadores (KPIs): quiebre, sobrestock y GMROI 54
3.2.1. Resumen 55
3.2.2. Quiebre 56
3.2.3. Sobrestock 58
3.2.4. GMROI 60
3.2.5. Códigos de respuesta 62
4. Cambios respecto a la versión 1.05 63
5. Cambios respecto a la versión 1.06 65
6. Cambios respecto a la versión 1.07 66
7. Cambios respecto a la versión 1.08 67
8. Cambios respecto a la versión 1.09 68
9. Cambios respecto a la versión 1.10 70
10. Cambios respecto a la versión 1.11 71

1. Descripción API

Supply Syncro provee a sus clientes una API **autodocumentada, segura y robusta**, que permite obtener, registrar y eliminar la información operativa en línea (productos, sucursales, demanda, inventarios, lead times, configuraciones) y consultar los indicadores que calcula Supply Syncro (quiebre, sobrestock y GMROI).

La dirección base de la API se encuentra en:

```
https://api.<CLIENTE>.supply-syncro.cl/
```

Donde `<CLIENTE>` corresponde al subdominio asignado al cliente (por ejemplo `api.miempresa.supply-syncro.cl`).

1.1. Trabajar con Swagger

Swagger nos ayuda a realizar consultas a la API **sin necesidad de programar ninguna línea de código**. Muestra todos los métodos disponibles con sus parámetros de entrada y tipos de datos, y al ejecutar un método genera una salida en formato JSON exportable.

Acceda a Swagger UI en la raíz de la URL de su cliente:

```
https://api.<CLIENTE>.supply-syncro.cl/
```

1.1.1. Autenticación

Para utilizar la API se requieren credenciales otorgadas por GPI Software (usuario y contraseña). Estas credenciales permiten obtener un **API-KEY** (token) para consultar los métodos disponibles.

Ejemplo de llamada usando curl

```
curl -X 'POST' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/api_login/' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d '{
        "username": "usuario",
        "password": "contraseña"
      }'
```

Salida exitosa `200`

```
{
  "mensaje": "Credenciales válidas.",
  "user_id": 3,
  "username": "usuario",
  "token": "9944b09199c62bcf9418ad846dd0e4bbdfc6ee4b"
}
```

Donde el campo `token` es el API-KEY que se debe usar en todas las llamadas posteriores.

> **IMPORTANTE** Cada llamada exitosa al método `api_login` **regenera el token** del usuario. Cualquier token previamente emitido queda inválido al iniciar sesión nuevamente.

Errores en el login

Caso	HTTP	Body
Credenciales inválidas	`401`	`{"mensaje": "Credenciales inválidas."}`
Faltan campos en el body	`400`	`{"mensaje": "Errores de validación.", "errores": {"username": ["This field is required."], "password": ["..."]}}`

Autorizar Swagger UI

En la esquina superior derecha de Swagger UI, presione **"Authorize"**. En la ventana emergente ingrese la palabra `token`, seguida por un espacio, y luego el API-KEY. Por ejemplo:

```
token 9944b09199c62bcf9418ad846dd0e4bbdfc6ee4b
```

1.1.2. Ejemplo de llamada

Una vez autorizado en Swagger, puede seleccionar cualquier método de la API y presionar **"Try it out"**, completar los parámetros de entrada y presionar **"Execute"** para ejecutar la petición.

Salida típica de un GET


```
{
  "links": {
    "next": null,
    "previous": null
  },
  "count": 1,
  "total_pages": 1,
  "results": [
    {
      "id_sucursal": "id_suc_1",
      "descripcion_sucursal": "nombre_suc1",
      "cd": 0,
      "pais_sucursal": "pais",
      "zona_sucursal": "zona",
      "canal_de_distribucion": "canal",
      "custom1": "", "custom2": "", "custom3": "", "custom4": "",
      "custom5": "", "custom6": "", "custom7": "", "custom8": "",
      "custom9": "", "custom10": ""
    }
  ]
}
```


OBSERVACIONES

- Todas las salidas de la API están en **formato JSON**.
- Todos los **GET** están paginados (1.000 registros por página por defecto, máximo 50.000 con `?page_size=N`).
- El bloque `links` incluye `next` (URL de la página siguiente) y `previous` (URL anterior). Si ambos son `null`, los resultados caben en una sola página.
- `count` es el total de registros que cumple la consulta.
- `results` contiene los registros de la página actual.

1.1.3. Códigos de salidas

Código	Descripción
200	OK. La petición se ejecutó correctamente. Devuelto en GETs, updates exitosos y DELETEs exitosos.
201	Creado. La petición se ejecutó correctamente y el registro fue **creado** (no actualizado).
400	Solicitud inválida. Los parámetros de entrada no cumplen el contrato (campos faltantes, tipos incorrectos, validaciones de rango).
401	No autorizado. El API-KEY proporcionado es incorrecto, está expirado, o no fue enviado.
413	Entidad demasiado grande. La API está limitada a un máximo de 120 días de información en filtros de fechas.
429	Demasiadas solicitudes. Los métodos GET están limitados a una consulta cada 30 segundos.
500	Error interno del servidor. La respuesta incluye un campo `detalle` con el tipo de error para diagnóstico.

1.2. Formato de respuestas

A partir de la versión 1.06, todas las respuestas siguen un formato uniforme. Esta sección describe cada caso.

1.2.1. Respuesta exitosa de POST

Los métodos **POST** de la API son idempotentes por su *llave natural*: si el registro ya existe, se actualiza; si no, se crea. La respuesta incluye contadores que permiten saber qué ocurrió.

Caso: registro único (objeto JSON)

```
{
  "mensaje": "Instancia creada correctamente.",
  "creado": true,
  "creados": 1,
  "actualizados": 0,
  "total_procesados": 1,
  "data": { /* registro guardado, serializado completo */ }
}
```

El HTTP status es `201` si fue creación, `200` si fue actualización. En el caso de update, el campo `mensaje` dice "*Instancia actualizada correctamente.*" y `creado` es `false`.

Caso: carga masiva (lista JSON)

```
{
  "mensaje": "Proceso masivo completado.",
  "total_procesados": 1000,
  "creados": 800,
  "actualizados": 200
}
```

HTTP `200` (o `201` si todos los items fueron creados). Para detalles sobre carga masiva, ver **sección 1.3**.

1.2.2. Respuesta exitosa de DELETE

```
{
  "mensaje": "Se eliminaron 3 registro(s).",
  "eliminados": 3,
  "eliminado": true
}
```

HTTP `200`. El campo `eliminados` es la cantidad real de filas borradas. `eliminado` (booleano) es `true` si se borró al menos un registro.

> **COMPATIBILIDAD** El campo `eliminado` (booleano) se mantiene por retrocompatibilidad. Para nuevas integraciones se recomienda usar `eliminados` (entero) que permite saber exactamente cuántas filas se afectaron.

1.2.3. Respuestas de error

Errores de validación `400`

Cuando los datos del body no cumplen el contrato (campo faltante, tipo incorrecto, valor fuera de rango):

```
{
  "mensaje": "Errores de validación.",
  "errores": {
    "id_material": ["This field is required."],
    "estado": ["Ensure this value is less than or equal to 1."]
  }
}
```

En modo **bulk**, `errores` es una *lista* donde cada posición corresponde a un item del array enviado (un dict vacío `{}` indica que ese item está OK):

```
{
  "mensaje": "Errores de validación.",
  "errores": [
    {},
    {"estado": ["Ensure this value is less than or equal to 1."]}
  ]
}
```

Parámetros faltantes `400`

Cuando un **DELETE** no incluye los parámetros obligatorios en el query string:

```
{
  "mensaje": "Faltan parámetros obligatorios.",
  "faltantes": ["id_material", "id_sucursal"]
}
```

Error interno `500`

```
{
  "mensaje": "Error interno al procesar la solicitud.",
  "detalle": "IntegrityError: <mensaje del driver>"
}
```

El stack trace completo queda registrado en el log del servidor para diagnóstico, pero **no se expone al cliente** por motivos de seguridad. El campo `detalle` entrega el tipo de error y mensaje para facilitar la identificación del problema.

1.3. Carga masiva (bulk) en POST

Todos los métodos POST de la API aceptan **tanto un objeto único como una lista de objetos** en el body. El comportamiento es:

- **Objeto JSON**: se procesa un solo registro. Respuesta single (con campo `data`).
- **Lista JSON**: se procesan los registros en lote. Respuesta resumida con contadores (`creados`, `actualizados`, `total_procesados`).

Atomicidad

Cada POST se ejecuta dentro de una **transacción atómica**. Si *cualquier* item del lote falla la validación o produce un error en la base de datos, se hace **rollback completo**: ningún registro queda parcialmente guardado.

Ejemplo de POST bulk

```
curl -X 'POST' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/maestro_producto/' \
  -H 'Authorization: token <API-KEY>' \
  -H 'Content-Type: application/json' \
  -d '[
    {"id_material":"P1","descripcion":"Producto
1","estado":1,"reposicion":1,"um_base":"UN"},
    {"id_material":"P2","descripcion":"Producto
2","estado":1,"reposicion":1,"um_base":"UN"},
    {"id_material":"P3","descripcion":"Producto
3","estado":1,"reposicion":1,"um_base":"UN"}
  ]'
```

```
{
  "mensaje": "Proceso masivo completado.",
  "total_procesados": 3,
  "creados": 2,
  "actualizados": 1
}
```

2. Métodos de carga de datos

A continuación se describe cada recurso con el que el cliente **carga y mantiene** su información (GET / POST / DELETE). Los métodos de solo lectura, que entregan información calculada por Supply Syncro, se describen en la **sección 3**. Todos los endpoints requieren autenticación vía header `Authorization: token <API-KEY>`, excepto `/api/v1/api_login/`.

2.1. Demanda Despacho

Descripción: Permite obtener, registrar y eliminar el registro diario de todas las ventas/despachos de productos, incluyendo el origen del movimiento (sucursal), el canal de venta y el cliente asociado.

GET POST DELETE

`https://api.<CLIENTE>.supply-syncro.cl/api/v1/demanda_despacho/`

POST acepta uno o más registros (objeto JSON único o array de objetos).

Parámetros de entrada (GET / DELETE)

- `fecha` — Fecha de transacción, formato AAAA-MM-DD *(opcional en GET, obligatorio en DELETE)*
- `id_material` — Código del producto *(opcional en GET, obligatorio en DELETE)*
- `id_sucursal` — Código de la sucursal del despacho/venta *(opcional en GET, obligatorio en DELETE)*
- `canal_venta` — Canal de la venta *(opcional)*
- `cliente` — Cliente asociado a la venta *(opcional)*

2.1.1. Ejemplo de llamada GET

```
curl -X 'GET' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/demanda_despacho/' \
  -H 'accept: application/json' \
  -H 'Authorization: token <API-KEY>'
```

2.1.2. Ejemplo de salida llamada GET


```
{
  "links": {
    "next": "https://api.<CLIENTE>.supply-syncro.cl/api/v1/demanda_despacho/?page=2",
    "previous": null
  },
  "count": 1942819,
  "total_pages": 1943,
  "results": [
    {
      "fecha": "2026-01-01",
      "id_material": "01210201145",
      "cantidad": 1,
      "id_despacho": "",
      "costo_unitario": 2877,
      "precio": 0,
      "venta_total": 0,
      "id_sucursal": "1020004",
      "canal_venta": "",
      "id_descuento": "",
      "porcentaje_descuento": 0,
      "custom1": null, "custom2": null, "custom3": null, "custom4": null,
      "custom5": null, "custom6": null, "custom7": null, "custom8": null,
      "custom9": null, "custom10": null,
      "cliente": "Nombre de cliente"
    }
  ]
}
```


2.1.3. Ejemplo de llamada POST

```
curl -X 'POST' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/demanda_despacho/' \
  -H 'Authorization: token <API-KEY>' \
  -H 'Content-Type: application/json' \
  -d '{
    "fecha": "2026-01-02",
    "id_material": "SKU_CODE1",
    "cantidad": 10,
    "id_despacho": "DESPACHO_CODE1",
    "costo_unitario": 500,
    "precio": 890,
    "venta_total": 8900,
    "id_sucursal": "IDSUC1",
    "canal_venta": "Retail",
    "id_descuento": "",
    "porcentaje_descuento": 0,
    "cliente": "Client name"
  }'
```

2.1.3.1. Descripción de campos

Campo	Tipo de dato	Descripción
`fecha` *	Fecha (AAAA-MM-DD)	Fecha de la transacción.
`id_material` *	Texto (100)	Código del producto.
`cantidad` *	Decimal	Cantidad de despachos (en la unidad de medida base).
`id_despacho`	Texto (100)	Código del formato de despacho (aplica solo para casos especiales con múltiples formatos). Si se omite, se almacena como string vacío.
`costo_unitario` *	Numérico	Costo unitario de la transacción.
`precio` *	Numérico	Precio de venta.
`venta_total`	Numérico	Factura total de la transacción.
`id_sucursal` *	Texto (100)	Código de la sucursal que realizó el despacho/venta.
`canal_venta`	Texto (100)	Canal donde se origina el movimiento. Tienda propia, retail, ecommerce, etc.
`id_descuento` *	Texto (100)	Variable explicativa para los modelos de IA. Tipo de descuento aplicado al producto. Si la transacción no tuvo descuento, puede ser `"sin descuento"`.
`porcentaje_descuento` *	Numérico	Valor del descuento ejecutado (en porcentaje).
`custom1, …, custom10`	Texto (100)	Características adicionales que pueda tener la empresa.
`cliente`	Texto (100)	Cliente asociado a la venta.

* Campos obligatorios.

> **CAMBIO V1.06** El campo `id_despacho`, anteriormente declarado opcional pero que producía error de integridad si se omitía, ahora se almacena automáticamente como string vacío cuando no es enviado. Ya no es necesario incluirlo si su empresa no usa formatos múltiples de despacho.

Llave natural para el upsert

El POST hace upsert por la combinación `(id_sucursal, id_material, fecha)`, más `canal_venta` y `cliente` si vienen en el cuerpo. Esto permite mantener registros separados por canal o cliente cuando se necesita.

2.1.4. Ejemplo de salida llamada POST 201


```
{
  "mensaje": "Instancia creada correctamente.",
  "creado": true,
  "creados": 1,
  "actualizados": 0,
  "total_procesados": 1,
  "data": {
    "fecha": "2026-01-02",
    "id_material": "SKU_CODE1",
    "cantidad": 10.0,
    "id_despacho": "DESPACHO_CODE1",
    "costo_unitario": 500.0,
    "precio": 890.0,
    "venta_total": 8900.0,
    "id_sucursal": "IDSUC1",
    "canal_venta": "Retail",
    "id_descuento": "",
    "porcentaje_descuento": 0.0,
    "cliente": "Client name"
  }
}
```


2.1.5. Ejemplo de llamada DELETE

```
curl -X 'DELETE' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/demanda_despacho/?
fecha=2026-01-02&id_material=SKU_CODE1&id_sucursal=IDSUC1' \
  -H 'Authorization: token <API-KEY>'
```

> **OBSERVACIÓN** En la eliminación, los campos `id_material`, `id_sucursal` y `fecha` son obligatorios. `canal_venta` y `cliente` son opcionales y, si se incluyen, restringen aún más el match.

2.1.6. Ejemplo de salida llamada DELETE 200


```
{
  "mensaje": "Se eliminaron 1 registro(s).",
  "eliminados": 1,
  "eliminado": true
}
```


2.2. Inventario Onhand

Descripción: Permite obtener, registrar y eliminar información sobre los inventarios físicos disponibles en todas las ubicaciones de la compañía por fecha (idealmente diario).

GET POST DELETE

`https://api.<CLIENTE>.supply-syncro.cl/api/v1/inventario_onhand/`

POST acepta uno o más registros (objeto JSON o array).

Parámetros de entrada (GET / DELETE)

- `fecha` — Fecha de la observación, formato AAAA-MM-DD
- `id_sucursal` — Código de la sucursal
- `id_material` — Código del producto

CAMBIO V1.06 El filtro `fecha` en el GET ahora se aplica correctamente cuando se envía junto con `id_sucursal` e `id_material`. En la versión 1.05 era ignorado en ese caso.

2.2.1. Ejemplo de llamada GET

```
curl -X 'GET' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/inventario_onhand/' \
  -H 'accept: application/json' \
  -H 'Authorization: token <API-KEY>'
```

2.2.2. Ejemplo de salida llamada GET


```
{
  "links": {
    "next": "https://api.<CLIENTE>.supply-syncro.cl/api/v1/inventario_onhand/?page=2",
    "previous": null
  },
  "count": 336146,
  "total_pages": 337,
  "results": [
    {
      "fecha": "2026-04-03",
      "id_material": "SKU-000123",
      "id_sucursal": "1001",
      "cantidad_oh": 0,
      "valor_inventario_oh": 0,
      "um_base": "UN",
      "cantidad_reservada": 0,
      "quiebre": 0,
      "quiebre_potencial": 0,
      "custom1": "", "custom2": "", "custom3": "", "custom4": "",
      "custom5": "", "custom6": "", "custom7": "", "custom8": "",
      "custom9": "", "custom10": ""
    }
  ]
}
```

2.2.3. Ejemplo de llamada POST

```
curl -X 'POST' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/inventario_onhand/' \
  -H 'Authorization: token <API-KEY>' \
  -H 'Content-Type: application/json' \
  -d '{
    "fecha": "2026-04-04",
    "id_material": "SKU-000123",
    "id_sucursal": "1001",
    "cantidad_oh": 0,
    "valor_inventario_oh": 0,
    "um_base": "UN",
    "cantidad_reservada": 0,
    "quiebre": 0,
    "quiebre_potencial": 0
  }'
```

2.2.3.1. Descripción de campos

Campo	Tipo	Descripción
`fecha` *	Fecha (AAAA-MM-DD)	Fecha de la observación.
`id_material` *	Texto (100)	Código del producto.
`id_sucursal` *	Texto (100)	Código de la sucursal.
`cantidad_oh` *	Decimal	Cantidad de inventario físico.
`valor_inventario_oh` *	Numérico	Valor en pesos del inventario disponible para venta (cantidad_oh).
`um_base` *	Texto (100)	Unidad de medida base de almacenamiento.
`cantidad_reservada` *	Decimal	Cantidad de stock físico no disponible para la venta.
`quiebre`	Numérico (0/1)	Variable dicotómica que indica si el producto está quebrado.
`quiebre_potencial`	Numérico (0/1)	Variable dicotómica que indica un quiebre potencial.
`custom1, …, custom10`	Texto (100)	Características adicionales.

* Campos obligatorios. **Llave natural del upsert:** `(id_material, id_sucursal, fecha)`.

2.2.4. Ejemplo de salida llamada POST 201

```
{
  "mensaje": "Instancia creada correctamente.",
  "creado": true,
  "creados": 1,
  "actualizados": 0,
  "total_procesados": 1,
  "data": { /* registro completo */ }
}
```

2.2.5. Ejemplo de llamada DELETE

```
curl -X 'DELETE' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/inventario_onhand/?
fecha=2026-04-04&id_sucursal=1001&id_material=SKU-000123' \
  -H 'Authorization: token <API-KEY>'
```

> **OBSERVACIÓN** En la eliminación todos los campos son obligatorios. Si falta alguno, la API responde **400** con la lista `faltantes` .

2.3. Inventario tránsito desde proveedores

Descripción: Permite obtener, registrar y eliminar información sobre el inventario en tránsito desde el proveedor hacia los centros de distribución de la compañía. Considera cantidades transportadas en unidad de medida base y montos.

GET **POST** **DELETE**
`https://api.<CLIENTE>.supply-syncro.cl/api/v1/inventario_transito_desde_proveedores/`

Parámetros de entrada (GET / DELETE)

- `fecha_pedido` — Fecha en que se realizó el pedido al proveedor (AAAA-MM-DD)
- `id_material` — Código del producto
- `id_proveedor` — Código del proveedor
- `id_sucursal_destino` — Código de la sucursal destino (en la mayoría de los casos el CD)

2.3.1. Ejemplo de llamada GET

```
curl -X 'GET' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/inventario_transito_desde_proveedores/' \
  -H 'accept: application/json' \
  -H 'Authorization: token <API-KEY>'
```

2.3.2. Ejemplo de salida llamada GET


```
{
  "links": { "next": null, "previous": null },
  "count": 1,
  "total_pages": 1,
  "results": [
    {
      "fecha_pedido": "2026-04-03",
      "numero_pedido": "1111",
      "id_material": "IDX2050",
      "id_proveedor": "22222",
      "id_sucursal_destino": "3333",
      "cantidad_pedido_oc": 5,
      "cantidad_entregada": 4,
      "cantidad_pendiente": 1,
      "fecha_est_llegada": "2026-05-03",
      "um_base": "UM",
      "valor_inventario_transito": 300,
      "custom1": "", "custom2": "", "custom3": "", "custom4": "",
      "custom5": "", "custom6": "", "custom7": "", "custom8": "",
      "custom9": "", "custom10": ""
    }
  ]
}
```

2.3.3. Ejemplo de llamada POST

```
curl -X 'POST' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/inventario_transito_desde_proveedores/' \
  -H 'Authorization: token <API-KEY>' \
  -H 'Content-Type: application/json' \
  -d '{
    "fecha_pedido": "2026-04-04",
    "numero_pedido": "1111",
    "id_material": "IDX2050",
    "id_proveedor": "22222",
    "id_sucursal_destino": "3333",
    "cantidad_pedido_oc": 5,
    "cantidad_entregada": 4,
    "cantidad_pendiente": 1,
    "fecha_est_llegada": "2026-05-03",
    "um_base": "UM",
    "valor_inventario_transito": 300
  }'
```

2.3.3.1. Descripción de campos

Campo	Tipo	Descripción
`fecha_pedido` *	Fecha (AAAA-MM-DD)	Fecha en que se realizó el pedido al proveedor.
`numero_pedido` *	Texto (100)	Código del pedido (en muchos casos puede ser la factura).
`id_material` *	Texto (100)	Código del producto.
`id_proveedor` *	Texto (100)	Código del proveedor.
`id_sucursal_destino` *	Texto (100)	Código de la sucursal donde se dirige el producto.
`cantidad_pedido_oc` *	Decimal	Cantidad de unidades solicitadas del producto.
`cantidad_entregada` *	Decimal	Cantidad de unidades entregadas.
`cantidad_pendiente` *	Decimal	Cantidad de unidades pendientes por entregar.
`um_base`	Texto (100)	Unidad de medida base de almacenamiento.
`fecha_est_llegada` *	Fecha (AAAA-MM-DD)	Fecha estimada de llegada al CD.
`valor_inventario_transito`	Numérico	Valor del inventario pendiente de entregar (en $).
`custom1, …, custom10`	Texto (100)	Características adicionales.

* Campos obligatorios. **Llave natural:** `(fecha_pedido, numero_pedido, id_material, id_proveedor, id_sucursal_destino, fecha_est_llegada)`. El número de pedido integra la llave

para que dos órdenes de compra distintas del mismo producto, proveedor y destino se guarden como filas separadas; la fecha estimada de llegada, para soportar entregas **parcializadas**.

2.3.4. Ejemplo de salida llamada POST `201`

```
{
  "mensaje": "Instancia creada correctamente.",
  "creado": true,
  "creados": 1,
  "actualizados": 0,
  "total_procesados": 1,
  "data": { /* registro completo */ }
}
```

2.3.5. Ejemplo de llamada DELETE

```
curl -X 'DELETE' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/inventario_transito_desde_proveedores/?
fecha_pedido=2026-04-03&id_material=11&id_proveedor=11111&id_sucursal_destino=22222' \
  -H 'Authorization: token <API-KEY>'
```

> **OBSERVACIÓN** Los cuatro campos del query string son obligatorios. Se pueden agregar dos parámetros **opcionales** para acotar el borrado: `numero_pedido` (si se omite, se borran los registros de todos los pedidos de esa llave) y `fecha_est_llegada` (si se omite, se borran todas las entregas parciales).

2.3.6. Truncar (operación destructiva)

El recurso **inventario tránsito desde proveedores** tiene un endpoint especial para vaciar la tabla completa, útil antes de realizar la carga diaria masiva.

DELETE
`https://api.<CLIENTE>.supply-syncro.cl/api/v1/inventario_transito_desde_proveedores/truncar/?confirm=true`

> **CAMBIO V1.06** Por seguridad, este endpoint ahora **requiere el parámetro** `?confirm=true` en el query string. Sin él, la API responde **`400`** y no ejecuta la operación. Solo el valor exacto `true` es aceptado.

> **NOVEDAD V1.08** El endpoint `truncar/` ahora está disponible para **todos los recursos CRUD**, no solo para inventario tránsito desde proveedores. El patrón de uso es idéntico en todos: `DELETE /api/v1/<recurso>/truncar/?confirm=true`. Cada recurso lo documenta en su propia subsección "*Truncar (operación destructiva)*" (ver índice).

Llamada

```
curl -X 'DELETE' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/inventario_transito_desde_proveedores/
truncar/?confirm=true' \
  -H 'Authorization: token <API-KEY>'
```

Respuesta exitosa 200

```
{
  "mensaje": "Tabla truncada. Se eliminaron 1796 registro(s).",
  "eliminados": 1796
}
```

Respuesta sin confirmación 400

```
{
  "mensaje": "Operación destructiva. Debe enviar ?confirm=true para ejecutar el
truncate.",
  "eliminados": 0
}
```

2.4. Inventario tránsito entre sucursales

Descripción: Permite obtener, registrar y eliminar información sobre inventario en movimiento entre sucursales o centros de distribución, con cantidades y montos.

GET POST DELETE
`https://api.<CLIENTE>.supply-syncro.cl/api/v1/inventario_transito_entre_sucursales/`

Parámetros de entrada (GET / DELETE)

- `fecha_pedido` — Fecha en que se realizó el pedido (AAAA-MM-DD)
- `id_material` — Código del producto
- `id_sucursal_origen` — Código de la sucursal origen del producto
- `id_sucursal_destino` — Código de la sucursal de destino (abastecimiento)

> **CAMBIO V1.06** El filtro `id_sucursal_origen` en el GET ahora funciona correctamente. En la versión 1.05 producía un error 500 al ser enviado.

2.4.1. Ejemplo de llamada GET

```
curl -X 'GET' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/inventario_transito_entre_sucursales/' \
  -H 'accept: application/json' \
  -H 'Authorization: token <API-KEY>'
```

2.4.2. Ejemplo de salida llamada GET


```
{
  "links": { "next": null, "previous": null },
  "count": 1,
  "total_pages": 1,
  "results": [
    {
      "fecha_pedido": "2026-04-18",
      "numero_pedido": "1888811",
      "id_material": "AX23456",
      "id_sucursal_origen": "22",
      "id_sucursal_destino": "33",
      "cantidad_pedido_interno": 0,
      "cantidad_entregada": 0,
      "cantidad_pendiente": 0,
      "fecha_est_llegada": "2026-04-20",
      "um_base": "UM",
      "valor_inventario_transito": 0,
      "custom1": null, "custom2": null, "custom3": null, "custom4": null,
      "custom5": null, "custom6": null, "custom7": null, "custom8": null,
      "custom9": null, "custom10": null
    }
  ]
}
```

2.4.3. Ejemplo de llamada POST

```
curl -X 'POST' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/inventario_transito_entre_sucursales/' \
  -H 'Authorization: token <API-KEY>' \
  -H 'Content-Type: application/json' \
  -d '{
    "fecha_pedido": "2026-04-18",
    "numero_pedido": "11",
    "id_material": "1111",
    "id_sucursal_origen": "22",
    "id_sucursal_destino": "33",
    "cantidad_pedido_interno": 0,
    "cantidad_entregada": 0,
    "cantidad_pendiente": 0,
    "fecha_est_llegada": "2026-04-20",
    "um_base": "UM",
    "valor_inventario_transito": 0
  }'
```

2.4.3.1. Descripción de campos

Campo	Tipo	Descripción
`fecha_pedido` *	Fecha (AAAA-MM-DD)	Fecha en que se realizó el pedido.
`numero_pedido` *	Texto (100)	Código del pedido (interno si aplica).
`id_material` *	Texto (100)	Código del producto.
`id_sucursal_origen` *	Texto (100)	Sucursal que envía el producto.
`id_sucursal_destino` *	Texto (100)	Sucursal que recibe el producto.
`cantidad_pedido_interno`	Decimal	Cantidad solicitada del producto.
`cantidad_entregada`	Decimal	Cantidad entregada.
`cantidad_pendiente` *	Decimal	Cantidad pendiente de entregar.
`um_base` *	Texto (100)	Unidad de medida base.
`fecha_est_llegada`	Fecha (AAAA-MM-DD)	Fecha estimada de llegada al destino. **Opcional desde v1.10**: si se omite, el registro queda sin fecha estimada. Forma parte de la llave natural, por lo que los envíos con fecha y sin fecha se guardan como filas distintas.
`valor_inventario_transito` *	Numérico	Valor del inventario pendiente (en $).
`custom1, …, custom10`	Texto (100)	Características adicionales.

* Campos obligatorios. **Llave natural:** `(fecha_pedido, id_material, id_sucursal_origen, id_sucursal_destino, fecha_est_llegada)`. La fecha estimada de llegada integra la llave para soportar entregas **parcializadas** (una misma orden que llega en varios envíos); si se omite, el registro se identifica por la llave con la fecha vacía.

2.4.4. Ejemplo de salida llamada POST 201

```
{
  "mensaje": "Instancia creada correctamente.",
  "creado": true,
  "creados": 1, "actualizados": 0, "total_procesados": 1,
  "data": { /* registro completo */ }
}
```

2.4.5. Ejemplo de llamada DELETE

```
curl -X 'DELETE' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/inventario_transito_entre_sucursales/?
fecha_pedido=2026-05-05&id_material=AX34498&id_sucursal_origen=222&id_sucursal_destino=333
' \
  -H 'Authorization: token <API-KEY>'
```

2.4.6. Truncar (operación destructiva)

Vacía por completo la tabla de **inventario tránsito entre sucursales**: borra **todos** los registros del recurso. Disponible desde v1.08.

DELETE
`https://api.<CLIENTE>.supply-syncro.cl/api/v1/inventario_transito_entre_sucursales/truncar/?confirm=true`

CONFIRMACIÓN OBLIGATORIA Requiere el parámetro exacto `?confirm=true`. Sin él, la API responde **400** y no toca la tabla.

```
curl -X 'DELETE' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/inventario_transito_entre_sucursales/
truncar/?confirm=true' \
  -H 'Authorization: token <API-KEY>'
```

Respuesta exitosa **200**: `{ "mensaje": "Tabla truncada. Se eliminaron N registro(s).", "eliminados": N }`.

2.5. Lead Times Internos

Descripción: Permite obtener, registrar y eliminar la configuración del tiempo de entrega entre sucursales — dato relevante para la planificación del abastecimiento.

GET **POST** **DELETE**
`https://api.<CLIENTE>.supply-syncro.cl/api/v1/lead_times_internos/`

Parámetros de entrada (GET / DELETE)

- `id_sucursal_origen` — Sucursal origen del producto
- `id_sucursal_destino` — Sucursal destino (abastecimiento)

2.5.1. Ejemplo de llamada GET

```
curl -X 'GET' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/lead_times_internos/' \
  -H 'accept: application/json' \
  -H 'Authorization: token <API-KEY>'
```

2.5.2. Ejemplo de salida llamada GET

```
{
  "links": { "next": null, "previous": null },
  "count": 227,
  "total_pages": 1,
  "results": [
    {
      "id_sucursal_origen": "1010022",
      "id_sucursal_destino": "1020033",
      "lead_time_despacho": 7,
      "desviacion_lead_time_despacho": 2,
      "custom1": "", "custom2": "", "custom3": "", "custom4": "",
      "custom5": "", "custom6": "", "custom7": "", "custom8": "",
      "custom9": "", "custom10": ""
    }
  ]
}
```

2.5.3. Ejemplo de llamada POST

```
curl -X 'POST' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/lead_times_internos/' \
  -H 'Authorization: token <API-KEY>' \
  -H 'Content-Type: application/json' \
  -d '{
    "id_sucursal_origen": "1010022",
    "id_sucursal_destino": "1020033",
    "lead_time_despacho": 7,
    "desviacion_lead_time_despacho": 2
  }'
```

2.5.3.1. Descripción de campos

Campo	Tipo	Descripción
`id_sucursal_origen` *	Texto (100)	Sucursal origen del producto.
`id_sucursal_destino` *	Texto (100)	Sucursal de destino (abastecimiento).
`lead_time_despacho` *	Numérico	Tiempo de despacho entre sucursales (en días).
`desviacion_lead_time_despacho`	Numérico	Desviación estándar del lead time de despacho.
`custom1, …, custom10`	Texto (100)	Características adicionales.

* Campos obligatorios. **Llave natural:** `(id_sucursal_origen, id_sucursal_destino)`.

2.5.4. Ejemplo de salida llamada POST 201

```
{
  "mensaje": "Instancia creada correctamente.",
  "creado": true, "creados": 1, "actualizados": 0, "total_procesados": 1,
  "data": { /* registro completo */ }
}
```

2.5.5. Ejemplo de llamada DELETE

```
curl -X 'DELETE' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/lead_times_internos/?
id_sucursal_origen=1010022&id_sucursal_destino=1020033' \
  -H 'Authorization: token <API-KEY>'
```

2.5.6. Truncar (operación destructiva)

Vacía por completo la tabla de **lead times internos**: borra **todos** los registros del recurso. Disponible desde v1.08.

DELETE
`https://api.<CLIENTE>.supply-syncro.cl/api/v1/lead_times_internos/truncar/?confirm=true`

CONFIRMACIÓN OBLIGATORIA Requiere el parámetro exacto `?confirm=true`. Sin él, la API responde **400** y no toca la tabla.

```
curl -X 'DELETE' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/lead_times_internos/truncar/?
confirm=true' \
  -H 'Authorization: token <API-KEY>'
```

Respuesta exitosa `200`: `{ "mensaje": "Tabla truncada. Se eliminaron N registro(s).", "eliminados": N }`.

2.6. Relación productos-proveedores

Descripción: Permite obtener, registrar y eliminar la relación detallada entre productos y proveedores, incluyendo lead times y cantidades máximas/mínimas de las órdenes de compra.

`GET` `POST` `DELETE`
`https://api.<CLIENTE>.supply-syncro.cl/api/v1/rel_proveedor_sku/`

Parámetros de entrada (GET / DELETE)

- `id_proveedor` — Código del proveedor
- `id_material` — Código del producto

2.6.1. Ejemplo de llamada GET

```
curl -X 'GET' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/rel_proveedor_sku/' \
  -H 'accept: application/json' \
  -H 'Authorization: token <API-KEY>'
```

2.6.2. Ejemplo de salida llamada GET


```
{
  "links": { "next": null, "previous": null },
  "count": 1, "total_pages": 1,
  "results": [
    {
      "id_proveedor": "1XXXX",
      "id_material": "3C0A0",
      "maximo_compra": 10,
      "minimo_compra": 5,
      "lead_time_oc": 7,
      "desviacion_lead_time": 1,
      "prioridad": 1,
      "custom1": "", "custom2": "", "custom3": "", "custom4": "",
      "custom5": "", "custom6": "", "custom7": "", "custom8": "",
      "custom9": "", "custom10": "",
      "id_sucursal_destino": null
    }
  ]
}
```


2.6.3. Ejemplo de llamada POST

```
curl -X 'POST' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/rel_proveedor_sku/' \
  -H 'Authorization: token <API-KEY>' \
  -H 'Content-Type: application/json' \
  -d '{
    "id_proveedor": "AX2738",
    "id_material": "3C0A0",
    "maximo_compra": 10,
    "minimo_compra": 5,
    "lead_time_oc": 7,
    "desviacion_lead_time": 1,
    "prioridad": 1
  }'
```

2.6.3.1. Descripción de campos

Campo	Tipo	Descripción
`id_proveedor` *	Texto (100)	Código del proveedor.
`id_material` *	Texto (100)	Código del producto.
`maximo_compra`	Numérico	Máximo de compra por la combinación proveedor-producto.
`minimo_compra`	Numérico	Mínimo de compra (combinación producto-proveedor).
`lead_time_oc` *	Numérico	Lead time del proveedor en días.
`desviacion_lead_time`	Numérico	Desviación estándar del lead time de compra.
`prioridad`	Numérico (0/1)	`1`: combinación principal; `0`: segunda o tercera opción.
`id_sucursal_destino`	Texto (100)	Código de la sucursal de destino.
`custom1, …, custom10`	Texto (100)	Características adicionales.

* Campos obligatorios. **Llave natural:** `(id_proveedor, id_material)`.

2.6.4. Ejemplo de salida llamada POST 201

```
{
  "mensaje": "Instancia creada correctamente.",
  "creado": true, "creados": 1, "actualizados": 0, "total_procesados": 1,
  "data": { /* registro completo */ }
}
```

2.6.5. Ejemplo de llamada DELETE

```
curl -X 'DELETE' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/rel_proveedor_sku/?
id_proveedor=AX30000&id_material=1111' \
  -H 'Authorization: token <API-KEY>'
```

2.6.6. Truncar (operación destructiva)

Vacía por completo la tabla de **relación productos-proveedores**: borra **todos** los registros del recurso. Disponible desde v1.08.

```
DELETE
https://api.<CLIENTE>.supply-syncro.cl/api/v1/rel_proveedor_sku/truncar/?
confirm=true
```

CONFIRMACIÓN OBLIGATORIA Requiere el parámetro exacto `?confirm=true`. Sin él, la API responde **400** y no toca la tabla.

```
curl -X 'DELETE' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/rel_proveedor_sku/truncar/?confirm=true'
\
  -H 'Authorization: token <API-KEY>'
```

Respuesta exitosa **200**: `{ "mensaje": "Tabla truncada. Se eliminaron N registro(s).", "eliminados": N }`.

2.7. Maestro Productos

Descripción: Permite obtener, registrar y eliminar todos los atributos de los productos.

GET **POST** **DELETE**
`https://api.<CLIENTE>.supply-syncro.cl/api/v1/maestro_producto/`

Parámetros de entrada (GET / DELETE)

- `id_material` — Código del producto

2.7.1. Ejemplo de llamada GET

```
curl -X 'GET' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/maestro_producto/' \
  -H 'accept: application/json' \
  -H 'Authorization: token <API-KEY>'
```

2.7.2. Ejemplo de salida llamada GET


```
{
  "links": {
    "next": "https://api.<CLIENTE>.supply-syncro.cl/api/v1/maestro_producto/?page=2",
    "previous": null
  },
  "count": 63049,
  "total_pages": 64,
  "results": [
    {
      "id_material": "SKU-000456",
      "descripcion": "PRODUCTO DE PRUEBA /50,RL50 R135QR",
      "jerarquia_1": "A", "jerarquia_2": "B", "jerarquia_3": "C", "jerarquia_4": "D",
      "jerarquia_5": "", "jerarquia_6": "", "jerarquia_7": "",
      "volumen": 0, "peso_g": 0,
      "articulo": "", "talla": "", "color": "", "marca": "MARCA 1",
      "estado": 1, "caducidad": 0, "reposicion": 0, "um_base": "UN",
      "empaque": null, "costo": null,
      "custom1": "FUERA DE MIX", "custom2": "AUTOS", "custom3": "ACCESORIOS",
      "custom4": "0", "custom5": "", "custom6": "", "custom7": "",
      "custom8": "", "custom9": "", "custom10": ""
    }
  ]
}
```


2.7.3. Ejemplo de llamada POST

```
curl -X 'POST' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/maestro_producto/' \
  -H 'Authorization: token <API-KEY>' \
  -H 'Content-Type: application/json' \
  -d '{
    "id_material": "SKU-000456",
    "descripcion": "TAPA MAZA SRAM KIT AXLCAP SET XD RM60/50,RL50 R135QR",
    "marca": "MONGOOSE",
    "estado": 1,
    "caducidad": 0,
    "reposicion": 0,
    "um_base": "UN"
  }'
```

2.7.3.1. Descripción de campos

Campo	Tipo	Descripción
`id_material` *	Texto (100)	Código del producto.
`descripcion` *	Texto (300)	Descripción del producto.
`jerarquia_1`	Texto (150)	Niveles de jerarquía dentro de la estructura organizativa de la compañía. Útiles para segmentar y agrupar (familia, categoría, subcategoría, etc.). **Opcional desde v1.10**.
`jerarquia_2`	Texto (150)	Idem jerarquía_1. **Opcional desde v1.10**.
`jerarquia_3`	Texto (150)	Idem jerarquía_1. **Opcional desde v1.10**.
`jerarquia_4`	Texto (150)	Idem jerarquía_1. **Opcional desde v1.10**.
`jerarquia_5, _6, _7`	Texto (150)	Idem jerarquía_1. Opcionales.
`volumen`	Numérico	Medida volumétrica del producto.
`peso_g`	Numérico	Peso del producto en gramos.
`articulo`	Texto (150)	Jerarquía del producto para agrupar/segmentar.
`talla`	Texto (150)	Talla del producto si aplica.
`color`	Texto (150)	Color del producto si aplica.
`marca`	Texto (150)	Marca del producto.
`estado` *	Numérico (0/1)	Indica si el producto está activo (1) para pronóstico y abastecimiento.
`caducidad`	Numérico (0/1)	Indica si el producto tiene fecha de caducidad.
`reposicion` *	Numérico (0/1)	Indica si el producto requiere reposición automática.
`um_base` *	Texto (150)	Unidad de medida base de almacenamiento.
`empaque`	Numérico	Cantidad de unidades en un paquete.
`costo`	Numérico	Costo unitario del producto.
`custom1, …, custom10`	Texto (150)	Características adicionales.

* Campos obligatorios. **Llave natural:** `(id_material)`.

2.7.4. Ejemplo de salida llamada POST 201

```
{
  "mensaje": "Instancia creada correctamente.",
  "creado": true, "creados": 1, "actualizados": 0, "total_procesados": 1,
  "data": { /* registro completo */ }
}
```

2.7.5. Ejemplo de llamada DELETE

```
curl -X 'DELETE' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/maestro_producto/?id_material=SKU-000456'
\
  -H 'Authorization: token <API-KEY>'
```

2.7.6. Truncar (operación destructiva)

Vacía por completo la tabla de **maestro productos**: borra **todos** los registros del recurso. Disponible desde v1.08.

DELETE
`https://api.<CLIENTE>.supply-syncro.cl/api/v1/maestro_producto/truncar/?confirm=true`

CONFIRMACIÓN OBLIGATORIA Requiere el parámetro exacto `?confirm=true`. Sin él, la API responde **400** y no toca la tabla.

```
curl -X 'DELETE' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/maestro_producto/truncar/?confirm=true' \
  -H 'Authorization: token <API-KEY>'
```

Respuesta exitosa 200: `{ "mensaje": "Tabla truncada. Se eliminaron N registro(s).", "eliminados": N }`.

2.8. Maestro Proveedor

Descripción: Permite obtener, registrar y eliminar todos los atributos de los proveedores.

GET **POST** **DELETE**
`https://api.<CLIENTE>.supply-syncro.cl/api/v1/maestro_proveedor/`

Parámetros de entrada (GET / DELETE)

- `id_proveedor` — Código del proveedor

2.8.1. Ejemplo de llamada GET

```
curl -X 'GET' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/maestro_proveedor/' \
  -H 'accept: application/json' \
  -H 'Authorization: token <API-KEY>'
```

2.8.2. Ejemplo de salida llamada GET

```
{
  "links": { "next": null, "previous": null },
  "count": 8, "total_pages": 1,
  "results": [
    {
      "id_proveedor": "PROVEEDOR_TEST_1",
      "nombre_proveedor": "nombre_test1",
      "rut": "1234567-8",
      "direccion": "direccion 1",
      "origen": "origen 1",
      "custom1": "", "custom2": "", "custom3": "", "custom4": "",
      "custom5": "", "custom6": "", "custom7": "", "custom8": "",
      "custom9": "", "custom10": ""
    }
  ]
}
```

2.8.3. Ejemplo de llamada POST

```
curl -X 'POST' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/maestro_proveedor/' \
  -H 'Authorization: token <API-KEY>' \
  -H 'Content-Type: application/json' \
  -d '{
    "id_proveedor": "PROVEEDOR_TEST_1",
    "nombre_proveedor": "nombre_test1",
    "rut": "1234567-8",
    "direccion": "direccion 1",
    "origen": "origen 1"
  }'
```

2.8.3.1. Descripción de campos

Campo	Tipo	Descripción
`id_proveedor` *	Texto (100)	Código del proveedor.
`nombre_proveedor` *	Texto (300)	Descripción del proveedor.
`rut`	Texto (150)	RUT o número de identificación.
`direccion`	Texto (300)	Dirección casa matriz del proveedor.
`origen`	Texto (150)	País de origen del proveedor.
`custom1, …, custom10`	Texto (150)	Características adicionales.

* Campos obligatorios. **Llave natural:** `(id_proveedor)`.

2.8.4. Ejemplo de salida llamada POST 201

```
{
  "mensaje": "Instancia creada correctamente.",
  "creado": true, "creados": 1, "actualizados": 0, "total_procesados": 1,
  "data": { /* registro completo */ }
}
```

2.8.5. Ejemplo de llamada DELETE

```
curl -X 'DELETE' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/maestro_proveedor/?
id_proveedor=proveedor_15' \
  -H 'Authorization: token <API-KEY>'
```

2.8.6. Truncar (operación destructiva)

Vacía por completo la tabla de **maestro proveedor**: borra **todos** los registros del recurso. Disponible desde v1.08.

DELETE
`https://api.<CLIENTE>.supply-syncro.cl/api/v1/maestro_proveedor/truncar/?confirm=true`

CONFIRMACIÓN OBLIGATORIA Requiere el parámetro exacto `?confirm=true`. Sin él, la API responde **400** y no toca la tabla.

```
curl -X 'DELETE' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/maestro_proveedor/truncar/?confirm=true'
\
  -H 'Authorization: token <API-KEY>'
```

Respuesta exitosa `200`: `{ "mensaje": "Tabla truncada. Se eliminaron N registro(s).", "eliminados": N }`.

2.9. Maestro Cliente

Descripción: Permite obtener, registrar y eliminar todos los atributos de los clientes.

GET **POST** **DELETE**
`https://api.<CLIENTE>.supply-syncro.cl/api/v1/maestro_cliente/`

Parámetros de entrada (GET / DELETE)

- `id_cliente` — Código del cliente

2.9.1. Ejemplo de llamada GET

```
curl -X 'GET' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/maestro_cliente/' \
  -H 'accept: application/json' \
  -H 'Authorization: token <API-KEY>'
```

2.9.2. Ejemplo de salida llamada GET


```
{
  "links": { "next": null, "previous": null },
  "count": 8, "total_pages": 1,
  "results": [
    {
      "id_cliente": "client_id_1",
      "nombre_cliente": "Cliente #1",
      "rut": "11222333-4",
      "direccion": "Address #1",
      "origen": "Orig",
      "custom1": "", "custom2": "", "custom3": "", "custom4": "",
      "custom5": "", "custom6": "", "custom7": "", "custom8": "",
      "custom9": "", "custom10": ""
    }
  ]
}
```

2.9.3. Ejemplo de llamada POST

```
curl -X 'POST' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/maestro_cliente/' \
  -H 'Authorization: token <API-KEY>' \
  -H 'Content-Type: application/json' \
  -d '{
    "id_cliente": "11222333",
    "nombre_cliente": "Cliente #1",
    "rut": "11222333-4",
    "direccion": "Address #1",
    "origen": "Orig"
  }'
```

2.9.3.1. Descripción de campos

Campo	Tipo	Descripción
`id_cliente` *	Texto (100)	Código del cliente.
`nombre_cliente` *	Texto (100)	Descripción del cliente.
`rut`	Texto (100)	RUT o identificación.
`direccion`	Texto (100)	Dirección casa matriz.
`origen`	Texto (100)	País de origen.
`custom1, …, custom10`	Texto (150)	Características adicionales.

* Campos obligatorios. **Llave natural:** `(id_cliente)`.

2.9.4. Ejemplo de salida llamada POST 201

```
{
  "mensaje": "Instancia creada correctamente.",
  "creado": true, "creados": 1, "actualizados": 0, "total_procesados": 1,
  "data": { /* registro completo */ }
}
```

2.9.5. Ejemplo de llamada DELETE

```
curl -X 'DELETE' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/maestro_cliente/?id_cliente=12345655' \
  -H 'Authorization: token <API-KEY>'
```

2.9.6. Truncar (operación destructiva)

Vacía por completo la tabla de **maestro cliente**: borra **todos** los registros del recurso. Disponible desde v1.08.

DELETE
`https://api.<CLIENTE>.supply-syncro.cl/api/v1/maestro_cliente/truncar/?confirm=true`

> **CONFIRMACIÓN OBLIGATORIA** Requiere el parámetro exacto `?confirm=true`. Sin él, la API responde **400** y no toca la tabla.

```
curl -X 'DELETE' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/maestro_cliente/truncar/?confirm=true' \
  -H 'Authorization: token <API-KEY>'
```

Respuesta exitosa **200**: `{ "mensaje": "Tabla truncada. Se eliminaron N registro(s).", "eliminados": N }`.

2.10. Maestro Sucursal

Descripción: Permite obtener, registrar y eliminar todos los atributos de las sucursales.

GET **POST** **DELETE**
`https://api.<CLIENTE>.supply-syncro.cl/api/v1/maestro_sucursal/`

Parámetros de entrada (GET / DELETE)

- `id_sucursal` — Código de la sucursal

2.10.1. Ejemplo de llamada GET

```
curl -X 'GET' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/maestro_sucursal/' \
  -H 'accept: application/json' \
  -H 'Authorization: token <API-KEY>'
```

2.10.2. Ejemplo de salida llamada GET


```
{
  "links": { "next": null, "previous": null },
  "count": 1, "total_pages": 1,
  "results": [
    {
      "id_sucursal": "id_suc_1",
      "descripcion_sucursal": "nombre_suc1",
      "cd": 0,
      "pais_sucursal": "pais",
      "zona_sucursal": "zona",
      "canal_de_distribucion": "canal",
      "custom1": "11", "custom2": "", "custom3": "33", "custom4": "",
      "custom5": "55", "custom6": "", "custom7": "", "custom8": "",
      "custom9": "", "custom10": "",
      "estado": true,
      "consolida": null,
      "consolida_stock_compra": null
    }
  ]
}
```


2.10.3. Ejemplo de llamada POST

```
curl -X 'POST' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/maestro_sucursal/' \
  -H 'Authorization: token <API-KEY>' \
  -H 'Content-Type: application/json' \
  -d '{
    "id_sucursal": "id_suc_1",
    "descripcion_sucursal": "nombre_suc1",
    "cd": 2,
    "pais_sucursal": "string",
    "zona_sucursal": "string",
    "canal_de_distribucion": "string",
    "estado": true
  }'
```

2.10.3.1. Descripción de campos

Campo	Tipo	Descripción
`id_sucursal` *	Texto (150)	Código de la sucursal.
`descripcion_sucursal` *	Texto (150)	Descripción de la sucursal.
`cd` *	Numérico (0/1/2)	Flag identificador de centro de distribución.
`pais_sucursal`	Texto (150)	País de la sucursal.
`zona_sucursal`	Texto (150)	Zona de la sucursal.
`canal_de_distribucion`	Texto (150)	Clasificación del canal.
`estado`	Booleano	Estado activo/inactivo de la sucursal.
`consolida`	Texto (150)	Consolidación del stock en bodegas para despacho.
`consolida_stock_compra`	Texto (150)	Consolidación del stock en bodegas para decisión de compra.
`custom1, …, custom10`	Texto (150)	Características adicionales.

* Campos obligatorios. **Llave natural:** `(id_sucursal)`.

2.10.4. Ejemplo de salida llamada POST 201

```
{
  "mensaje": "Instancia creada correctamente.",
  "creado": true, "creados": 1, "actualizados": 0, "total_procesados": 1,
  "data": { /* registro completo */ }
}
```

2.10.5. Ejemplo de llamada DELETE

```
curl -X 'DELETE' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/maestro_sucursal/?id_sucursal=id_suc_1' \
  -H 'Authorization: token <API-KEY>'
```

2.10.6. Truncar (operación destructiva)

Vacía por completo la tabla de **maestro sucursal**: borra **todos** los registros del recurso. Disponible desde v1.08.

DELETE

```
https://api.<CLIENTE>.supply-syncro.cl/api/v1/maestro_sucursal/truncar/?
confirm=true
```

CONFIRMACIÓN OBLIGATORIA Requiere el parámetro exacto `?confirm=true`. Sin él, la API responde **400** y no toca la tabla.

```
curl -X 'DELETE' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/maestro_sucursal/truncar/?confirm=true' \
  -H 'Authorization: token <API-KEY>'
```

Respuesta exitosa **200**: `{ "mensaje": "Tabla truncada. Se eliminaron N registro(s).", "eliminados": N }`.

2.11. Configuración SKU por sucursal

Descripción: Permite obtener, registrar y eliminar la configuración de un producto en una sucursal específica.

GET **POST** **DELETE**
`https://api.<CLIENTE>.supply-syncro.cl/api/v1/config_sku_sucursal/`

Parámetros de entrada (GET / DELETE)

- `id_material` — Código del producto
- `id_sucursal` — Código de la sucursal

2.11.1. Ejemplo de llamada GET

```
curl -X 'GET' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/config_sku_sucursal/' \
  -H 'accept: application/json' \
  -H 'Authorization: token <API-KEY>'
```

2.11.2. Ejemplo de salida llamada GET

```
{
  "links": { "next": null, "previous": null },
  "count": 8, "total_pages": 1,
  "results": [
    {
      "id_material": "sku_id_1",
      "id_sucursal": "suc_id_1",
      "min_inv_requerido": 0,
      "max_repo": 0,
      "cd_origen": "",
      "consolida_compra": null,
      "max_stock": -1
    }
  ]
}
```

2.11.3. Ejemplo de llamada POST

```
curl -X 'POST' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/config_sku_sucursal/' \
  -H 'Authorization: token <API-KEY>' \
  -H 'Content-Type: application/json' \
  -d '{
    "id_material": "sku_id_2",
    "id_sucursal": "suc_id_2",
    "min_inv_requerido": 0,
    "max_repo": 0,
    "cd_origen": "",
    "max_stock": 500
  }'
```

2.11.3.1. Descripción de campos

Campo	Tipo	Descripción
`id_material` *	Texto (150)	Código del producto.
`id_sucursal` *	Texto (150)	Código de la sucursal.
`max_repo`	Numérico	Máxima cantidad de un SKU enviable a una sucursal específica.
`min_inv_requerido`	Numérico	Mínima cantidad requerida del SKU en la sucursal (productos críticos).
`cd_origen` *	Texto (150)	De qué CD puede ser abastecido el SKU.
`consolida_compra`	Texto (150)	Consolidación del stock para decisión de compra (a nivel SKU-sucursal).
`max_stock`	Numérico (decimal)	Stock máximo permitido del SKU en la sucursal. Acepta decimales. **Opcional**: si se omite, toma el valor por defecto `-1` (sin límite). Disponible desde v1.09.

* Campos obligatorios. **Llave natural:** `(id_material, id_sucursal)`.

> **NOVEDAD V1.09** Se agrega el campo **opcional** `max_stock` (numérico, acepta decimales) para definir el stock máximo permitido del SKU en la sucursal. Si no se envía, toma el valor por defecto `-1` (sin límite). Cambio aditivo y retrocompatible: las integraciones que no envíen el campo siguen funcionando sin ajustes.

2.11.4. Ejemplo de salida llamada POST `201`

```
{
  "mensaje": "Instancia creada correctamente.",
  "creado": true, "creados": 1, "actualizados": 0, "total_procesados": 1,
  "data": { /* registro completo */ }
}
```

2.11.5. Ejemplo de llamada DELETE

```
curl -X 'DELETE' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/config_sku_sucursal/?
id_material=AX305070&id_sucursal=12345' \
  -H 'Authorization: token <API-KEY>'
```

2.11.6. Truncar (operación destructiva)

Vacía por completo la tabla de **configuración SKU por sucursal**: borra **todos** los registros del recurso. Disponible desde v1.08.

DELETE
https://api.<CLIENTE>.supply-syncro.cl/api/v1/config_sku_sucursal/truncar/?confirm=true

CONFIRMACIÓN OBLIGATORIA Requiere el parámetro exacto `?confirm=true`. Sin él, la API responde `400` y no toca la tabla.

```
curl -X 'DELETE' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/config_sku_sucursal/truncar/?confirm=true' \
  -H 'Authorization: token <API-KEY>'
```

Respuesta exitosa `200`: `{ "mensaje": "Tabla truncada. Se eliminaron N registro(s).", "eliminados": N }`.

2.12. Descuento

Descripción: Permite obtener, registrar y eliminar descuentos y promociones que tenga la empresa.

GET POST DELETE
https://api.<CLIENTE>.supply-syncro.cl/api/v1/descuento/

Parámetros de entrada (GET / DELETE)

- `id_descuento` — Código del descuento

CAMBIO V1.06 El filtro `id_descuento` en el GET ahora funciona correctamente. En la versión 1.05 producía un error 500 al ser enviado.

2.12.1. Ejemplo de llamada GET

```
curl -X 'GET' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/descuento/' \
  -H 'accept: application/json' \
  -H 'Authorization: token <API-KEY>'
```

2.12.2. Ejemplo de salida llamada GET

```
{
  "links": { "next": null, "previous": null },
  "count": 8, "total_pages": 1,
  "results": [
    {
      "id_descuento": "id_desc_1",
      "descripcion": "descuento1",
      "custom1": "", "custom2": "", "custom3": "", "custom4": ""
    }
  ]
}
```

2.12.3. Ejemplo de llamada POST

```
curl -X 'POST' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/descuento/' \
  -H 'Authorization: token <API-KEY>' \
  -H 'Content-Type: application/json' \
  -d '{
    "id_descuento": "id_desc_50",
    "descripcion": "descuento50"
  }'
```

2.12.3.1. Descripción de campos

Campo	Tipo	Descripción
`id_descuento` *	Texto (150)	Código del descuento.
`descripcion` *	Texto (150)	Descripción del descuento (ej. "50%", "2x1", "3x2").
`custom1, …, custom4`	Texto (150)	Características adicionales.

* Campos obligatorios. **Llave natural:** `(id_descuento)`.

2.12.4. Ejemplo de salida llamada POST 201

```
{
  "mensaje": "Instancia creada correctamente.",
  "creado": true, "creados": 1, "actualizados": 0, "total_procesados": 1,
  "data": { /* registro completo */ }
}
```

2.12.5. Ejemplo de llamada DELETE

```
curl -X 'DELETE' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/descuento/?id_descuento=id_desc_50' \
  -H 'Authorization: token <API-KEY>'
```

2.12.6. Truncar (operación destructiva)

Vacía por completo la tabla de **descuento**: borra **todos** los registros del recurso. Disponible desde v1.08.

DELETE
`https://api.<CLIENTE>.supply-syncro.cl/api/v1/descuento/truncar/?confirm=true`

> **CONFIRMACIÓN OBLIGATORIA** Requiere el parámetro exacto `?confirm=true`. Sin él, la API responde **400** y no toca la tabla.

```
curl -X 'DELETE' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/descuento/truncar/?confirm=true' \
  -H 'Authorization: token <API-KEY>'
```

Respuesta exitosa **200**: `{ "mensaje": "Tabla truncada. Se eliminaron N registro(s).", "eliminados": N }`.

2.13. Jerarquía

Descripción: Permite obtener, registrar y eliminar las diferentes jerarquías que se configuran para la empresa.

GET **POST** **DELETE**
`https://api.<CLIENTE>.supply-syncro.cl/api/v1/jerarquia/`

Parámetros de entrada (GET / DELETE)

- `id_jerarquia` — Código de la jerarquía

2.13.1. Ejemplo de llamada GET

```
curl -X 'GET' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/jerarquia/' \
  -H 'accept: application/json' \
  -H 'Authorization: token <API-KEY>'
```

2.13.2. Ejemplo de salida llamada GET

```
{
  "links": { "next": null, "previous": null },
  "count": 8, "total_pages": 1,
  "results": [
    {
      "id_jerarquia": "jer1",
      "descripcion": "jerarquia1",
      "custom1": "", "custom2": "", "custom3": "", "custom4": ""
    }
  ]
}
```

2.13.3. Ejemplo de llamada POST

```
curl -X 'POST' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/jerarquia/' \
  -H 'Authorization: token <API-KEY>' \
  -H 'Content-Type: application/json' \
  -d '{
    "id_jerarquia": "jerarquia2",
    "descripcion": "descripcion jerarquia2"
  }'
```

2.13.3.1. Descripción de campos

Campo	Tipo	Descripción
`id_jerarquia` *	Texto (150)	Código de la jerarquía utilizada en maestro proveedor.
`descripcion` *	Texto (150)	Descripción de la jerarquía.
`custom1, …, custom4`	Texto (150)	Características adicionales.

* Campos obligatorios. **Llave natural:** `(id_jerarquia)`.

2.13.4. Ejemplo de salida llamada POST 201

```
{
  "mensaje": "Instancia creada correctamente.",
  "creado": true, "creados": 1, "actualizados": 0, "total_procesados": 1,
  "data": { /* registro completo */ }
}
```

2.13.5. Ejemplo de llamada DELETE

```
curl -X 'DELETE' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/jerarquia/?id_jerarquia=jerarquia2' \
  -H 'Authorization: token <API-KEY>'
```

2.13.6. Truncar (operación destructiva)

Vacía por completo la tabla de **jerarquía**: borra **todos** los registros del recurso. Disponible desde v1.08.

DELETE
`https://api.<CLIENTE>.supply-syncro.cl/api/v1/jerarquia/truncar/?confirm=true`

> **CONFIRMACIÓN OBLIGATORIA** Requiere el parámetro exacto `?confirm=true`. Sin él, la API responde **400** y no toca la tabla.

```
curl -X 'DELETE' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/jerarquia/truncar/?confirm=true' \
  -H 'Authorization: token <API-KEY>'
```

Respuesta exitosa **200**: `{ "mensaje": "Tabla truncada. Se eliminaron N registro(s).", "eliminados": N }`.

2.14. Conversión de unidades

Descripción: Permite obtener, registrar y eliminar conversiones de unidad de medida entre la unidad base y formatos de despacho específicos.

GET **POST** **DELETE**
`https://api.<CLIENTE>.supply-syncro.cl/api/v1/conversion_unidades/`

Parámetros de entrada (GET / DELETE)

- `id_despacho` — Identificador del formato de despacho
- `id_material` — Código del producto

2.14.1. Ejemplo de llamada GET

```
curl -X 'GET' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/conversion_unidades/' \
  -H 'accept: application/json' \
  -H 'Authorization: token <API-KEY>'
```

2.14.2. Ejemplo de salida llamada GET

```
{
  "links": { "next": null, "previous": null },
  "count": 1, "total_pages": 1,
  "results": [
    {
      "id_despacho": "CAJA",
      "id_material": "SKU001",
      "um_base": "UN",
      "um_despacho": "CAJA",
      "conversion": 12
    }
  ]
}
```

2.14.3. Ejemplo de llamada POST

```
curl -X 'POST' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/conversion_unidades/' \
  -H 'Authorization: token <API-KEY>' \
  -H 'Content-Type: application/json' \
  -d '{
    "id_despacho": "CAJA",
    "id_material": "SKU001",
    "um_base": "UN",
    "um_despacho": "CAJA",
    "conversion": 12
  }'
```

2.14.3.1. Descripción de campos

Campo	Tipo	Descripción
`id_despacho` *	Texto (150)	Identificador del formato de despacho.
`id_material` *	Texto (150)	Código del producto.
`um_base`	Texto (150)	Unidad de medida base.
`um_despacho`	Texto (150)	Unidad de medida de despacho.
`conversion`	Numérico	Factor de conversión entre unidades.

* Campos obligatorios. **Llave natural:** `(id_despacho, id_material)`.

2.14.4. Ejemplo de salida llamada POST 201

```
{
  "mensaje": "Instancia creada correctamente.",
  "creado": true, "creados": 1, "actualizados": 0, "total_procesados": 1,
  "data": { /* registro completo */ }
}
```

2.14.5. Ejemplo de llamada DELETE

```
curl -X 'DELETE' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/conversion_unidades/?
id_despacho=CAJA&id_material=SKU001' \
  -H 'Authorization: token <API-KEY>'
```

2.14.6. Truncar (operación destructiva)

Vacía por completo la tabla de **conversión de unidades**: borra **todos** los registros del recurso. Disponible desde v1.08.

DELETE
`https://api.<CLIENTE>.supply-syncro.cl/api/v1/conversion_unidades/truncar/?confirm=true`

CONFIRMACIÓN OBLIGATORIA Requiere el parámetro exacto `?confirm=true`. Sin él, la API responde **400** y no toca la tabla.

```
curl -X 'DELETE' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/conversion_unidades/truncar/?
confirm=true' \
  -H 'Authorization: token <API-KEY>'
```

Respuesta exitosa 200: `{ "mensaje": "Tabla truncada. Se eliminaron N registro(s).", "eliminados": N }`.

3. Métodos de consulta (solo lectura)

Los métodos de esta sección **entregan información calculada por Supply Syncro**. A diferencia de los métodos de carga de la sección 2, solo exponen **GET**: no reciben registros, no borran nada y no tienen llave natural ni operación `truncar/`. Se autentican con el mismo token (`Authorization: token <API-KEY>`).

3.1. Resultado Abastecimiento (solo lectura)

Descripción: Permite consultar los resultados del proceso de abastecimiento generados por el motor de cálculo de Supply Syncro. Este recurso entrega, por la combinación *sucursal* × *producto*, las cantidades de despacho recomendado, ajustado y posible.

> **SOLO LECTURA** Este endpoint expone únicamente el método **GET**. El contenido es generado y mantenido por procesos internos de Supply Syncro; no se permite la creación, modificación ni eliminación de registros vía API.

GET
`https://api.<CLIENTE>.supply-syncro.cl/api/v1/resultado_abastecimiento/`

Disponible en v1.06 en adelante.

Parámetros de entrada

- `id_sucursal` — Filtra por código de la sucursal *(opcional)*
- `id_material` — Filtra por código del producto *(opcional)*
- `page` — Número de página, default `1`
- `page_size` — Tamaño de página, default `1000` , máximo `50000`

3.1.1. Ejemplo de llamada GET

```
curl -X 'GET' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/resultado_abastecimiento/' \
  -H 'accept: application/json' \
  -H 'Authorization: token <API-KEY>'
```

Filtrando por sucursal y producto

```
curl -X 'GET' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/resultado_abastecimiento/?
id_sucursal=2000-A010&id_material=SKU001' \
  -H 'accept: application/json' \
  -H 'Authorization: token <API-KEY>'
```

3.1.2. Ejemplo de salida llamada GET 200


```
{
  "links": {
    "next": "https://api.<CLIENTE>.supply-syncro.cl/api/v1/resultado_abastecimiento/?
page_size=2&page=2",
    "previous": null
  },
  "count": 125256,
  "total_pages": 62628,
  "send_date": "2026-05-14",
  "results": [
    {
      "id_sucursal": "2000-A010",
      "id_material": "SKU001",
      "despacho_recomendado": 12.0,
      "despacho_ajustado": 10.0,
      "despacho_posible": 8.0
    },
    {
      "id_sucursal": "2000-A010",
      "id_material": "SKU002",
      "despacho_recomendado": 0.0,
      "despacho_ajustado": null,
      "despacho_posible": 5.0
    }
  ]
}
```


3.1.3. Descripción de campos de salida

Campos de nivel superior del body (al mismo nivel que `count` y `total_pages`):

Campo	Tipo	Descripción
`send_date`	Fecha (YYYY-MM-DD)	Fecha en que el proceso de abastecimiento generó los resultados que se están consultando. Toda la respuesta corresponde a una sola ejecución del proceso, por lo que `send_date` es el mismo para todas las filas. Disponible desde v1.07. Puede ser `null` si el dato no está disponible.

Campos por cada elemento del array `results`:

Campo	Tipo	Descripción
`id_sucursal`	Texto (255)	Código de la sucursal.
`id_material`	Texto (255)	Código del producto.
`despacho_recomendado`	Decimal	Cantidad de despacho recomendada por el motor de cálculo (basada en la política de inventario configurada).
`despacho_ajustado`	Decimal	Cantidad ajustada manualmente respecto al recomendado (cuando un usuario edita el resultado).
`despacho_posible`	Decimal	Cantidad efectivamente despachable considerando restricciones de stock disponible y límites operativos.

> **VALORES NULOS** Cualquier campo decimal puede retornar `null` cuando el motor de cálculo no produjo un valor para esa combinación sucursal × producto.

3.1.4. Paginación

El endpoint sigue el mismo patrón de paginación del resto de la API (ver **sección 1.1.2**): la respuesta incluye `links.next`, `links.previous`, `count` (total) y `total_pages`. Cuando ambos links son `null` los resultados caben en una sola página.

> **RECOMENDACIÓN** Para consultas masivas, prefiera filtrar por `id_sucursal` y/o `id_material` antes de iterar por todas las páginas. Sin filtros, una consulta puede devolver más de 100.000 registros y consumir muchas páginas.

3.2. Indicadores (KPIs): quiebre, sobrestock y GMROI

> **DISPONIBILIDAD** Estos endpoints están habilitados solo para los clientes que tienen contratado el módulo de indicadores por API. Para el resto responden **`404`**. Consulte con su ejecutivo de Supply Syncro para habilitarlos.

Descripción: Entregan los indicadores de quiebre, sobrestock y GMROI que muestra el dashboard de Supply Syncro, con las **mismas cifras y las mismas reglas de cálculo**. Están pensados para integrar los indicadores en sistemas propios del cliente (BI, reportes, alertas).

> **VENTANA FIJA DE 30 DÍAS** Todos los indicadores corresponden a los **últimos 30 días cerrados**: desde hoy menos 30 días hasta ayer. El día en curso no se incluye porque todavía no está cerrado. La ventana no se puede cambiar por parámetro; cada respuesta informa las fechas exactas en `fecha_desde` y `fecha_hasta`.

ACTUALIZACIÓN Los indicadores se recalculan una vez al día, durante el proceso nocturno de Supply Syncro, a partir de los datos cargados con los métodos de la sección 2 (demanda, inventario, configuración SKU por sucursal, etc.). Si una carga llega con retraso, el indicador de ese día se actualiza en la corrida siguiente. Las respuestas pueden tardar hasta 1 hora en reflejar una corrida nueva.

Endpoint	Entrega	Filtros opcionales
`/api/v1/kpi/resumen/`	Resumen de los tres indicadores en una sola llamada	—
`/api/v1/kpi/quiebre/`	Resumen + serie diaria	`id_sucursal`, `canal_venta`, `id_material`
`/api/v1/kpi/sobrestock/`	Resumen + serie diaria	`id_sucursal`, `canal_venta`, `id_material`
`/api/v1/kpi/gmroi/`	Resumen (acumulado de la ventana)	`canal_venta`, `id_material`

Los filtros son opcionales y se combinan entre sí (Y lógico). Cada uno recibe un solo valor. En `id_material` se acepta tanto el código original del producto como su código homologado, si el cliente trabaja con homologación de códigos.

3.2.1. Resumen

GET `https://api.<CLIENTE>.supply-syncro.cl/api/v1/kpi/resumen/`

Disponible desde v1.12. Sin parámetros.

```
curl -X 'GET' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/kpi/resumen/' \
  -H 'accept: application/json' \
  -H 'Authorization: token <API-KEY>'
```

Ejemplo de salida `200`


```
{
  "fecha_desde": "2026-09-16",
  "fecha_hasta": "2026-10-15",
  "quiebre": {
    "promedio_diario_quiebres": 412.37,
    "quiebres_ultimo_dia": 398,
    "skus_quebrados_ultimo_dia": 211
  },
  "sobrestock": {
    "monto_sobrestock_ultimo_dia": 18234567.12,
    "monto_sobrestock_promedio_diario": 17950321.40,
    "porcentaje_sobrestock_ultimo_dia": 21.39,
    "calidad_porcentaje_ultimo_dia": 56.73
  },
  "gmroi": {
    "gmroi": 0.3184,
    "ventas": 95123456.00,
    "ventas_sin_rebate": 98765432.00,
    "margen": 28123456.00,
    "inventario_promedio": 88321456.00,
    "dias_con_inventario": 31
  }
}
```


Cada bloque es idéntico al `resumen` del endpoint específico del indicador (secciones 3.2.2 a 3.2.4). Si el indicador de GMROI no está disponible para el cliente, el bloque `gmroi` viene en `null` y los otros dos se entregan igual.

3.2.2. Quiebre

Definición: un **quiebre** es una combinación *producto* × *sucursal* que, en un día, tenía demanda pronosticada y **no tenía stock** en la sucursal. Se cuentan combinaciones distintas: un producto quebrado en 3 sucursales suma 3 quiebres, aunque se venda por varios canales.

- Solo se consideran las sucursales marcadas para medición de quiebre en el maestro de sucursales.
- No se cuentan los productos sin reposición, las sucursales inactivas ni las sucursales que no informan stock.
- No se cuentan las combinaciones producto × sucursal que no están en la configuración SKU por sucursal (sin canal de reposición configurado).

GET
`https://api.<CLIENTE>.supply-syncro.cl/api/v1/kpi/quiebre/`

Disponible desde v1.12.

Parámetros de entrada

- `id_sucursal` — Código de la sucursal *(opcional)*

- `canal_venta` — Canal de venta *(opcional)*
- `id_material` — Código del producto *(opcional)*

```
curl -X 'GET' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/kpi/quiebre/?id_sucursal=2000-A010' \
  -H 'accept: application/json' \
  -H 'Authorization: token <API-KEY>'
```

Ejemplo de salida `200`

```
{
  "fecha_desde": "2026-09-16",
  "fecha_hasta": "2026-10-15",
  "resumen": {
    "promedio_diario_quiebres": 412.37,
    "quiebres_ultimo_dia": 398,
    "skus_quebrados_ultimo_dia": 211
  },
  "causas_etiquetas": {
    "1": "Quiebre en CD",
    "2": "Máximo stock",
    "3": "Gestión de replenishment",
    "4": "En camino",
    "5": "Demanda",
    "9": "Sin determinar"
  },
  "results": [
    {
      "fecha": "2026-09-16",
      "rellenado": false,
      "quiebres": 405,
      "skus_quebrados": 215,
      "causas": { "1": 118, "3": 84, "5": 152, "9": 51 }
    },
    ...
  ]
}
```

Descripción de campos de salida

Campo	Tipo	Descripción
`resumen.promedio_diario_quiebres`	Decimal	Promedio diario de quiebres en la ventana. Es el indicador "Quiebres PDV" del dashboard.
`resumen.quiebres_ultimo_dia`	Entero	Quiebres del último día de la ventana (ayer).
`resumen.skus_quebrados_ultimo_dia`	Entero	Productos distintos quebrados en al menos una sucursal el último día.
`causas_etiquetas`	Objeto	Nombre de cada código de causa usado en `results[].causas`.
`results[].fecha`	Fecha	Día, formato AAAA-MM-DD. La serie trae los 30 días, en orden.
`results[].quiebres`	Entero	Combinaciones producto × sucursal quebradas ese día.
`results[].skus_quebrados`	Entero	Productos distintos quebrados ese día.
`results[].causas`	Objeto	Desglose de `quiebres` por causa (código → cantidad). Las partes suman el total del día. Las causas sin quiebres no aparecen.
`results[].rellenado`	Booleano	`true` cuando el día no tiene foto de inventario (o no registró quiebres) y repite los valores del último día válido, igual que el gráfico del dashboard. Evita que un día sin carga de inventario se vea como un salto artificial.

3.2.3. Sobrestock

Definición: el **sobrestock** es el inventario on hand que supera el **límite máximo** calculado por Supply Syncro para cada producto × sucursal, valorizado a costo unitario:

`sobrestock = Σ (inventario on hand − límite máximo) × costo unitario`, sumando solo donde el inventario supera el límite.

- Corresponde al indicador "Calidad de inventario on hand en tiendas" del dashboard: solo sucursales de venta a público (tiendas, sucursales, minoristas, almacenes), con el inventario físicamente disponible en el local.
- Solo productos presentes en la configuración SKU por sucursal. Las sucursales cuyo stock está consolidado en otra no se suman dos veces.
- Cuando un producto se vende por varios canales, su inventario se reparte entre ellos según la participación de cada canal en la venta; por eso el filtro `canal_venta` entrega la porción del inventario atribuible a ese canal.

GET
`https://api.<CLIENTE>.supply-syncro.cl/api/v1/kpi/sobrestock/`

Disponible desde v1.12.

Parámetros de entrada

- `id_sucursal` — Código de la sucursal *(opcional)*
- `canal_venta` — Canal de venta *(opcional)*
- `id_material` — Código del producto *(opcional)*

```
curl -X 'GET' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/kpi/sobrestock/' \
  -H 'accept: application/json' \
  -H 'Authorization: token <API-KEY>'
```

Ejemplo de salida `200`

```
{
  "fecha_desde": "2026-09-16",
  "fecha_hasta": "2026-10-15",
  "resumen": {
    "monto_sobrestock_ultimo_dia": 18234567.12,
    "monto_sobrestock_promedio_diario": 17950321.40,
    "porcentaje_sobrestock_ultimo_dia": 21.39,
    "calidad_porcentaje_ultimo_dia": 56.73
  },
  "results": [
    {
      "fecha": "2026-09-16",
      "rellenado": false,
      "monto_sobrestock": 17654321.00,
      "monto_optimo": 51234567.00,
      "monto_faltante": 6612345.00,
      "monto_inventario": 68888888.00,
      "calidad_porcentaje": 67.86
    },
    ...
  ]
}
```

Descripción de campos de salida

Campo	Tipo	Descripción
`resumen.monto_sobrestock_ultimo_dia`	Decimal	Sobrestock valorizado del último día de la ventana (ayer).
`resumen.monto_sobrestock_promedio_diario`	Decimal	Promedio diario del sobrestock valorizado en la ventana.
`resumen.porcentaje_sobrestock_ultimo_dia`	Decimal	Sobrestock del último día como porcentaje del inventario valorizado total.
`resumen.calidad_porcentaje_ultimo_dia`	Decimal	Calidad de inventario del último día (ver `calidad_porcentaje`).
`results[].fecha`	Fecha	Día, formato AAAA-MM-DD. La serie trae los 30 días, en orden.
`results[].monto_sobrestock`	Decimal	Inventario por sobre el límite máximo, a costo.
`results[].monto_optimo`	Decimal	Inventario dentro del límite máximo, a costo.
`results[].monto_faltante`	Decimal	Lo que falta para llegar al inventario de seguridad, a costo (inventario que debería haber y no hay).
`results[].monto_inventario`	Decimal	Inventario on hand total, a costo (`monto_optimo + monto_sobrestock`).
`results[].calidad_porcentaje`	Decimal	`monto_optimo / (monto_inventario + monto_faltante) × 100`. 100% = todo el inventario está dentro de su límite y no falta stock de seguridad.
`results[].rellenado`	Booleano	`true` cuando el día no tiene datos válidos y repite los valores del último día válido, igual que el dashboard.

3.2.4. GMROI

Definición: el **GMROI** (retorno del margen bruto sobre la inversión en inventario) indica cuánto margen generó cada peso invertido en inventario durante la ventana:

`GMROI = margen bruto de los últimos 30 días / inventario promedio a costo`

- **Margen bruto** = venta neta − costo de lo vendido. La venta neta descuenta el rebate comercial del canal cuando corresponde; la venta antes de ese descuento se informa en `ventas_sin_rebate`.
- **Inventario promedio** = inventario on hand valorizado a costo, promediado sobre los días de la ventana que tienen foto de inventario.
- Solo productos activos y con reposición habilitada en el maestro de productos.
- Es un GMROI **de 30 días, no anualizado**: un GMROI de 0,30 en 30 días equivale aproximadamente a 3,6 anual.

GET
`https://api.<CLIENTE>.supply-syncro.cl/api/v1/kpi/gmroi/`

Disponible desde v1.12. Sin filtro por sucursal.

Parámetros de entrada

- `canal_venta` — Canal de venta *(opcional)*
- `id_material` — Código del producto *(opcional)*

```
curl -X 'GET' \
  'https://api.<CLIENTE>.supply-syncro.cl/api/v1/kpi/gmroi/' \
  -H 'accept: application/json' \
  -H 'Authorization: token <API-KEY>'
```

Ejemplo de salida 200

```
{
  "fecha_desde": "2026-09-16",
  "fecha_hasta": "2026-10-15",
  "resumen": {
    "gmroi": 0.3184,
    "ventas": 95123456.00,
    "ventas_sin_rebate": 98765432.00,
    "margen": 28123456.00,
    "inventario_promedio": 88321456.00,
    "dias_con_inventario": 31
  }
}
```

Descripción de campos de salida

Campo	Tipo	Descripción
`resumen.gmroi`	Decimal	`margen / inventario_promedio`, con 4 decimales. `0` si no hay inventario.
`resumen.ventas`	Decimal	Venta neta de la ventana.
`resumen.ventas_sin_rebate`	Decimal	Venta antes del descuento por rebate comercial.
`resumen.margen`	Decimal	Venta neta menos costo de lo vendido.
`resumen.inventario_promedio`	Decimal	Inventario promedio a costo.
`resumen.dias_con_inventario`	Entero	Días con foto de inventario usados para el promedio.

> **SIN SERIE DIARIA** El GMROI se calcula sobre la ventana completa (margen e inventario acumulados), por lo que no se entrega día a día.

3.2.5. Códigos de respuesta

- `200` — Consulta exitosa.
- `404` — El módulo de indicadores no está habilitado para el cliente.
- `401` — Falta el token o no es válido.
- `503` — Solo en `/kpi/gmroi/`: el indicador de GMROI no está habilitado para el cliente. Body: `{ "mensaje": "..." }`.
- `500` — Error interno. Body: `{ "mensaje": "...", "detalle": "..." }`.

4. Cambios respecto a la versión 1.05

Esta sección resume las mejoras y correcciones aplicadas en la versión 1.06. Para integraciones existentes, leer especialmente la subsección de **cambios incompatibles**.

4.1. Correcciones de errores

Endpoint	Antes (v1.05)	Ahora (v1.06)
`GET /descuento/?id_descuento=X`	Error 500 (campo de filtro incorrecto)	Funciona correctamente
`GET /inventario_transito_entre_sucursales/?id_sucursal_origen=X`	Error 500 (campo de filtro inexistente)	Funciona correctamente
`GET /inventario_onhand/?fecha=X&id_sucursal=Y&id_material=Z`	Filtro `fecha` ignorado al venir junto con los otros	Aplica los tres filtros
`POST` con update exitoso	Respondía "*Error en el guardado.*" con HTTP 200, induciendo a confusión	Responde "*Instancia actualizada correctamente.*" y status correcto
`POST` bulk con un item inválido	Podía dejar registros parcialmente guardados	Hace rollback completo (atomicidad garantizada)
`POST /demanda_despacho/` sin `id_despacho`	Error 500 por NOT NULL en BD	Se almacena como string vacío automáticamente
Sin token o token inválido	Respondía 403 (incorrecto)	Responde 401 (correcto)

4.2. Nuevos endpoints

- **`GET /api/v1/resultado_abastecimiento/`** — Permite consultar los resultados del proceso de abastecimiento (despacho recomendado, ajustado y posible) por sucursal y producto. Endpoint de solo lectura. Ver **sección 3.1**.

4.3. Mejoras en respuestas

- **POST** ahora siempre devuelve contadores: `creados`, `actualizados`, `total_procesados`.
- **DELETE** ahora incluye `eliminados` (cantidad real de filas afectadas) además del flag `eliminado`.

- Errores de validación devueltos como JSON estructurado `{mensaje, errores: {campo: [...]}}`, no como string concatenado.
- Errores 500 ya no exponen stack traces al cliente; sólo un campo `detalle` con el tipo y mensaje del error.
- POST single ahora retorna el registro completo en el campo `data`.

4.4. Operaciones destructivas

El endpoint `/inventario_transito_desde_proveedores/truncar/` ahora **requiere el parámetro** `?confirm=true` en el query string. Sin él, la API responde 400 sin tocar la tabla.

4.5. Cambios incompatibles (breaking changes)

> **ATENCIÓN** Si su integración asume el comportamiento anterior, debe ajustarse antes de migrar a v1.06.

1. **DELETE rechaza con 400 si faltan parámetros obligatorios.** En v1.05 silenciosamente borraban 0 filas. Ahora la API responde con la lista `faltantes`. Asegúrese de que sus ETLs envíen todos los parámetros requeridos.
2. **Texto del campo `mensaje` en POST cambió.** Si parsea el campo `mensaje` para detectar errores, migre a usar el HTTP status code (400 / 500) o el campo estructurado `errores`. La frase "*Error en el guardado.*" ya no se utiliza en respuestas exitosas (era un bug histórico).
3. **Truncate exige `?confirm=true`.** Cualquier rutina automatizada que llame al endpoint debe agregar el parámetro.

4.6. Soporte

Para reportar problemas, solicitar nuevos endpoints o coordinar la migración de su integración a la v1.06, contáctenos en **info@gpisoftware.cl**.

5. Cambios respecto a la versión 1.06

Esta sección resume los cambios aplicados en la versión 1.07. Son todos **aditivos y retrocompatibles** con v1.06: las integraciones existentes no requieren ajustes para seguir funcionando.

5.1. Nuevo campo `send_date` en `/resultado_abastecimiento/`

La respuesta del endpoint `GET /api/v1/resultado_abastecimiento/` ahora incluye, al mismo nivel que `count` y `total_pages`, un nuevo campo:

- **`send_date`** — Fecha (formato `YYYY-MM-DD`) en que el proceso de abastecimiento generó los resultados consultados. Como toda la respuesta corresponde a una sola ejecución del proceso, el valor es único para toda la respuesta. Devuelve `null` si la información no está disponible.

Ver la sección **3.1.2 / 3.1.3** para el ejemplo de salida y la descripción de campos actualizada.

> **RETROCOMPATIBILIDAD** Los clientes que ya consumen este endpoint en v1.06 seguirán funcionando sin cambios. El nuevo campo aparecerá automáticamente en la respuesta — los parsers que ignoren campos desconocidos no se ven afectados.

6. Cambios respecto a la versión 1.07

Esta sección resume los cambios aplicados en la versión 1.08. Son todos **aditivos y retrocompatibles** con v1.07: las integraciones existentes no requieren ajustes para seguir funcionando.

6.1. Endpoint `truncar/` para todos los recursos

El endpoint de truncate destructivo, que en v1.07 solo existía para `inventario_transito_desde_proveedores`, ahora está disponible para **todos los recursos CRUD**:

- `DELETE /api/v1/maestro_producto/truncar/?confirm=true`
- `DELETE /api/v1/maestro_proveedor/truncar/?confirm=true`
- `DELETE /api/v1/maestro_cliente/truncar/?confirm=true`
- `DELETE /api/v1/maestro_sucursal/truncar/?confirm=true`
- `DELETE /api/v1/rel_proveedor_sku/truncar/?confirm=true`
- `DELETE /api/v1/lead_times_internos/truncar/?confirm=true`
- `DELETE /api/v1/inventario_transito_desde_proveedores/truncar/?confirm=true`
- `DELETE /api/v1/inventario_transito_entre_sucursales/truncar/?confirm=true`
- `DELETE /api/v1/config_sku_sucursal/truncar/?confirm=true`
- `DELETE /api/v1/descuento/truncar/?confirm=true`
- `DELETE /api/v1/jerarquia/truncar/?confirm=true`
- `DELETE /api/v1/conversion_unidades/truncar/?confirm=true`

Todos comparten el mismo comportamiento: borran **todos** los registros de la tabla del recurso en el tenant y exigen el parámetro exacto `?confirm=true`. Sin él, la API responde `400` sin tocar la tabla. Respuesta exitosa `200`: `{ "mensaje": "...", "eliminados": <n> }`.

> **RETROCOMPATIBILIDAD** Cambio puramente aditivo: el endpoint existente de `inventario_transito_desde_proveedores` no cambia. Los demás recursos simplemente ganan la nueva ruta `truncar/`.

7. Cambios respecto a la versión 1.08

Esta sección resume los cambios aplicados en la versión 1.09. Son todos **aditivos y retrocompatibles** con v1.08: las integraciones existentes no requieren ajustes para seguir funcionando.

7.1. Nuevo campo `max_stock` en configuración SKU por sucursal

El recurso `config_sku_sucursal` incorpora el campo **opcional** `max_stock` (numérico, acepta decimales) para definir el **stock máximo permitido** de un SKU en una sucursal.

- Se expone en las respuestas **GET** y se acepta en el cuerpo del **POST** (upsert) del recurso.
- Es **no obligatorio**: si el cliente lo omite al crear o actualizar un registro, toma el valor por defecto `-1` (sin límite).
- La llave natural del upsert no cambia: sigue siendo `(id_material, id_sucursal)`.

> **RETROCOMPATIBILIDAD** Cambio puramente aditivo. Las integraciones existentes que no envíen `max_stock` siguen funcionando sin ajustes y sus registros quedan con el valor por defecto `-1`. Los parsers que ignoren campos desconocidos no se ven afectados por la aparición del nuevo campo en las respuestas GET.

8. Cambios respecto a la versión 1.09

Esta sección resume los cambios aplicados en la versión 1.10. Son todos **retrocompatibles** con v1.09: relajan campos que antes eran obligatorios, así que ninguna integración existente requiere ajustes.

8.1. `fecha_est_llegada` pasa a ser opcional en tránsito entre sucursales

En el recurso `inventario_transito_entre_sucursales` el campo `fecha_est_llegada` deja de ser obligatorio en el **POST**. Los clientes que no conocen la fecha estimada de llegada de un traslado ya no necesitan inventar un valor para poder cargar el registro.

- Si se omite (o se envía `null`), el registro se guarda **sin fecha estimada de llegada** y así se devuelve en el **GET**.
- El campo **sigue formando parte de la llave natural**: `(fecha_pedido, id_material, id_sucursal_origen, id_sucursal_destino, fecha_est_llegada)`. Esto es lo que permite que una misma orden entregada de forma **parcializada** (por ejemplo 300 unidades una fecha y 200 otra) quede como filas distintas y no se pisen entre sí.
- Reenviar el mismo registro sin fecha **actualiza** la fila existente, no crea una nueva: la carga sigue siendo idempotente.
- En el **DELETE** el parámetro `fecha_est_llegada` ya era opcional y no cambia: si se omite, se borran todas las entregas parciales de esa llave.

> **RECOMENDACIÓN** Envíe la fecha estimada de llegada siempre que la conozca. Los cálculos de abastecimiento la usan para ubicar el inventario en tránsito dentro del horizonte de planificación; un traslado sin fecha no puede asignarse a un período futuro.

El recurso `inventario_transito_desde_proveedores` **no cambia**: ahí `fecha_est_llegada` sigue siendo obligatoria.

8.2. `jerarquia_1` a `jerarquia_4` pasan a ser opcionales en maestro de productos

En el recurso `maestro_producto`, los niveles de jerarquía `jerarquia_1`, `jerarquia_2`, `jerarquia_3` y `jerarquia_4` dejan de estar marcados como obligatorios, igual que `jerarquia_5` a `jerarquia_7`. Un producto puede cargarse informando solo los niveles que el cliente maneja.

- Los niveles omitidos quedan vacíos y se devuelven como `null` en el **GET**.
- La llave natural del upsert no cambia: sigue siendo `(id_material)`.
- El resto de los campos obligatorios del recurso se mantiene sin cambios: `id_material`, `descripcion`, `estado`, `reposicion` y `um_base`.

RECOMENDACIÓN Las jerarquías son la base de los agrupamientos de los reportes y del análisis por categoría. Informarlas sigue siendo lo recomendado; la flexibilización existe para no bloquear la carga del maestro cuando algún nivel no está definido en el origen.

RETROCOMPATIBILIDAD Ambos cambios solo **relajan** validaciones. Las integraciones que ya envían estos campos siguen funcionando exactamente igual y sus datos no se ven afectados.

9. Cambios respecto a la versión 1.10

Esta sección resume los cambios aplicados en la versión 1.11.

9.1. `numero_pedido` pasa a formar parte de la llave en tránsito desde proveedores

En el recurso `inventario_transito_desde_proveedores` el número de pedido se incorpora a la **llave natural** del upsert. La nueva llave es:

`(fecha_pedido, numero_pedido, id_material, id_proveedor, id_sucursal_destino, fecha_est_llegada)`

Hasta v1.10, dos órdenes de compra **distintas** del mismo producto, proveedor y sucursal de destino, con iguales fechas de pedido y de llegada, se identificaban como el mismo registro: la segunda reemplazaba a la primera y se perdía parte del inventario en tránsito. Desde v1.11 cada número de pedido se guarda como una fila propia.

- **POST:** no cambian los campos ni su obligatoriedad (`numero_pedido` ya era obligatorio). Reenviar el mismo registro con el mismo número de pedido sigue **actualizando** la fila existente: la carga sigue siendo idempotente.
- **GET:** se agrega el filtro opcional `numero_pedido` .
- **DELETE:** se agrega el parámetro opcional `numero_pedido` . Si se omite, se borran los registros de todos los pedidos que coinciden con el resto de la llave, igual que antes.
- El recurso `inventario_transito_entre_sucursales` **no cambia**.

IMPORTANTE Si su integración corregía un pedido reenviándolo con un **número de pedido distinto** al original, ahora se crea una fila nueva y la anterior permanece. En ese caso, borre el registro anterior (DELETE con su `numero_pedido`) o use el endpoint `truncar/` antes de la carga completa.

10. Cambios respecto a la versión 1.11

Esta sección resume los cambios aplicados en la versión 1.12.

10.1. Nuevos endpoints de consulta de indicadores (KPIs)

Se agregan cuatro endpoints de solo lectura que entregan los indicadores del dashboard de Supply Syncro de los últimos 30 días cerrados. Ver **sección 3.2**.

- `GET /api/v1/kpi/resumen/` — Resumen de quiebre, sobrestock y GMROI en una llamada.
- `GET /api/v1/kpi/quiebre/` — Quiebres en punto de venta, con serie diaria y desglose por causa.
- `GET /api/v1/kpi/sobrestock/` — Sobrestock valorizado en tiendas, con serie diaria.
- `GET /api/v1/kpi/gmroi/` — GMROI de los últimos 30 días.

10.2. Reorganización de la documentación

La sección 2 queda dedicada a los **métodos de carga** (GET / POST / DELETE de cada recurso). La nueva sección 3 agrupa los **métodos de consulta** de solo lectura: Resultado Abastecimiento (antes sección 2.15, ahora 3.1) y los nuevos indicadores (3.2). Los capítulos de cambios de versiones anteriores pasan a numerarse desde el 4.

> **RETROCOMPATIBILIDAD** No hay cambios en los endpoints existentes. Las integraciones actuales siguen funcionando exactamente igual.