COBI.shipping

COBI.shipping is a REST-API service that acts as a connector between **Barcode Shipping** and **SAP Business One**. Its primary purpose is to exchange delivery and shipping data:

1. It retrieves open delivery notes from SAP Business One and
2. reports back shipping information (including tracking numbers, weights, and item details) after the shipment has been processed.

Features

Delivery note retrieval: COBI.shipping provides Barcode Shipping with open (or optionally closed) delivery notes from SAP Business One. Only delivery notes with allowed shipping methods and unassigned items (not yet packed) are included.

Shipping confirmation: After a shipment has been processed in Barcode Shipping, COBI.shipping sends shipping details back into SAP Business One, including tracking numbers, shipping weight, and item-package assignments.

Automatic delivery note creation: If no delivery note exists for an order, COBI.shipping can automatically generate one based on the corresponding Sales Order, ensuring correct serial number or bin location assignment.

Status updates: The service updates delivery notes with shipping statuses (“In progress” for partial shipments or “Completed” when fully shipped) and can optionally trigger automatic invoice creation.

Integration with Coresuite Print: After invoice creation, COBI.shipping can trigger automatic printing through the COBI.wms Printservice, using predefined layouts and print definitions.

Shop/Partner feedback: Once a shipment is completed, COBI.shipping can send status updates (including tracking numbers, delivery note references, and customer codes) back to connected shops or partners.

Implementation details

1. Delivered as a Windows service (installation via included batch files).
2. Requires Barcode Shipping software and Java 11.
3. API endpoints are provided under `/api/v1/` with authentication through an API key.
4. Configuration is stored in SAP Business One (`@CMP_CONFIG`) and must be reloaded by restarting the service after changes.
5. Supports customizable UDF mappings for address fields, tracking numbers, and commodity codes.
6. Optional automatic invoice creation and Coresuite Print integration.
7. Reads address and shipping information directly from SAP table `DLN12`.

Required UDFs in SAP Business One

For COBI.shipping to function properly, several **User-Defined Fields (UDFs)** must be created in SAP Business One. These fields store address information, shipping metadata, and customs information that are required by the integration.

Under *Delivery Notes → Lines (DLN12)*:

Title	Data type	Size	Description
cobi-shipping_firstNameField	Alphanumeric	50	First name of the recipient
cobi-shipping_lastNameField	Alphanumeric	50	Last name of the recipient
cobi-shipping_companyField	Alphanumeric	100	Company name of the recipient
cobi-shipping_phoneField	Alphanumeric	30	Phone number of the recipient
cobi-shipping_emailField	Alphanumeric	100	Email address of the recipient

Under *Delivery Notes → Header*:

Title	Data type	Size	Description
cobi-shipping_trackingInHeaderField	Alphanumeric	100	Field to store tracking number(s) in delivery note
cobi-shipping_alarmGroupId	Numeric	11	Alarm group for invoice errors

Under *Items (OITM)*:

Title	Data type	Size	Description
cobi-shipping_oitmCommodityCodeField	Alphanumeric	20	Customs tariff code for export/import handling

Installation

1. Install Barcode Shipping

* Download and install Barcode Shipping from the internal archive (`N:\Entwicklung\Barcode Shipping\Downloads.7z`). * During setup, choose **Shop → Eigentwicklung** and **Shipper → Einfacher Versender**, then enable additional functions as needed. * Install **Java 11** (required by Barcode Shipping). Instructions: https://www.barcodeshipping.de/java-installation/ * Log in to Barcode Shipping with COMP.net credentials.

2. Install COBI.shipping service

* Extract the `cobi-shipping` ZIP archive (recommended path: `C:\cobi-shipping`). * Configure the connection values in the file `cobi-shipping-service.exe.config` (SAP database settings). * Run the provided batch files:

- `InstallService.bat` → installs the Windows service.
- `DeleteService.bat` → removes the Windows service.

3. Configure Barcode Shipping

* Open Barcode Shipping → File → Settings. * General:

- Default shipper: *Einfacher Versender*
- Disable *Check order status*
- Enable *Mouse Mode*
- Default shop: *Connector*

* Connector:

- API URL: `http:*<COBI.shipping server IP>:<Port>/api/v1/` (example: `http://127.0.0.1:62005/api/v1/`) * API Version: `V2` * API Key: `Y29iaXNoaXBwaW5nOkRoNUchNnM=` (Base64 of `cobishipping:Dh5G!6s`) **4. Configure COBI.shipping in SAP Business One** * Go to **Extras → User-Defined Windows → @CMP_CONFIG**. * Set the configuration values as documented (see sections *UDFs* and *Optional toggles*). * Mark shipping methods as "Barcode Shipping enabled": **Administration → Definition → Inventory Management → Shipping Methods**. ⚠ **Note:** Configuration values are only loaded at service startup.
  After making changes, always restart the COBI.shipping Windows service. ===== Optional toggles (Boolean, 0/1) ===== All of the following configuration keys are stored in **@CMP_CONFIG** and are read **on service start** (restart the COBI.shipping service after changes). ^ Key ^ Meaning (EN) ^ | `cobi-shipping_companyFromCardName` | Use **CardName** as company name on the delivery address. | | `cobi-shipping_firstLastNameFromContacts` | Take **first/last name** from the linked **Contact Person**. | | `cobi-shipping_firstNameLastNameFromCardName` | Derive **first/last name** from **CardName**. | | `cobi-shipping_companyFirstNameLastNameFromAddress3` | Pull **company / first / last name** from **Address3** field. | | `cobi-shipping_addressFromAddress2` | Use **Address2** as the street address. | | `cobi-shipping_companyAddressFromAddress2` | Take **name/company** from **Address2**. | | `cobi-shipping_privateAddressFromAddress2` | Treat **Address2** as a **private** address. | | `cobi-shipping_createInvoiceAfterFullyPacked` | **Auto-create Invoice** when the delivery is fully packed. | | `cobi-shipping_invoiceDateEqDeliveryNoteDate` | Set **Invoice Date = Delivery Note Date** for auto-created invoices. | | `cobi-shipping_displayClosedDeliveryNotes` | Also **show closed Delivery Notes** to Barcode Shipping. | ===== Related non-boolean keys (for reference) ===== These are **not** toggles, but often configured alongside the above: ^ Key ^ Type / Example ^ Purpose ^ | `cobi-shipping_trackingInHeaderFieldName` | Text (UDF name in Delivery header) | Stores tracking number(s) in the delivery header. | | `cobi-shipping_alarmGroupId` | Numeric (e.g., `11`) | Alarm group used for invoice errors. | | `cobi-shipping_oitmCommodityCodeField` | Text (UDF name on `OITM`) | Customs tariff code mapping. | | `cobi-shipping_alternativeRecipient` | Text (UDF name) | Alternative recipient field. | > Note: Address field mappings like `cobi-shipping_firstNameField`, `..._lastNameField`, `..._companyField`, `..._phoneField`, `..._emailField` remain part of the **address configuration** (DLN12 mapping). ===== Coresuite Print configuration ===== If the system is configured for automatic invoice creation, COBI.shipping can trigger a Coresuite Print job right after the invoice is generated. This requires that the COBI.wms Printservice is installed and reachable. ^ Key ^ Type / Example ^ Purpose ^ | cobi-shipping_printService | URL (e.g. http://127.0.0.1:62005) | URL of the COBI.wms Printservice endpoint | | cobi-shipping_printDefinition | Text / ID (e.g. DeliveryNoteLayout) | Print Definition ID to be used for output | | cobi-shipping_layoutId | Text / ID (e.g. 1001) | Layout ID for the document | | cobi-shipping_printMode | Text (Print or Email) | Determines whether the invoice is printed or sent by email |

From:
https://docs.cobisoft.de/wiki/ - **COBISOFT Documentation**

Permanent link:
https://docs.cobisoft.de/wiki/cobi.shipping?rev=1755862132

Last update: **2025/08/22 11:28**