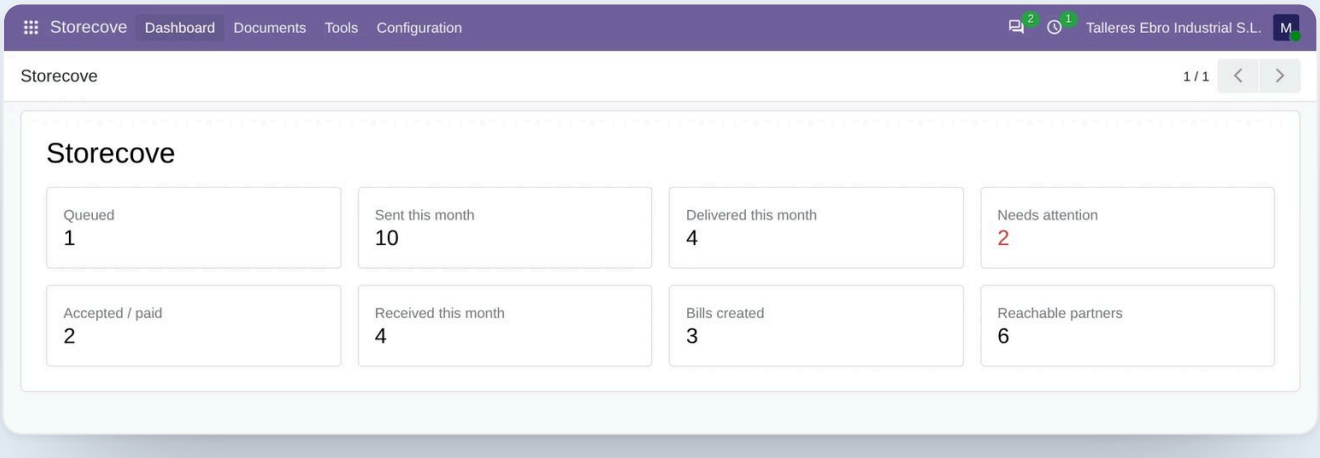
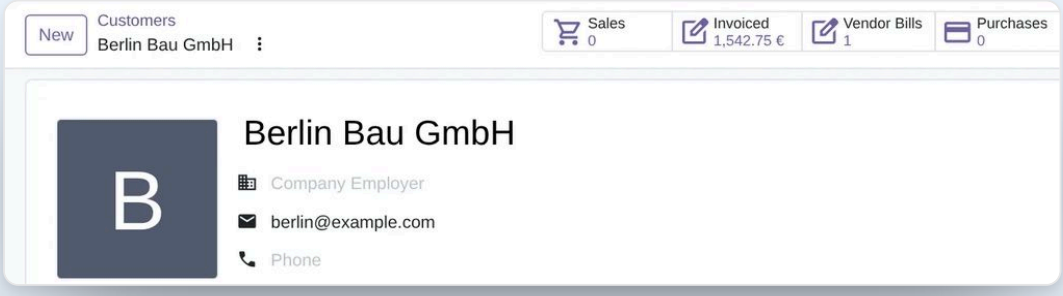
CONNECTOR GUIDE · E-INVOICING

Storecove connector for Odoo

Send and receive e-invoices and orders from Odoo through Storecove, which delivers them over Peppol and other networks.

Odoo 17 · 18 · 19 · 20 | 14-day trial, no card | Edition of 18-11-2026





Real Odoo 20 screenshot. The summary of the app and a contact's accessibility on Storecove.

Contents

On the right, the page where each chapter starts. Menu and button names are those of Odoo in English.

1	Who it is for and what it solves	3
2	The connector in figures	4
3	Where you see it in your Odoo	5
4	What you need before you start	6
5	The FlexigoTech Cloud trial	6
6	Connecting your account step by step	8
7	Legal entities and identifier schemes in Storecove	10
8	E-invoicing on the contact form and reachability check	12
9	Sending e-invoices and credit notes through Storecove	16
10	Tax categories for e-invoicing	20
11	Purchase orders via Peppol	21
12	Invoices and orders you receive from your suppliers	23
13	Tests and sending raw documents	27
14	Dashboard and sync log	29
15	Data going out and coming back	32
16	What happens on its own	33
17	The account options and their defaults	34
18	Security and personal data	36
19	Multiple accounts and multiple companies	36
20	If something fails	37
21	Other ways of doing this	38
22	Plans and pricing	39
23	Frequently asked questions	40
24	Glossary	43
25	Checklist and contact	44

This guide describes version 20.0.1.2.4 of the module (Odoo branch 20.0, commit a76bb9d4c). Everything in it comes from its published code, the platform's official documentation or real screenshots.

The screenshots come from a test Odoo 20 database, with a made-up company and data and no real connection to Storecove. The module also has a version for Odoo 17, 18 and 19.

1 Who it is for and what it solves

> The connector requires a FlexigoTech Cloud subscription, with a 14-day trial and no card required. Without the trial or the subscription, it does not talk to Storecove.

Storecove is an e-invoicing platform. It receives your invoice, converts it into the format each destination network requires, and delivers it. It is a Peppol access point and appears on the OpenPeppol list of service providers. With the connector you send customer invoices and credit notes from Odoo, which Storecove delivers via Peppol or other networks, such as SDI, KSeF or FACeB2B, and purchase orders via Peppol. The connector also brings in the invoices and orders sent to you.

You need your own Storecove account with its API key. Storecove's fees are not included.

ew Upload Invoices ⋮ Search... 1-10 / 10

Number	Customer	Invoice Date	Due Date	Tax Excluded	Total	Amount Due	Status
INV/2026/00010	Berlin Bau GmbH	13 Nov	2 days ago	825.00 €	998.25 €	998.25 €	Posted
INV/2026/00009	Iberia Componentes SA	11 Nov	4 days ago	2,250.00 €	2,722.50 €	2,722.50 €	Posted
INV/2026/00008	Porto Ferramentas Lda	9 Nov	6 days ago	1,350.00 €	1,633.50 €	1,633.50 €	Posted
INV/2026/00007	Stockholm Design AB	7 Nov	8 days ago	600.00 €	726.00 €	726.00 €	Posted
INV/2026/00006	Antwerp Logistics NV	5 Nov	10 days ago	1,575.00 €	1,905.75 €	1,905.75 €	Posted
INV/2026/00005	Lyon Distribution SARL	3 Nov	12 days ago	900.00 €	1,089.00 €	1,089.00 €	Posted
INV/2026/00004	Milano Servizi SRL	1 Nov	2 weeks ago	375.00 €	453.75 €	453.75 €	Posted
INV/2026/00003	Rotterdam Trading BV	30 Oct	2 weeks ago	900.00 €	1,089.00 €	1,089.00 €	Posted
INV/2026/00002	Berlin Bau GmbH	28 Oct	2 weeks ago	450.00 €	544.50 €	544.50 €	Posted
INV/2026/00001	Iberia Componentes SA	26 Oct	2 weeks ago	150.00 €	181.50 €	181.50 €	Posted
				9,375.00 €	**11,343.75 €**	**11,343.75 €**	

- **Real Odoo 20 screenshot.** The list of customer invoices.

New Submissions ⋮ Search... 1-11 / 1

Created on	Display Reference	Invoice	Evidence Network	State
15 Nov, 17:18	82926e62-f5ce-4f2c-a08b-c935dc121d68			Sent
15 Nov, 17:18	INV/2026/00010	INV/2026/00010		Under Query
15 Nov, 17:18	INV/2026/00009	INV/2026/00009		Failed
15 Nov, 17:18	INV/2026/00008	INV/2026/00008		Queued
15 Nov, 17:18	INV/2026/00007	INV/2026/00007		Rejected
15 Nov, 17:18	INV/2026/00006	INV/2026/00006	peppol	Paid
15 Nov, 17:18	INV/2026/00005	INV/2026/00005	peppol	Accepted
15 Nov, 17:18	INV/2026/00004	INV/2026/00004		Acknowledged
15 Nov, 17:18	INV/2026/00003	INV/2026/00003	peppol	Delivered
15 Nov, 17:18	INV/2026/00002	INV/2026/00002		Cleared (Tax Authority)
15 Nov, 17:18	INV/2026/00001	INV/2026/00001		Sent

- **Real Odoo 20 screenshot.** The **Submissions** list, with the status of each document sent.

What changes day to day

- You send the invoice directly from the invoice itself, with its status visible in the **Submissions** list.

- Every five minutes a task sends whatever is queued.
- Documents you receive appear in **Received Documents** and become draft vendor bills in Odoo, or sales orders if they are orders.
- With **Check Partner Reachability** you can see which customers receive via the network before sending.
- The **Sync Log** records every call and every notice from Storecove.

What it does not do

- It does not include Storecove's fees: the account and the API key are yours.
- Received invoices come in as draft vendor bills and someone approves them in Odoo.
- It does not include custom configuration, import of historical data or migrations of your database.
- It only offers the seven common tax categories; those specific to some countries are not.

2 The connector in figures

4
Odoo versions (17, 18, 19 and 20)

2
scheduled tasks, active on installation

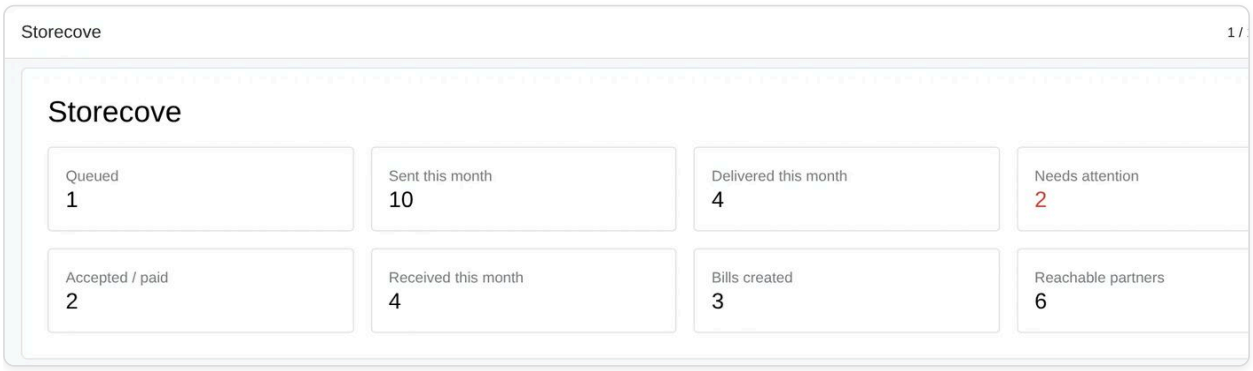
3
accounts on the Standard plan

10
accounts on the Pro plan

298
automated tests

It works on Odoo 17, 18, 19 and 20, with the Invoicing, Purchase and Sales apps installed. The module is its own app and depends on FlexigoTech Cloud. According to Odoo Apps, version 20.0 has 8,005 lines of code, counting FlexigoTech Cloud. The automated tests simulate Storecove's responses according to its documentation.

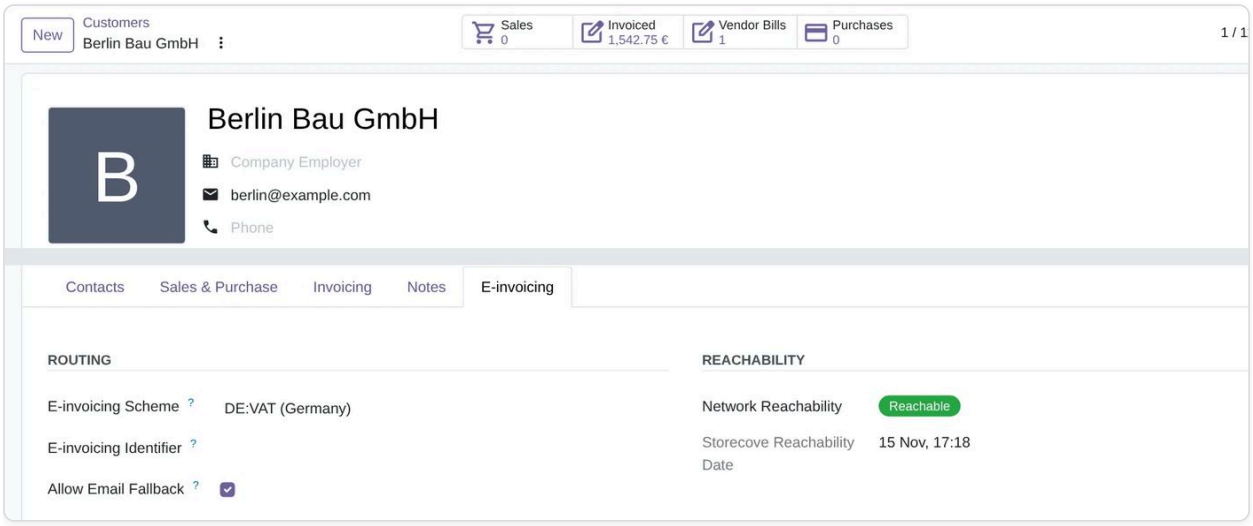
3 Where you see it in your Odoo



- **Real Odoo 20 screenshot.** The app summary, with the counters for sent and received documents.

Dashboard	The app summary.
Documents	**Submissions**, with the documents you send, and **Received Documents**, with those you receive.
Tools	**Send Test Invoice**, **Check Partner Reachability** and **Send Raw Document (UBL/CII/IDOC)**.
Configuration	**Accounts**, **Legal Entities**, **Identifier Schemes**, **Tax Mappings** and **Sync Log**. This menu, **Send Test Invoice** and **Send Raw Document (UBL/CII/IDOC)** are only visible to the connector's **Manager** role.

Outside its own app, the connector extends Odoo's invoices, purchase orders, contacts and units of measure. On the contact form it adds an **E-invoicing** tab with routing and accessibility, and on the invoice, a **Preview Storecove Route** button.



- **Real Odoo 20 screenshot.** A contact's **E-invoicing** tab, with its accessibility on Storecove.

4 What you need before you start

- Your own Storecove account (sandbox or production) and its API key. Storecove's fees are not included.
- A FlexigoTech Cloud subscription or its 14-day trial with no card required. The API key of each account is stored encrypted in FlexigoTech Cloud and not in your Odoo.
- Odoo 17, 18, 19 or 20, on Odoo.sh or on your own server. It does not work on Odoo Online, which does not allow installing third-party modules.
- An Odoo administrator user: only an administrator can accept the consent and save the API key.

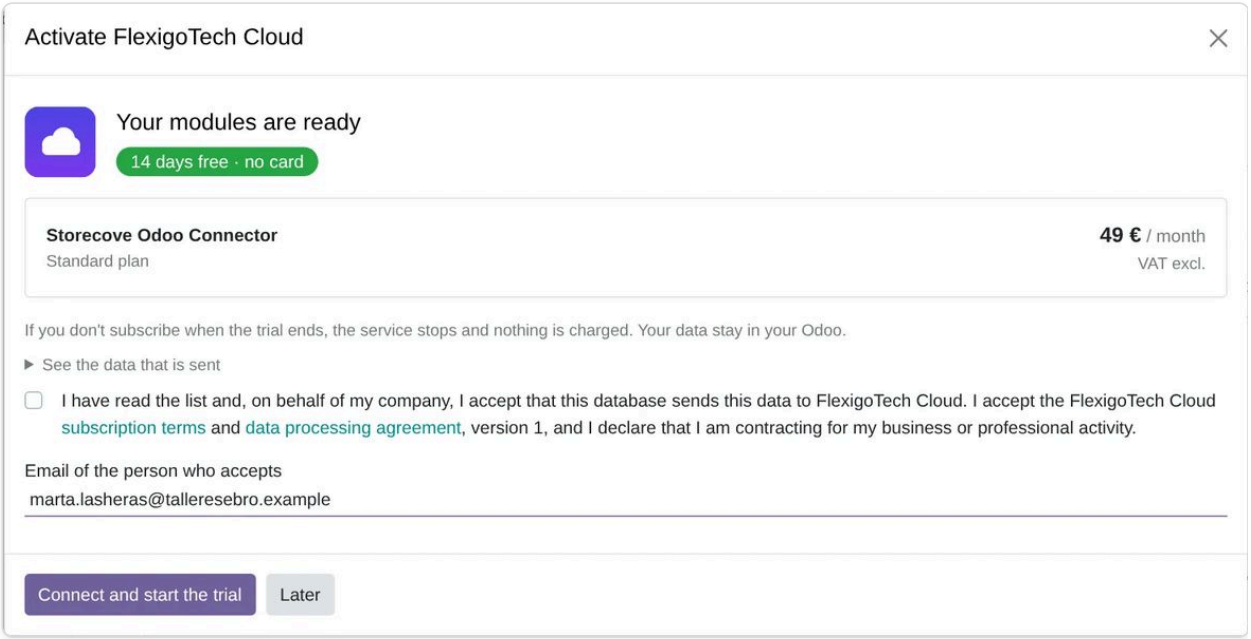
> The connector does not register you with Storecove or create your account, nor does it obtain the API key for you: you get it yourself from Storecove. Storecove's fees are your responsibility and are not included in the connector.

It is installed like any third-party Odoo module, on Odoo.sh or on your server. Odoo installs `flexigotech_cloud`, the connection to FlexigoTech Cloud, together with the apps it depends on: Invoicing, Purchase and Sales.

5 The FlexigoTech Cloud trial

1 **Click Activate in the top bar**

It appears as soon as you install the module. It opens the **Activate FlexigoTech Cloud** window, with the Standard plan and its 14 free days, no card required.



- **Real Odoo 20 screenshot.** The activation window: read the list of data, tick the box and enter your email.

2 **Accept and enter your email**

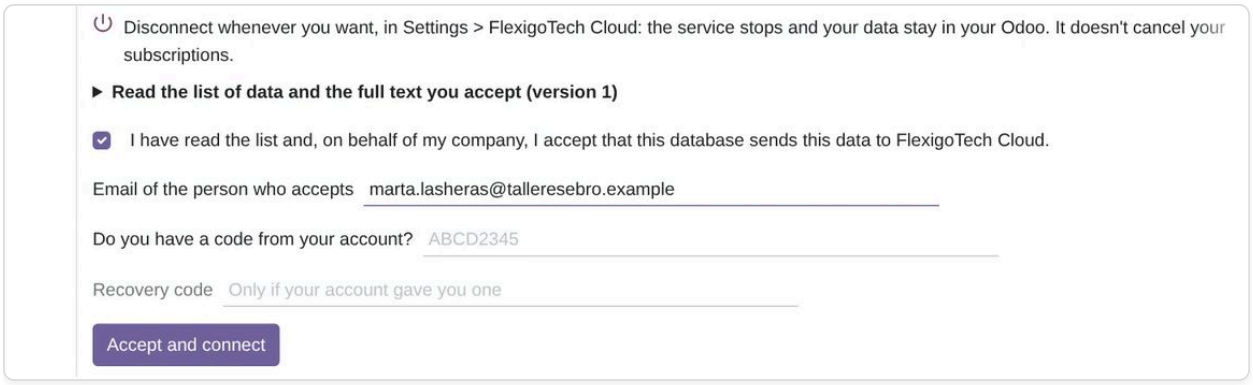
Tick "I have read the list and, on behalf of my company..." (it includes the subscription terms and the data processing agreement) and enter the **Email of the person who accepts**.

3 **Click Connect and start the trial**

Odoo registers the database with FlexigoTech Cloud and starts the module's trial. If you leave it for later, **Later** closes the window.

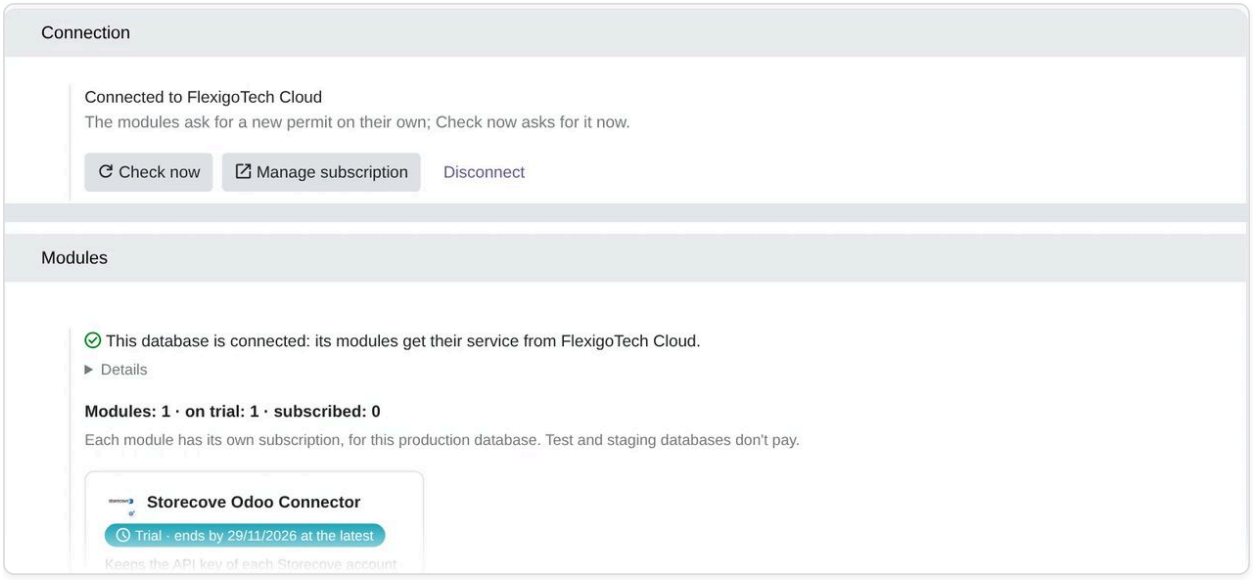
The same, from Settings

In **Settings › FlexigoTech Cloud** you will find the full list of what the database sends. Tick the box, enter the email and click **Accept and connect**. If you already have an account at app.flexigotech.com, the field **Do you have a code from your account?** links the database to it (the code is valid for 10 minutes).



- **Real Odoo 20 screenshot.** Settings › FlexigoTech Cloud, before connecting: the ticked box, the email and **Accept and connect**.

With the database connected, the **Modules** block shows each module with its status: **On trial** and the date it ends, **Subscribed**, etc. From there you choose the plan (Standard or Pro, monthly or yearly) and click **Subscribe**: payment is made on the Stripe page, outside Odoo.



- **Real Odoo 20 screenshot.** Settings › FlexigoTech Cloud with the database connected and the connector on trial.

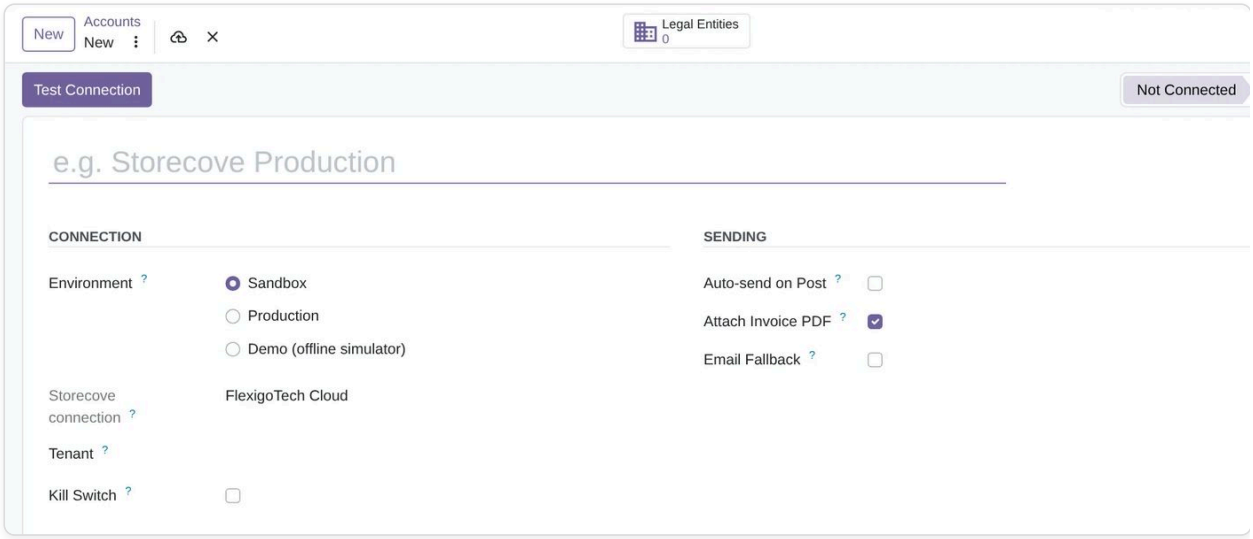
Prices exclude VAT. If you do not subscribe when the trial ends, the service stops and nothing is charged; your data stays in your Odoo. **Check now** requests permission immediately (otherwise the module renews it automatically every 6 hours).

> You cannot start a trial on a neutralised copy (staging) or on a database with demo data: Settings says so with "This database is a neutralised copy or has demo data, so trials can't be started here."

6 Connecting your account step by step

1 Open the accounts menu

Go to **Storecove › Configuration › Accounts** and click **New**. On the new account, **Storecove connection** is already set to **FlexigoTech Cloud** and cannot be changed.



- **Real Odoo 20 screenshot.** The form of a new account, with **Storecove connection** set to **FlexigoTech Cloud**.

2 Enter the name and the environment

Fill in **Name** and choose the **Environment**: **Sandbox**, **Production** or **Demo (offline simulator)**. Sandbox and production are independent Storecove accounts, each with its own keys. **Demo (offline simulator)** never contacts Storecove.

3 Adjust sending

In the **Sending** group you can switch on **Auto-send on Post**, **Attach Invoice PDF** and **Email Fallback**. **Attach Invoice PDF** is on by default.

4 Accept the consent

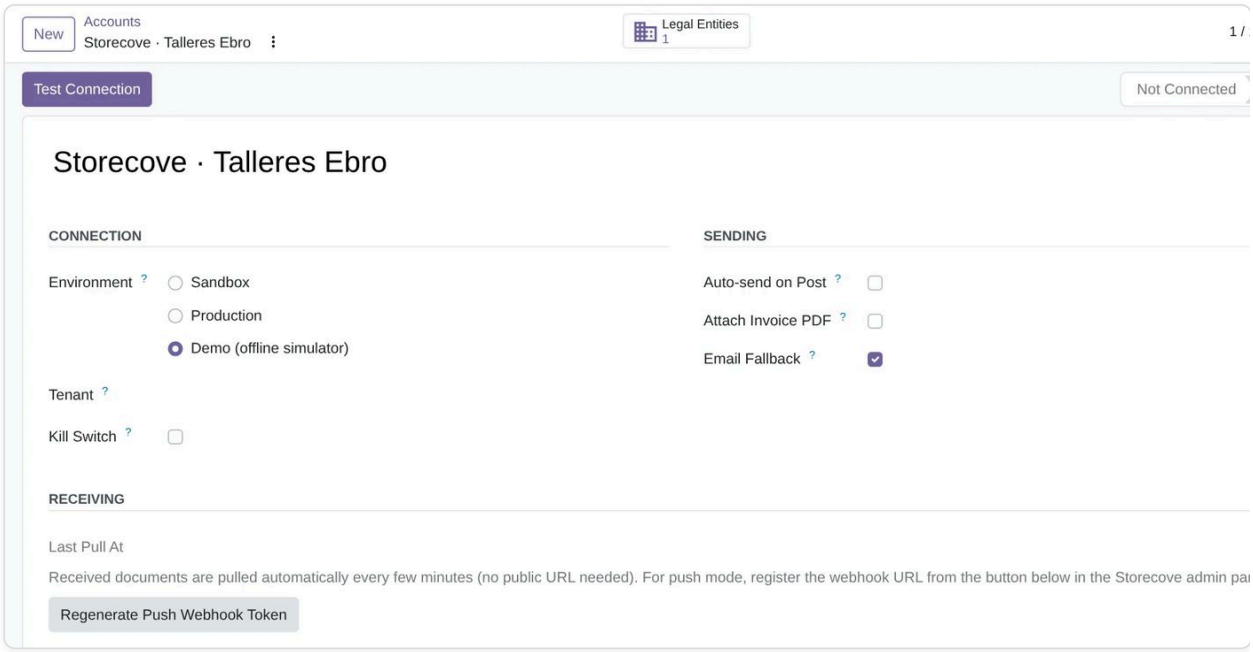
In the **FlexigoTech Cloud (requires FlexigoTech Cloud subscription)** group, tick the box **I accept that my Storecove API key and my calls to Storecove go through FlexigoTech Cloud**.

5 Enter the API key

Fill in **API key (FlexigoTech Cloud)** and save. The key is sent once, encrypted, to FlexigoTech Cloud and is not kept in Odoo. An active trial or subscription is required: without the service, Odoo does not send the key. The date then appears in **Saved in FlexigoTech Cloud**.

6 Test the connection

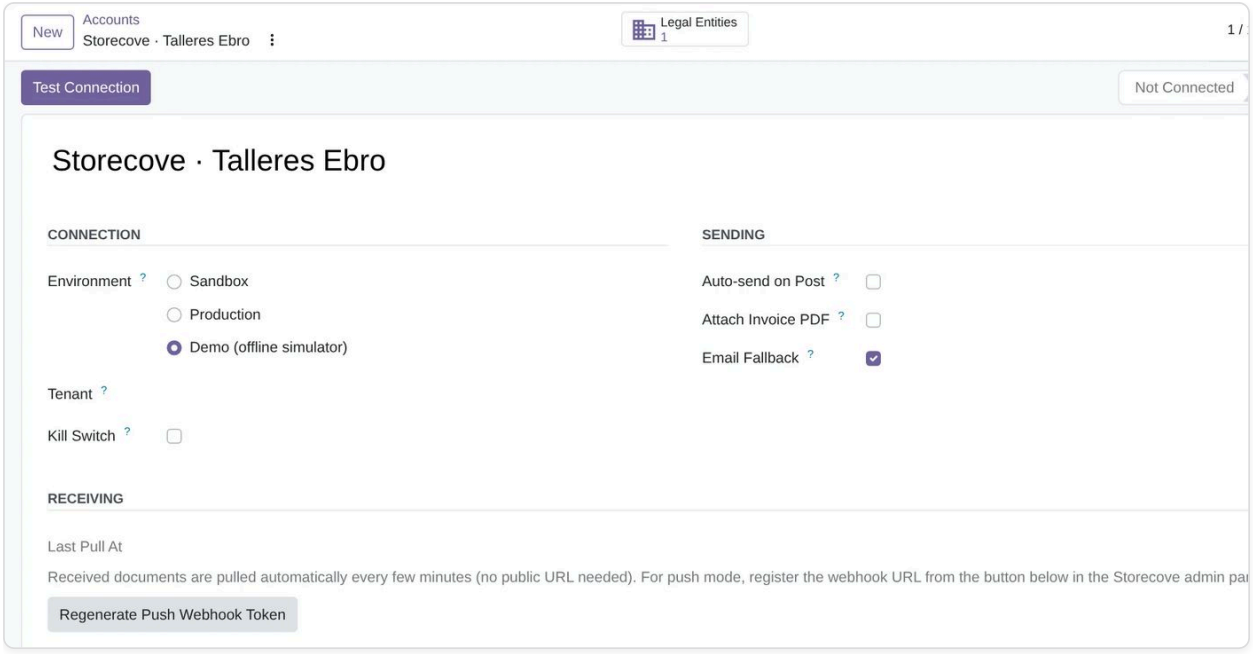
Click **Test Connection**. If all goes well, the account moves to **Connected**; if not, to **Connection Error**.



- **Real Odoo 20 screenshot.** An account with the **Test Connection** and **Regenerate Push Webhook Token** buttons.

> Only an Odoo administrator can accept the consent and enter the API key. Without the consent, the connector does not send the key and warns you with "Before saving the API key, an administrator has to accept that the key and the calls to Storecove go through FlexigoTech Cloud. Nothing was sent." A new account does not store the key in Odoo: if you try to enter it outside the FlexigoTech Cloud section, you get "A new Storecove account keeps its API key in FlexigoTech Cloud, never in Odoo: save it in the FlexigoTech Cloud section of the account."

After connecting, data arrives by two routes. Queued invoices are sent by the **Storecove: send queued e-invoices** task, every 5 minutes. Received documents are retrieved by the **Storecove: poll webhook queue (pull mode)** task, also every 5 minutes, and no public URL is needed. On the account you can see the **Last Pull At** date. The code also has a push mode, in which Storecove calls a URL on your Odoo, but the form does not show that URL. **Regenerate Push Webhook Token** changes the token in it and invalidates the previous one. If **Kill Switch** is ticked, nothing is sent or polled towards Storecove and queued documents wait.

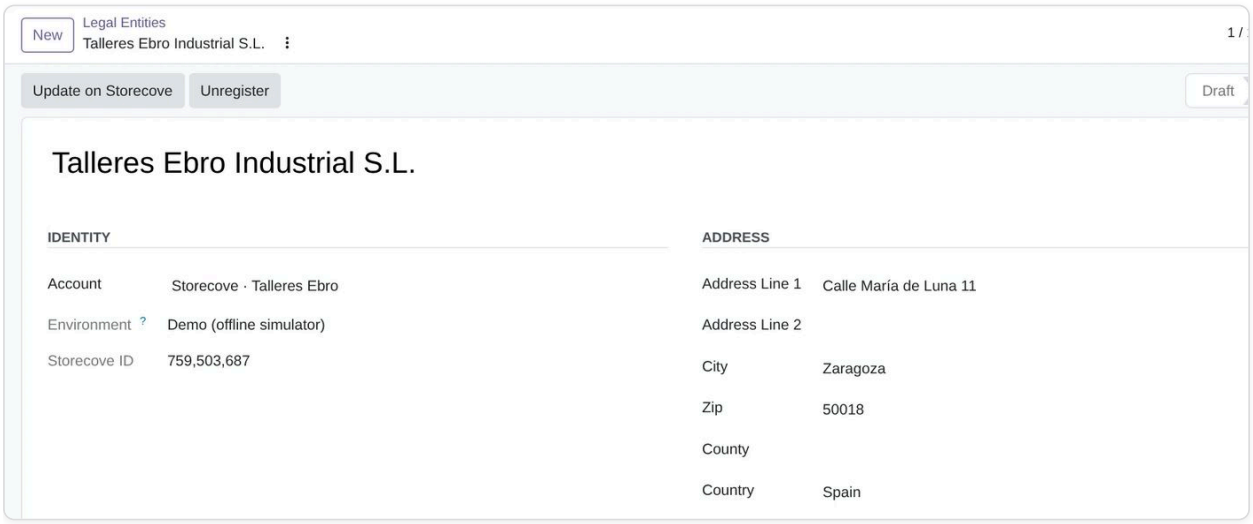


- **Real Odoo 20 screenshot.** The account Talleres Ebro, with its environment, its sending and its receiving.

CONNECTOR GUIDE · STORECOVE

7 Legal entities and identifier schemes in Storecove

Legal entities are under **Storecove › Configuration › Legal Entities**. The catalogue of countries and schemes is under **Storecove › Configuration › Identifier Schemes**. You create the legal entities yourself. The scheme catalogue is loaded on installation, with 86 schemes, and from its list you cannot create or delete schemes.



- **Real Odoo 20 screenshot.** Registered legal entity, with the **Update on Storecove** button.

Register on Storecove	Registers the entity on Storecove. It is hidden when the entity is already in **Registered** status.
Update on Storecove	Sends the changes of an already registered entity. It is only visible in **Registered** status.
Unregister	Deletes the entity on Storecove, together with its identifiers, after asking for confirmation. It is only visible in **Registered** status.

Provision	It is on each line of the **Identifiers** tab and registers that identifier on the exchange network. It is hidden when the identifier is already **Provisioned**.
De-provision	Removes it from the network and returns it to **Draft**. It is only visible with the identifier **Provisioned**.

The **Identifiers** tab lists the entity's Peppol identifiers.

Entity statuses

- **Draft**: the entity is not yet on Storecove. This is the initial status.
- **Registered**: Storecove already has the entity and has given it its **Storecove ID**.
- **Error**: the last registration attempt failed.

> An entity must act as sender, receiver or both. To register it you need the party's name, address line 1, city, postal code and country. An identifier cannot be deleted while it is provisioned: first de-provision it.

What each entity stores

The entity belongs to a Storecove account and represents a company in Odoo: the outgoing invoices of that company are sent through it. When registered, Storecove returns its own identifier, and if it does not, the registration fails.

Roles	**Acts As Sender** and **Acts As Receiver**. The second provisions identifiers on the exchange networks so that it can receive documents.
Public in Peppol Directory	Includes the entity in the public Peppol directory.
Tax Registered	Affects Storecove's data validation when sending.
Advertise Invoice Response	Announces the Peppol invoice response, so that senders receive the events for the invoices you receive and you can send them.
Advertise Order Receiving	Announces that the entity can receive purchase orders.

Receiving and CTC / eReporting

In **Receiving** you choose the **Purchase Journal** for the vendor bills created from received documents. **Auto-create Vendor Bills** and **Auto-create Sales Orders** are switched on and create a draft for each document that can be interpreted.

The **CTC / eReporting** group only appears if the entity's country has a documented tax reporting network that Storecove does not use by default: es-aeat-verifactu (Spain), pl-ksef (Poland), ro-anaf (Romania) or my-lhdnm (Malaysia). **Force CTC Reporting** requests that network on all issued invoices. In Poland and Romania, Storecove must enable it for that entity first.

States

- **Draft**: pending registration. You see **Register on Storecove**.
- **Registered**: synchronised with Storecove. You see **Update on Storecove** and **Unregister**.
- **Error**: the registration did not complete. Correct the data and register again.

Peppol identifiers have their own states: **Draft**, **Provisioned** and **Error**. To provision an identifier, the entity must be registered first.

Rules and warnings

- If registration fails, the entity is set to **Error**, the reason is noted in its history, and a warning is shown.
- If a company has entities registered on more than one Storecove account, for example a demo and a live one, sending stops with a warning that names the accounts, so that nothing is sent through the wrong one. Deactivate the account you do not use for that company.

Identifier schemes

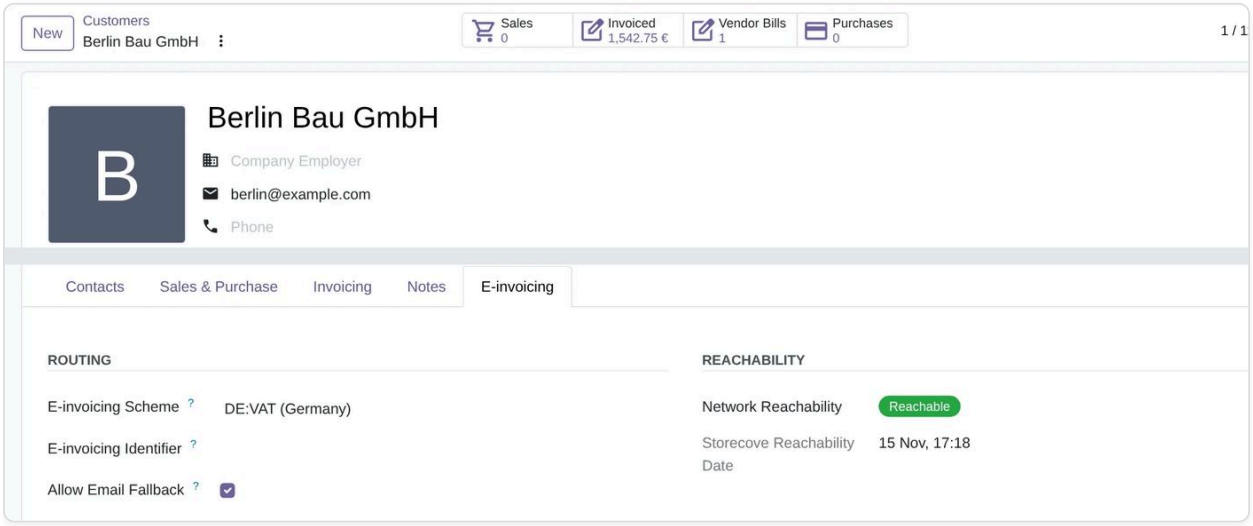
Each scheme shows the country, the name, whether it is a legal, tax or routing identifier, the type of operation (**B2B**, **B2G** or **B2B + B2G**) and, where one exists, a sandbox test identifier. The list is filtered by **Routing** and **With Test Identifier**, grouped by **Country**, and does not allow creating or deleting schemes. For a country with a documented routing chain, the connector follows it in priority order.

Example

Talleres Ebro Industrial S.L. starts in **Draft**. With the account, address and country filled in, you press **Register on Storecove** and it moves to **Registered** with its **Storecove ID**. From then on you can change details and press **Update on Storecove**.

CONNECTOR GUIDE · STORECOVE

8 E-invoicing on the contact form and reachability check



- **Real Odoo 20 screenshot.** **E-invoicing** tab of a reachable contact.

Where each piece of data comes from

You fill in the scheme, the identifier and the fallback-by-email checkbox. The **Guess Identifier** button sets the scheme that corresponds to the contact's country and, if the identifier is empty, takes it from the tax number. In countries whose documented sending chain ends in email, switch on the email fallback as well. The reachability

and its date are written by the connector when you press **Check Reachability**, on the form or from **Storecove › Tools › Check Partner Reachability**. No scheduled task updates them.

Form buttons

Guess Identifier	Fills in the scheme and the identifier from the country and the tax ID. If the country has no known scheme, it warns you with «No known scheme for country "...".»
Check Reachability	Asks Storecove whether the contact can receive e-invoices and stores the result and the date. If there is no account, it warns with "Connect a Storecove account first.".

Network Reachability states

STATE	MEANING
Not checked	The contact has not been checked yet.
Reachable	Storecove finds it on an exchange network.
Not reachable	Storecove does not find it on any network.

Checking several contacts at once

1. **Open the wizard**

   Go to **Storecove › Tools › Check Partner Reachability**.



- **Real Odoo 20 screenshot.** **Check Partner Reachability** wizard.

2. **Choose the contacts**

   In **Contacts** select the ones you want to check.

3. **Decide whether to guess**

   Guess Missing Identifiers is switched on: it fills in scheme and identifier for the contacts that lack them.

4. **Press Check**

   Each contact is checked and the result is stored on the form. Afterwards the **Check** button disappears and **Close** remains.

> If no contacts are selected, the wizard warns you with "Select at least one partner." A failure on one contact does not stop the others from being checked.

The three reachability states

Network Reachability starts at **Not checked**: nobody has asked Storecove yet. **Reachable** means Storecove found the contact on an exchange network with the saved scheme and identifier. **Not reachable** means it found it on none. **Storecove Reachability Date** is read-only and marks when the last check was made.

What is in the E-invoicing tab

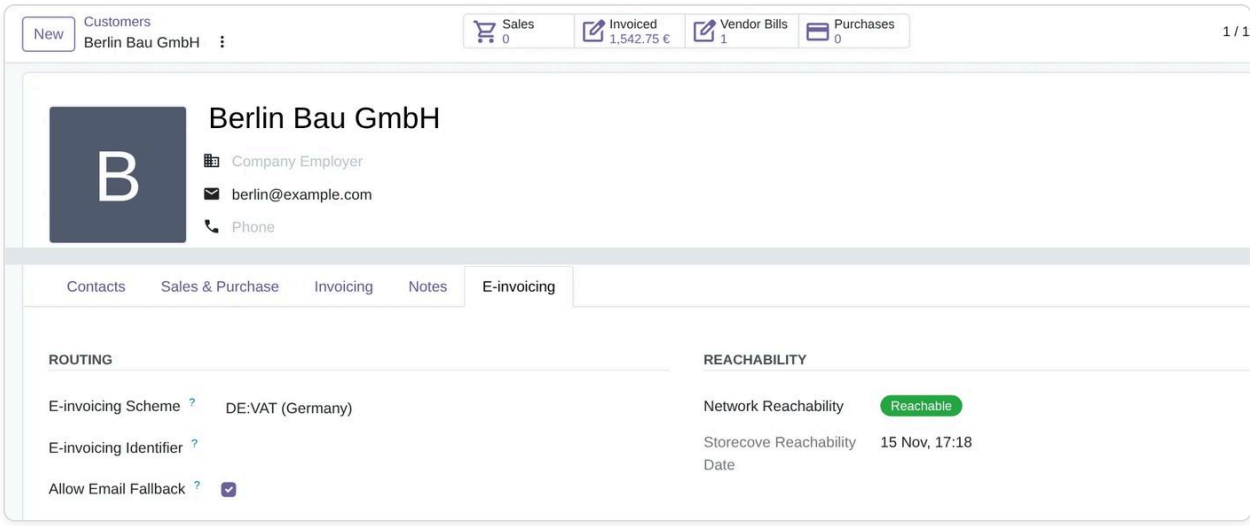
Routing	**E-invoicing Scheme** (for example ES:VAT, NL:KVK or DE:LWID), **E-invoicing Identifier** (for VAT-type schemes, the tax number) and **Allow Email Fallback**.
Reachability	**Network Reachability** and **Storecove Reachability Date**.
Compliance	**Compliance Model** and **Storecove Special Routing Warning**, both computed. The group only appears when the contact's country has a documented model or the chosen scheme requires special routing. The model can be **V-flow (routed through the tax authority)**, **Hard-Clearance Y (pre-clearance required before delivery)** or **Reporting Y (reported to the tax authority, non-blocking)**.
Advanced Routing (GLN)	**Advanced Routing (GLN)**, checkbox off by default, and **GLN Routing Identifier**.

Rules and conditions

- Email fallback only works if the contact allows it and the account allows it too.
- The V-flow and Hard-Clearance Y models may lengthen delivery time. Reporting Y does not block sending.
- The special routing warning appears when the chosen scheme requires a manual, one-off configuration supplied by the recipient, rather than automatic discovery.
- With advanced routing (GLN), documents go to the GLN number, but the usual identifier is still used for the legal identification of the invoice.
- **Guess Identifier** fails with «No known scheme for country "...".» if the contact's country has no scheme.
- **Check Reachability** fails with "Connect a Storecove account first." if there is no account.
- In the bulk check, a contact with an error does not cancel the results for the others.

End-to-end example

1. **Berlin Bau GmbH form**
   Open the contact, go to **E-invoicing** and press **Check Reachability**.



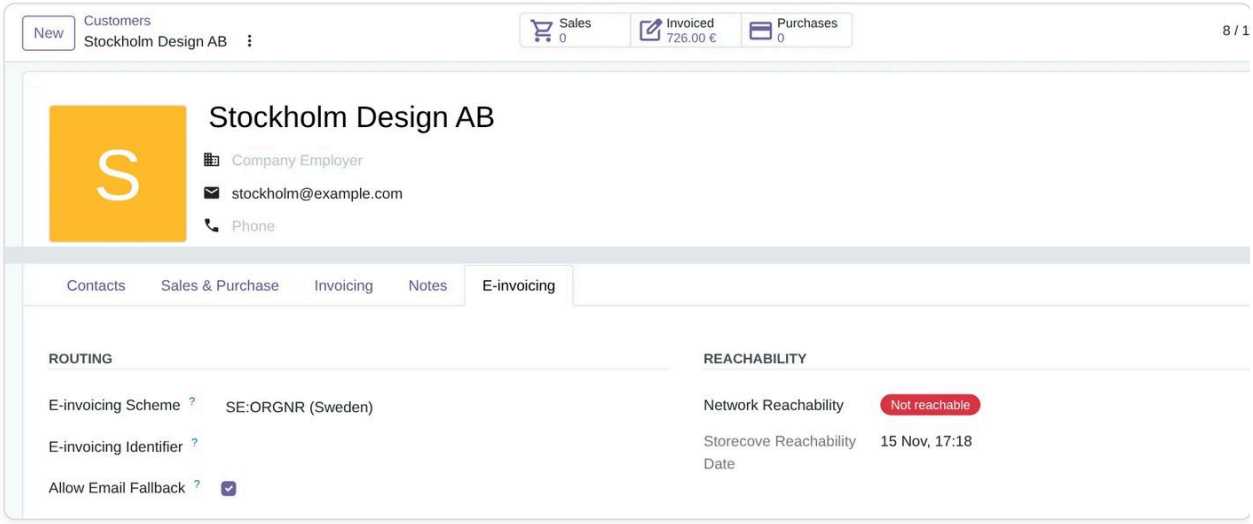
- **Real Odoo 20 screenshot.** Contact with **Network Reachability** at **Reachable**.

2 **Result**

Network Reachability is set to **Reachable** and **Storecove Reachability Date** records the time of the check.

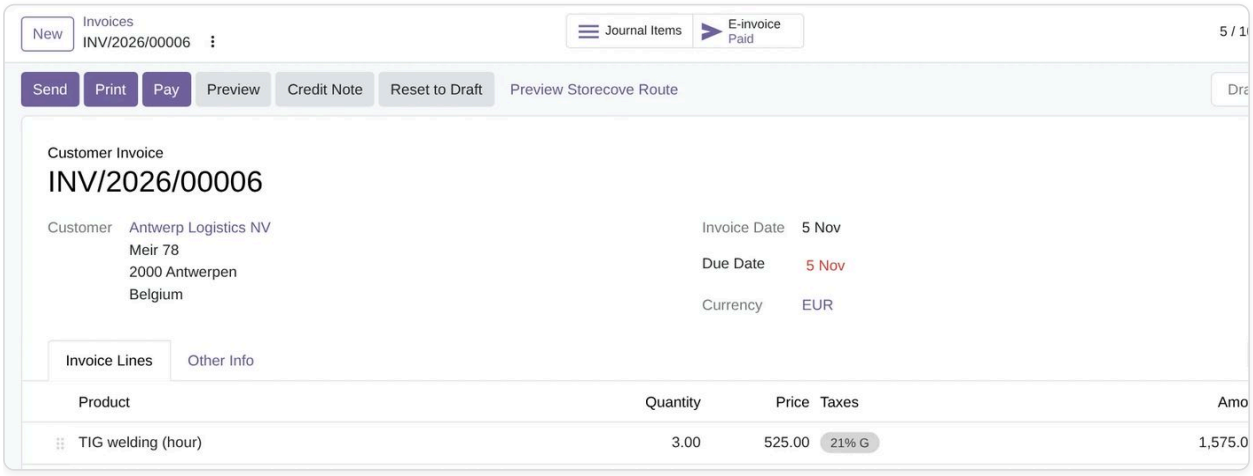
3 **Another contact**

Stockholm Design AB comes out as **Not reachable**, so its invoices could only go by email if you switch on the email fallback.



- **Real Odoo 20 screenshot.** Contact with **Network Reachability** at **Not reachable**.

9 Sending e-invoices and credit notes through Storecove



- **Real Odoo 20 screenshot.** Confirmed invoice with the sending buttons and the e-invoice status.

Where each submission comes from

- When you confirm an invoice, the connector queues it if the account has automatic sending switched on. If something needed for sending is missing, such as the route or the customer's country, the invoice is still confirmed but is not queued, and no warning is shown.
- The **Send via Storecove** button sends it to Storecove immediately. If FlexigoTech Cloud is paused, it stays queued and the task retries it.
- The scheduled task **Storecove: send queued e-invoices** delivers to Storecove whatever is queued, every 5 minutes.
- The task **Storecove: poll webhook queue (pull mode)** collects, every 5 minutes, the notifications from Storecove that update the status.

Invoice buttons

Send via Storecove	Appears on confirmed customer invoices and credit notes that have no active submission.
Self-Bill via Storecove	The same, for confirmed vendor bills and refunds: the invoice is generated by the buyer on behalf of the supplier.
Preview Storecove Route	Only on confirmed customer invoices and credit notes. It is indicative and does not call Storecove: it shows the planned sending route, even if there is no account or legal entity yet.
E-invoice	The button above with the submission status. It opens the submissions for that invoice and only appears if it already has one.

Submission buttons

Send Now	Delivers a queued submission to Storecove without waiting for the task.
Cancel	Cancels a queued submission.

Re-queue	Available on failed, cancelled or **No Route Found** submissions. Prepares a new submission with the current data of the invoice and the contact.
Fetch Evidence	Asks Storecove for proof of transmission. Not shown in **Queued**, **Failed**, **No Route Found** or **Cancelled** states.

States

The status goes from **Queued** to **Sent**, **Delivered** and, if the customer's network allows it, **Acknowledged**, **In Process**, **Under Query**, **Accepted** and **Paid**. It can also end up **Rejected**, **Failed**, **No Route Found** or **Cancelled**.

Steps

1. **Confirm the invoice**

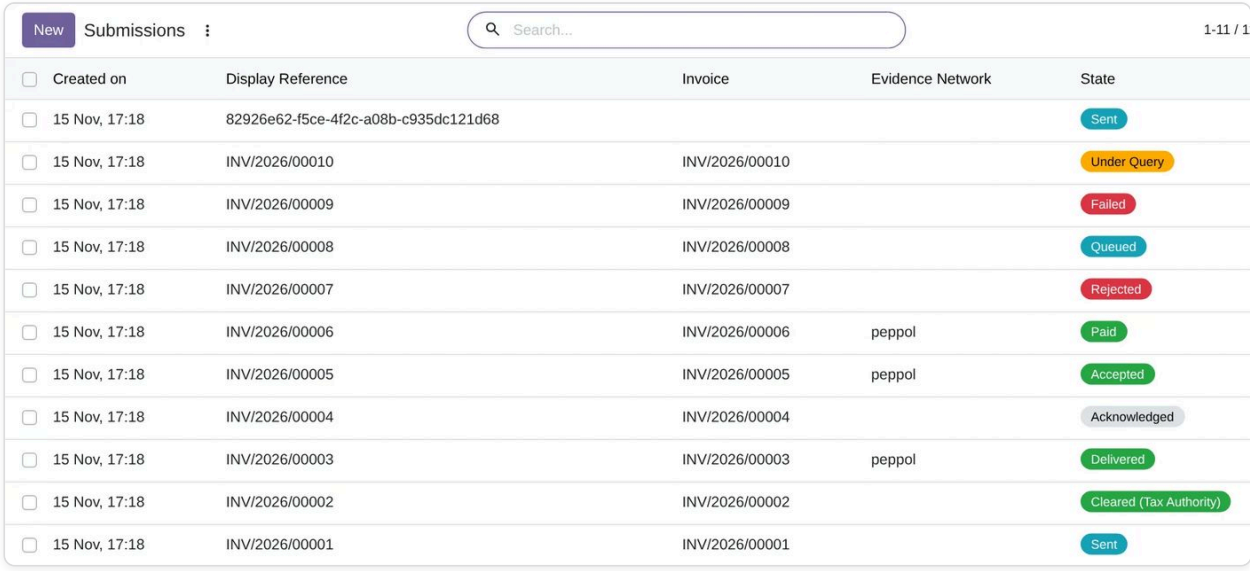
   If automatic sending is active, it is queued by itself.

2. **Press Send via Storecove**

   Only if it was not queued by itself. The invoice goes to Storecove at that moment.

3. **Follow the status**

   Check the **E-invoice** button on the invoice, which shows the submission status, or the **Submissions** list.



- **Real Odoo 20 screenshot.** List of submissions with their status.

> You cannot send a draft invoice, or an invoice that already has an active submission. Only queued submissions can be cancelled: «Only queued submissions can be cancelled; "..." was already handed to Storecove.»

Submission statuses

STATUS	WHAT IT MEANS
Queued	Prepared in Odoo, not yet delivered to Storecove. It can be sent now or cancelled.
Sent	Delivered to Storecove.

STATUS	WHAT IT MEANS
Cleared (Tax Authority)	Cleared by the tax authority.
Delivered	Delivered to the recipient.
Acknowledged	The recipient has acknowledged receipt.
In Process	The recipient is processing it.
Under Query	The recipient has queries about the invoice.
Conditionally Accepted	Accepted with conditions.
Accepted	Accepted by the recipient.
Partially Paid	The recipient has paid part of it.
Paid	The recipient has paid it.
Rejected	The recipient has rejected it.
Failed	The submission has failed.
No Route Found	Storecove could not find a route to send it by.
Cancelled	Cancelled before being delivered to Storecove.

> The statuses from **Acknowledged** to **Paid** only arrive when the recipient's flow supports invoice response.

> If delivery is presumed to be handled by the local network (for example SDI), only clearance evidence exists and no further notices may arrive. This is flagged by the **V-delivery** checkbox.

Rules before sending

The connector checks these conditions when preparing the submission and, if any fails, shows an error message:

- The invoice must be posted and have an invoice date: "Post the invoice before sending it." and "The invoice has no invoice date."
- Each line needs exactly one percentage VAT tax, with no additional fixed rates.
- The customer needs a country.
- The sending company needs a country, and the Storecove legal entity must be registered.
- If the customer has **Advanced Routing (GLN)** enabled, they need their GLN identifier.
- Without a Storecove identifier on the contact and without email fallback enabled on both contact and account, there is no possible route.
- The invoice must have product lines.

Country rules

- Germany: a contact email on the sending company and at least one payment means, that is, a bank account on the invoice.

- Austria, government recipients (AT:GOV): an origin document on the invoice, which serves as the order reference.
- Spain, FACe/FACeB2B: a due date and a bank account on the invoice.
- Spain, Verifactu: the invoice PDF must be generable; if it cannot be generated, the submission stops.
- Switzerland: a Swiss IBAN on the bank account and a valid payment reference. If QR-Reference, the IBAN must be a QR-IBAN; if Creditor Reference, it must start with RF and its check digit must be correct. A PDF is also required.

When it won't let you do something

- **Send Now** and **Cancel** only work on queued submissions: "Only queued submissions can be sent."
- **Re-queue** only applies to failed, cancelled or **No Route Found** submissions; with any other status you get "Only failed or cancelled submissions can be re-queued."
- If an active submission already exists, you cannot create another: "Bill ... already has an active Storecove submission (...)."
- **Self-Bill via Storecove** only accepts vendor bills and their credit notes.

The submission record

Document Type	**Invoice**, **Invoice Response**, **Self-Billed Invoice** or **Order**.
Display Reference	The name under which the submission appears in lists.
Submission GUID	The submission identifier in Storecove.
Sent At	When it was delivered to Storecove.
Status Details	The reason, if the submission failed, and the details Storecove gives about its status.
Evidence Network, **Evidence Transmission**, **Evidence Message**, **Evidence Accesspoint**, **Evidence Qr Code**	Proof of delivery data. Filled in by **Fetch Evidence**; **Evidence Fetched At** shows when.
Raw Parse Strategy	Whether the submission was built from a UBL, CII, IDOC or SETU14 file instead of the invoice data.

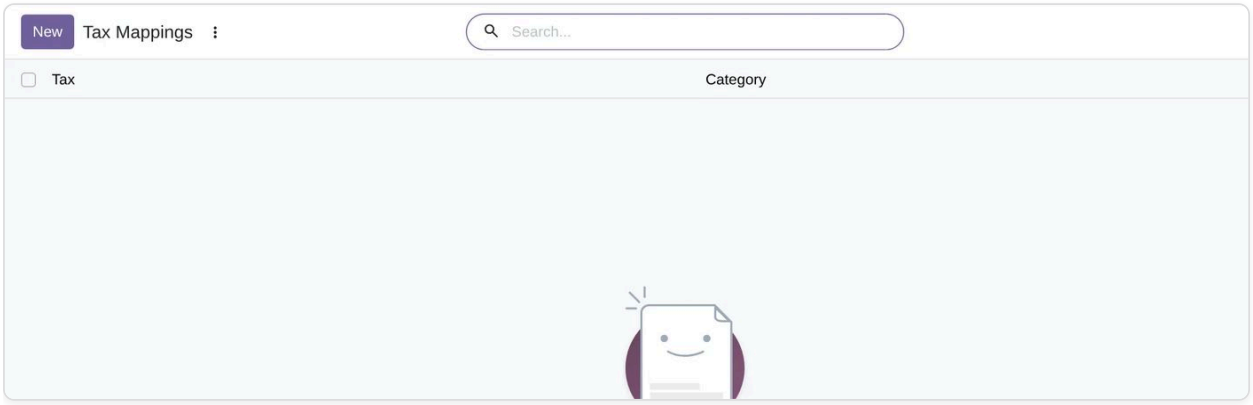
The **Evidence**, **Request Payload** and **Response Document** tabs store what was sent to Storecove and what it replied. The request tab is only visible to the connector's **Manager** role.

End-to-end example

In the sample database, with the account in demo mode, invoice INV/2026/00006 from Antwerp Logistics NV, for €1,905.75, is confirmed and sent. In the **Submissions** list it appears with the peppol network. Its submission moves from **Sent** to **Delivered** and finally to **Paid**. In the same list, the recipient rejected INV/2026/00007, which is in **Rejected**.

10 Tax categories for e-invoicing

Go to **Storecove › Configuration › Tax Mappings**. By default the list is empty and the connector works out each tax's category on its own.



- **Real Odoo 20 screenshot.** The **Tax Mappings** list, still with no overrides.

You create the mappings yourself, one per tax: **New** adds a line to the list itself, which has no separate form.

What you fill in on each mapping

Tax	The Odoo tax whose category you want to force. Required.
Company	The tax's company. Copied automatically from the tax.
Category	The category used for that tax. Defaults to **Standard rate**. The others are **Zero rated**, **Reverse charge**, **Intra-community**, **Exempt**, **Export** and **Outside scope**. Required.

How it is applied

For each tax on an invoice line, the connector first checks whether there is a mapping. If there is, it uses that category. If not, it derives the category from the tax itself.

> A mapping can only force one of the seven categories in the list. The Storecove API has other categories, specific to some countries, such as those for India's tax (IGST, CGST and SGST), and the connector does not offer them.

What happens without a mapping

Without a mapping, the connector derives the category from the tax itself. The **Tax Mappings** screen notes this while the list is empty: "No overrides yet". The possible categories are standard, zero rated, intra-community, export, exempt and reverse charge.

How the category of each line is decided

One category is resolved for each tax on each invoice line. First the per-tax overrides you have created are checked. If the tax has none, the documented derivation rules apply. This lets you leave the connector to derive almost everything and force only the taxes that need it.

The record's data

Tax	Link to the Odoo tax. Required, so a mapping cannot be saved without a tax.
Company	Copied from the tax's company. Used to tell which company each mapping belongs to.
Category	One of seven options: **Standard rate**, **Zero rated**, **Reverse charge**, **Intra-community**, **Exempt**, **Export** or **Outside scope**. If you choose none, it stays as **Standard rate**.

About the categories

The seven are the common categories in the Storecove API. That API has more, specific to some countries, such as IGST, CGST and SGST in India, and the connector does not offer them.

End-to-end example

Invoice INV/2026/00006 from Antwerp Logistics NV has a line for TIG welding (hour) with a 21% tax and a base amount of €1,575.00. If you create no mapping, the connector derives that tax's category. If you want a different one, go to **Storecove › Configuration › Tax Mappings**, click **New**, choose the tax and the **Category**, and save. From then on, lines with that tax use the category you forced.

> Mappings are edited in the list itself, line by line, and have no buttons or statuses.

CONNECTOR GUIDE · STORECOVE

11 Purchase orders via Peppol

The connector also sends purchase orders. Here you are the buyer: the order goes out from your legal entity to the vendor's Storecove identifier, and Storecove delivers it via Peppol. In the **Submissions** list it appears as one more submission, with its status.

What is needed

- A legal entity registered in Storecove for the ordering company, on a connected account.
- The vendor's Storecove identifier, on the **E-invoicing** tab of their company's record. If the order goes to a contact person, the connector looks at the company.
- A confirmed purchase order with at least one product line. Sections and notes are not sent.
- One of the two roles of the connector. Without a role, the button does not appear.

How to send it

1. **Confirm the order and click Send via Storecove**
   The button is in the header of the order. The connector sends the order at that moment and opens the submission it has just created.

2. **Track the status from the order**
   The **E-order** button shows the submission status and takes you to it. When Storecove accepts the order, the submission moves to **Sent** and the order's history keeps the note "E-order handed to Storecove (submission ...)."

While an order has a submission in progress, the **Send via Storecove** button does not appear. It appears again if the submission ends up in **Failed**, **No Route Found**, **Rejected** or **Cancelled**. Automatic sending on confirmation applies only to invoices: orders are always sent using the button.

What data it carries

- The order number and date.
- The delivery date you request, which is the earliest of the planned dates on its lines.
- The supplier reference, if you have entered it on the order, as their quotation reference.
- Each line with its description, quantity, unit, unit price and amount excluding tax. A discount goes as a reduction on its line.
- The total, which is the sum of the lines excluding tax. Taxes do not travel with the order.

The unit travels with its UNECE code (Recommendation 20). The connector recognises a short list of Odoo units: unit, dozen, hour, day, minute, millimetre, centimetre, metre, kilometre, square metre, millilitre, litre, cubic metre, gram, kilogram, tonne and kWh. Any other, such as the inch, the pound or a box you have created yourself, is sent as a generic unit (C62). If you buy by the box, state in the line description how many pieces each one holds.

Before the first submission

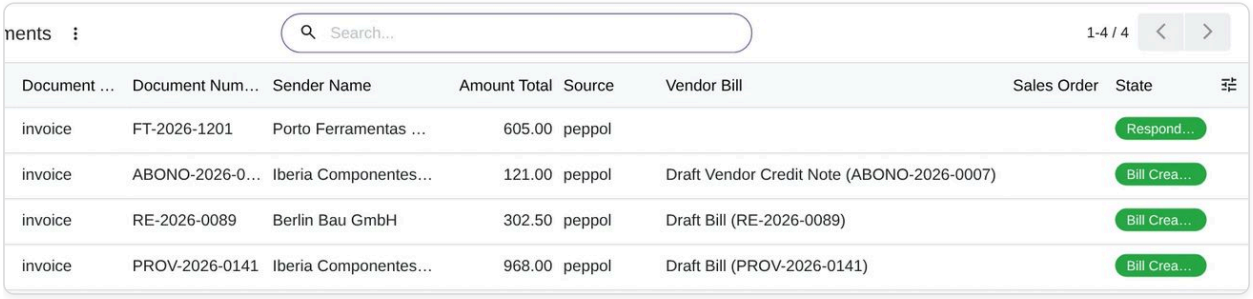
Check Reachability only asks Storecove whether the contact receives invoices via Peppol. There is no check for orders: confirm with the supplier that they receive orders through the network before the first submission. The fallback by email does not work for orders either, and if the supplier has no identifier, the order will not be sent.

If it does not go

- «No route to "...": set a Storecove identifier on the vendor (Accounting tab) before sending a purchase order.»: the vendor's identifier is missing. The message mentions the Accounting tab, but the field is on the **E-invoicing** tab of the vendor's form.
- "The purchase order has no lines to send.": the order contains only sections or notes.
- If the company of the order has no registered legal entity, a warning tells you. Register it in **Storecove › Configuration › Legal Entities**.
- If Storecove does not accept the order, the submission is left in **Failed** with the reason in **Status Details**. Correct the order or the supplier and click **Re-queue**, just as with an invoice.

12 Invoices and orders you receive from your suppliers

Open **Storecove › Documents › Received Documents**. Each row is a document: its type, number, sender, amount, origin, the vendor bill or sales order created from it, and its status.



- **Real Odoo 20 screenshot.** The **Received Documents** list with its statuses.

Where the documents come from

- Storecove leaves a notice in its queue for each new document, and the connector records it.
- The scheduled task **Storecove: poll webhook queue (pull mode)** checks the notice queue every 5 minutes.
- If Storecove cannot deliver a document, the connector requests it again on the next pass. You can request it yourself with **Fetch**.

The buttons on the document

Fetch	Asks Storecove for the content of the document. It appears when the document is in **New** or **Failed** status.
Create Vendor Bill	Creates a draft vendor bill. It appears when the document is **Fetched**, is not an order and does not yet have a bill.
Create Sales Order	Creates the sales order. It appears only for orders that are **Fetched** and do not yet have a sales order.
Accept (AP)	Tells the sender that you accept the invoice. It appears when the document is **Fetched** or **Bill Created**, and it is not an order.
Reject (RE)	Tells the sender that you reject the invoice, under the same conditions as the accept button.

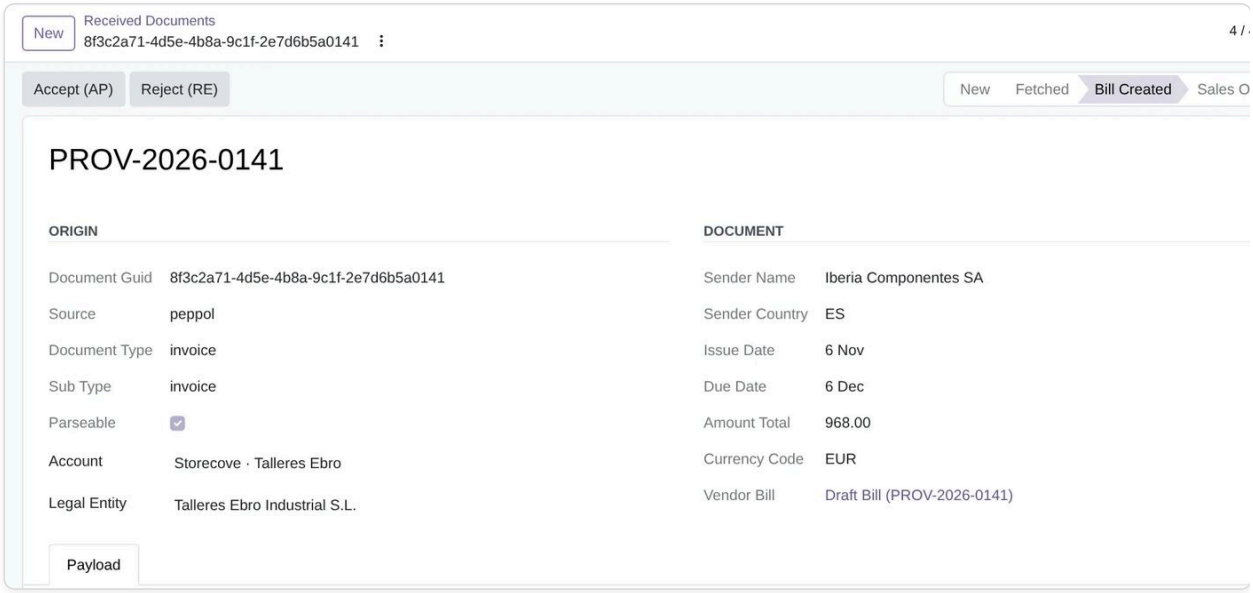
Statuses

- **New**: Storecove has notified the document and it has not yet been fetched.
- **Fetched**: the document is already in your Odoo.
- **Bill Created** and **Sales Order Created**: the draft or the order already exists.
- **Responded**: you have accepted or rejected the document.
- **Failed**: it could not be fetched or processed.

How to use it

1 Open the document

Check the record: **Origin** and **Document** show the sender, the dates and the amount.



- **Real Odoo 20 screenshot.** The record of a received document, with **Accept (AP)** and **Reject (RE)**.

2 Create the bill or the order

Click **Create Vendor Bill** or **Create Sales Order** if it has not been created automatically.

3 Respond to the sender

Click **Accept (AP)** or **Reject (RE)**. The document moves to **Responded**.

> You cannot create a second bill or a second order for the same document. Orders cannot be accepted or rejected. To create the bill or the order, the document must first be fetched.

What each status means

New	Storecove has notified the document and it has not yet been fetched. The connector requests it on the next pass. If Storecove cannot serve it, it stays in this status and is retried.
Fetched	The content is in your Odoo. From here you can create the bill or the order, or respond.
Bill Created	There is a draft vendor bill with the received document.
Sales Order Created	There is a sales order with the received order.
Responded	You have accepted or rejected the document with the sender.
Failed	One day after arrival, a document that Storecove has not been able to serve moves to this status. The reason is stored and an activity is created. **Fetch** tries again.

What it does not allow and why

IF THIS HAPPENS	MESSAGE
The document already has a bill	"A vendor bill already exists for this document."
The document already has an order	"A sales order already exists for this document."
The content has not yet been fetched	"Fetch the document first."
No legal entity matches the one Storecove indicates	"Cannot route this document to a company: no legal entity matches Storecove legal_entity_id."
The company has no purchase journal	«No purchase journal for company "...".»
The received invoice comes without lines	"The received invoice has no lines."
The received order comes without lines	"The received purchase order has no lines."
The customer of an order cannot be found	Odoo warns that it could not identify a customer for the received order: it finds no contact either by identifier or by the sender's name. The customer must first be created or linked, and then you should try again.

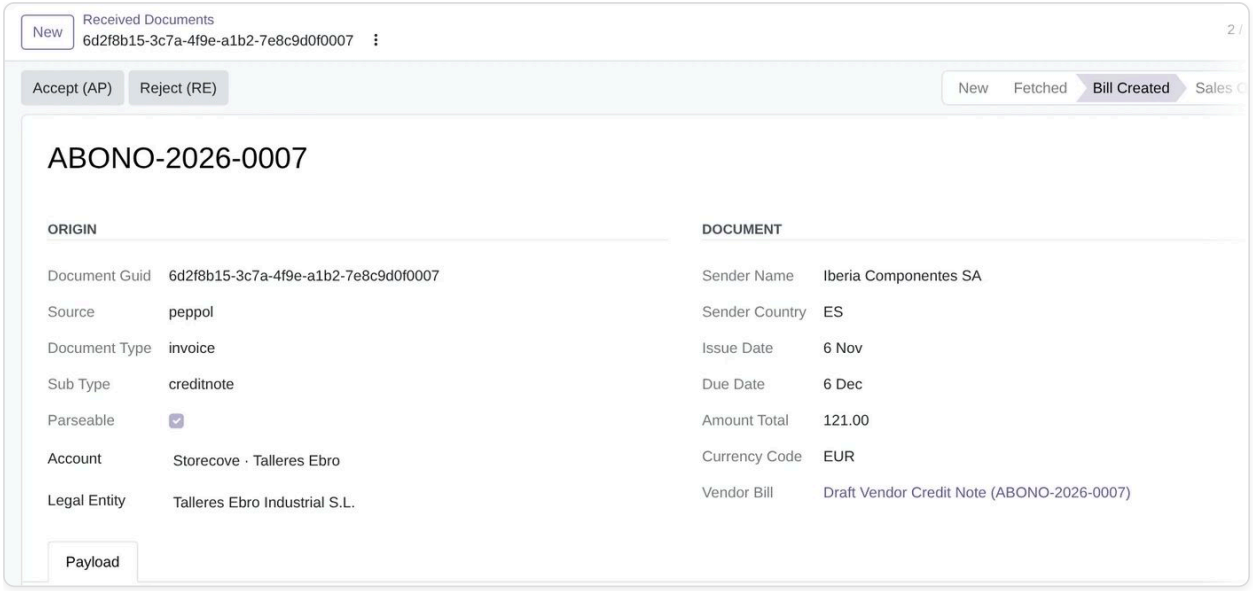
The response to the sender only accepts the codes AB, IP, UQ, CA, RE, AP and PD. The record buttons use AP and RE. If the response fails, a submission of type **Invoice Response** is left in **Failed**, with the reason, and a note on the document, whose status does not change.

The document record

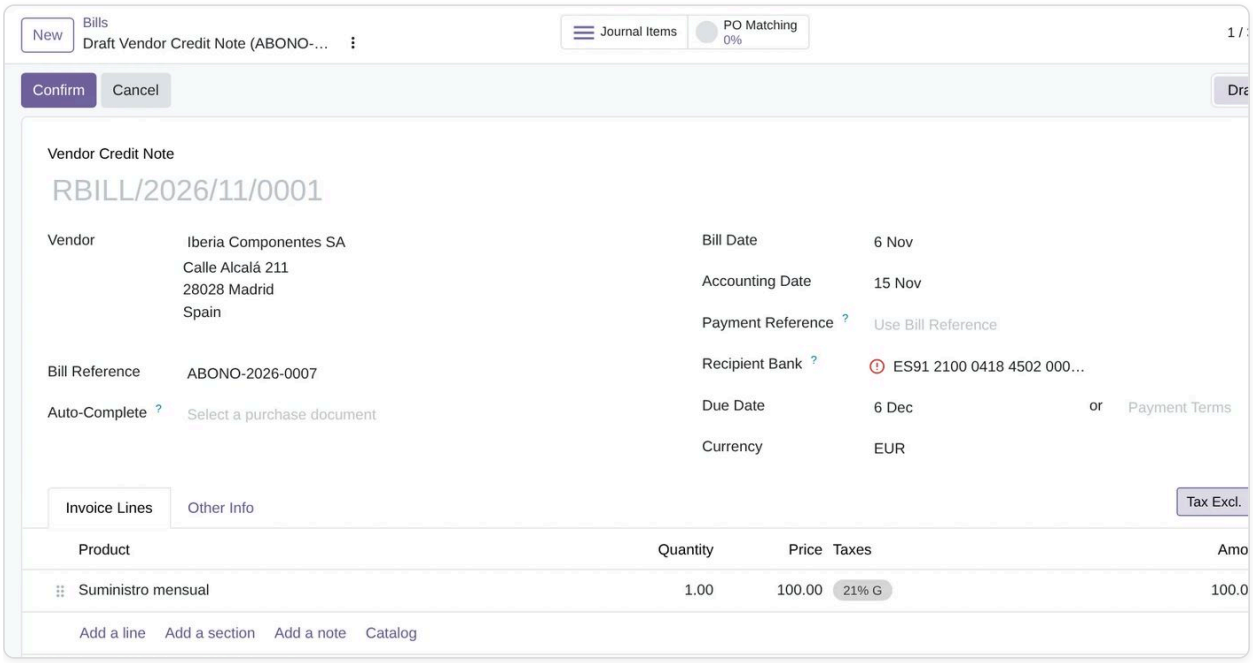
Origin	**Document Guid**, **Origin**, **Document Type**, **Sub Type**, **Parseable**, **Account** and **Legal Entity**.
Document	**Sender Name**, **Sender Country**, **Issue Date**, **Due Date**, **Amount Total**, **Currency Code**, plus the **Vendor Bill** or the **Sales Order** created.
Processing Notes	The notes Storecove leaves about how it processed the document.
Payload	The content of the document as it arrived.

An end-to-end example

Iberia Componentes SA issues a credit note, ABONO-2026-0007, for EUR 121.00. It arrives via Peppol and appears in **Received Documents** with the subtype creditnote. When the draft is created, the record shows "Draft Vendor Credit Note (ABONO-2026-0007)" and the document moves to **Bill Created**. With **Accept (AP)** you respond to the sender and the document is **Responded**.

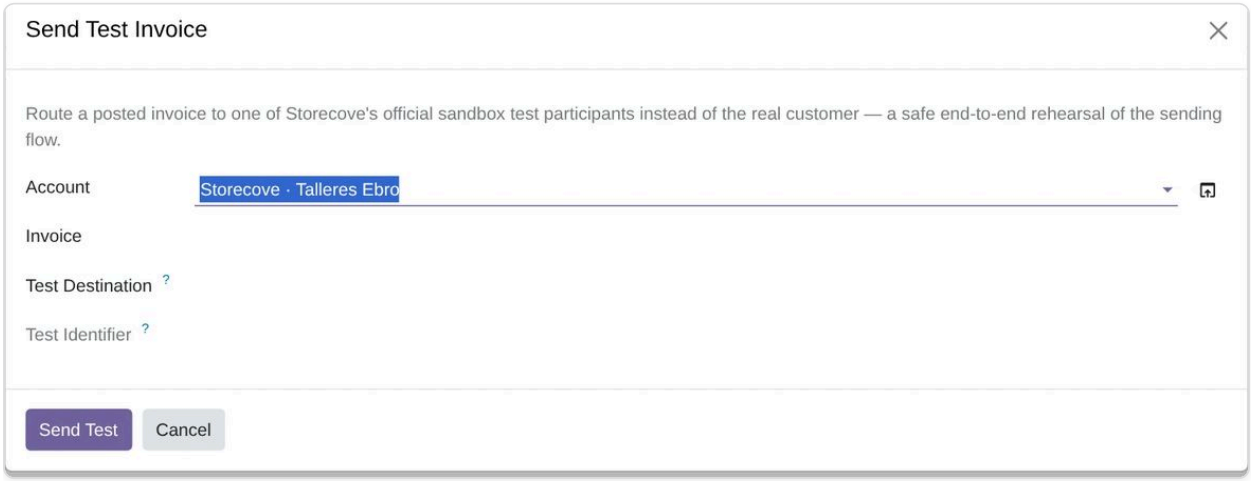


- **Real Odoo 20 screenshot.** Credit note ABONO-2026-0007 with its draft vendor credit note.



- **Real Odoo 20 screenshot.** The draft vendor credit note created from the credit note.

13 Tests and sending raw documents



- **Real Odoo 20 screenshot.** The **Send Test Invoice** wizard with the account, the invoice and the test destination.

Each piece of data comes from what you choose in the wizard: the account, the invoice, the test destination or the file you upload. Both wizards send the document to Storecove immediately, without waiting for the scheduled task, and open the submission they create, which is also recorded in **Storecove › Documents › Submissions**.

Send Test Invoice

Account	The Storecove account from which it is sent. By default the first connected account is selected.
Invoice	The confirmed customer invoice or credit note you want to test.
Test Destination	The official test participant in the sandbox to which the invoice is routed, instead of your real customer.
Test Identifier	The identifier of the destination you chose, as the schema has it.
Send Test	Sends the invoice to the test destination.
Cancel	Closes the wizard without sending anything.

1. **Open the wizard**
   Go to **Storecove › Tools › Send Test Invoice**.
2. **Choose account and invoice**
   Fill in **Account** and **Invoice**.
3. **Choose the destination**
   Select the **Test Destination**; the **Test Identifier** fills in on its own.
4. **Send**
   Click **Send Test**.

Send raw document

Invoice	It is only used to resolve the legal entity and routing. What gets sent comes from the file you upload: Storecove parses it and regenerates the document.
Parse Strategy	The file format: **UBL**, **CII**, **IDOC** or **SETU14 (India e-way-bill)**. It defaults to **UBL**.
Document	The file to be sent.
Send	Sends the document.
Cancel	Closes the wizard without sending anything.

> Test invoices can only be sent from accounts in **Sandbox** or **Demo (offline simulator)**. From an account in **Production** you get the error "Test invoices can only be sent from sandbox or demo accounts - never from production." In addition, the invoice's company needs a legal entity registered on that account.

What each tool does

The test invoice lets you rehearse the sending flow from start to finish without touching your real customer. The invoice is routed to an official test participant in the Storecove sandbox instead of to its recipient. The raw send is for those who already generate their own UBL, CII or IDOC documents: the connector sends the file to Storecove, which parses it and regenerates the document, instead of using the connector's conversion.

Submission states

Both wizards send the document immediately and open the submission they create. The test invoice ends up in **Sent**; if Storecove rejects it, the error is shown and no submission is created. The raw document follows the states of any submission.

Rules and conditions

- A test invoice is not sent from an account in **Production**: you get the error "Test invoices can only be sent from sandbox or demo accounts - never from production."
- If the invoice's company has no legal entity registered on the chosen account, you get «Company "..." has no registered legal entity on account "...".», with the company name and the account name.
- **Account**, **Invoice** and **Test Destination** are mandatory for the test invoice.
- For the raw send, **Invoice**, **Parse Strategy** and **Document** are mandatory.
- The **Invoice** in the raw send only provides the routing context and the legal entity. The content sent is the file.

The wizard fields

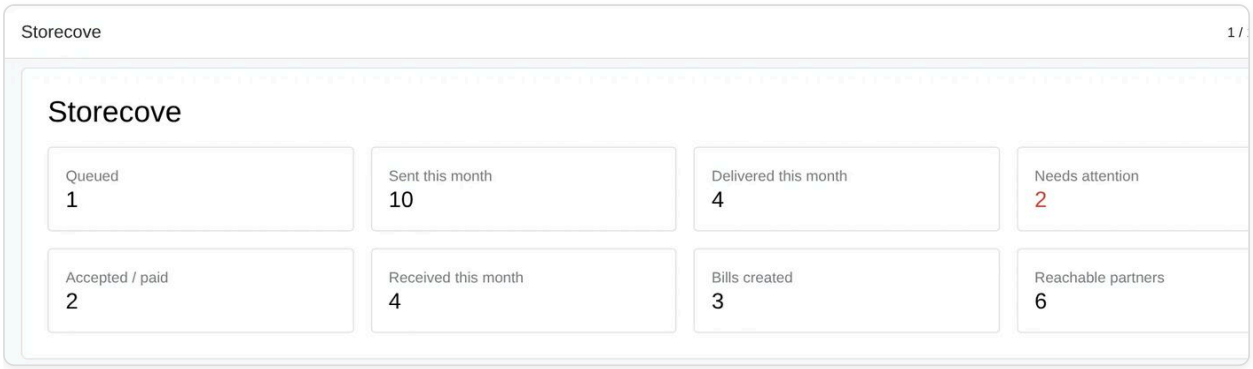
Account	By default, the first account with status connected.
Test Destination	Official test identifier from the Storecove sandbox. The invoice is routed to this participant instead of the real customer.
Test Identifier	Copied from the chosen destination.
Parse Strategy	**UBL**, **CII**, **IDOC** or **SETU14 (India e-way-bill)**.
Document	The file to send.

End-to-end example

You have an account in **Sandbox** and want to rehearse with invoice INV/2026/00006, from Antwerp Logistics NV. You open **Storecove › Tools › Send Test Invoice**, choose the account, the invoice and a **Test Destination**, and click **Send Test**. Odoo sends it to Storecove at that moment and opens the submission, in **Sent**. You also see it in **Storecove › Documents › Submissions**.

CONNECTOR GUIDE · STORECOVE

14 Dashboard and sync log



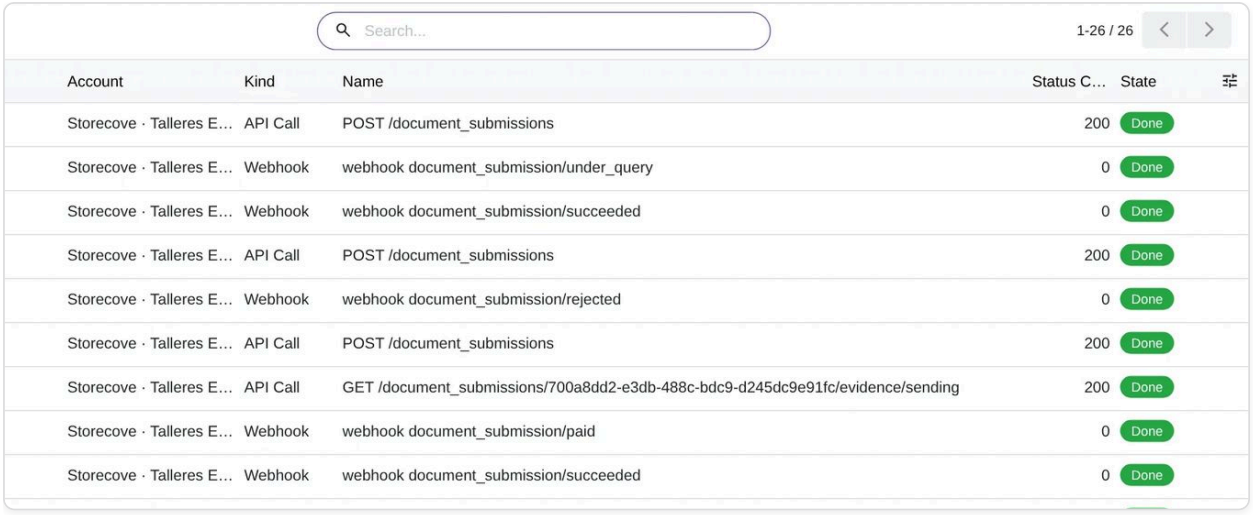
- **Real Odoo 20 screenshot.** The Storecove dashboard with its counters.

You enter the dashboard from **Storecove › Dashboard**. Each tile is a counter that opens the list behind it: **Queued**, **Sent this month**, **Delivered this month**, **Needs Attention**, **Accepted / paid**, **Received this month**, **Bills created** and **Reachable partners**.

While a connected account, a registered legal entity, a provisioned identifier or a reachable contact is missing, the dashboard does not show the counters. Instead, four setup steps appear: **1. Connect your Storecove account**, **2. Register your company as a legal entity**, **3. Provision its Peppol identifiers** and **4. Verify a customer's reachability**. Each one opens the screen where that step is done.

Where the data comes from

- Two scheduled jobs run every 5 minutes: **Storecove: send queued e-invoices**, which takes submissions out of the queue, and **Storecove: poll webhook queue (pull mode)**, which collects notices from Storecove.
- Every API call and every notice received is logged in **Storecove › Configuration › Sync Log**.



- **Real Odoo 20 screenshot.** The sync log with API calls and webhooks.

The sync log

Reprocess Webhook	Reprocesses the stored body of a notice. It only appears on **Webhook** entries in **Error** state, and only for the Storecove administrators group.

Each entry has a state: **Pending**, **Done** or **Error**. In the list you can filter by **Errors**, **Webhooks** and **API Calls**, or group by **State** and **Kind**.

> A notice can only be reprocessed if it is of type **Webhook** and stores its body. Otherwise Odoo shows "Only webhook log entries with a stored body can be reprocessed." If you are not a Storecove manager, it shows "Only a Storecove Manager can reprocess a webhook log entry."

What each tile on the dashboard counts

Queued	All submissions waiting in the queue.
Sent this month and **Delivered this month**	Submissions created this month that have already left for Storecove, and those that have already reached the recipient.
Needs Attention	All submissions in **Failed**, **No Route Found** or **Rejected**.
Accepted / paid	Submissions created this month that are accepted, paid or partly paid.
Received this month and **Bills created**	Documents received this month, and those from this month still in **Bill Created**.
Reachable partners	Contacts that Storecove has confirmed can receive e-invoices.

The four setup buttons follow the order of go-live: connect the account, register the legal entity, provision identifiers and verify a customer. The second and third open the same screen, **Storecove › Configuration › Legal Entities**.

Types and states in the log

API Call	A request Odoo makes to Storecove.

Webhook	A notice Storecove sends to Odoo.
Scheduled Job	A run of a scheduled job.
Pending	The entry has been created and has not finished yet.
Done	The call or notice finished successfully.
Error	The entry failed. If it is a webhook, you can reprocess it.

What you see in each entry

Each entry shows the **Account**, the **Kind**, the **Name**, the **Status Code** and the **State**. The **Request** and **Response** tabs show what was sent and what Storecove replied.

Reprocessing rules

- The **Reprocess Webhook** button is hidden if the entry is not a webhook or if it is not in **Error** state.
- Only Storecove administrators can use it.
- The entry must have the notice body stored. Without it, there is nothing to replay.

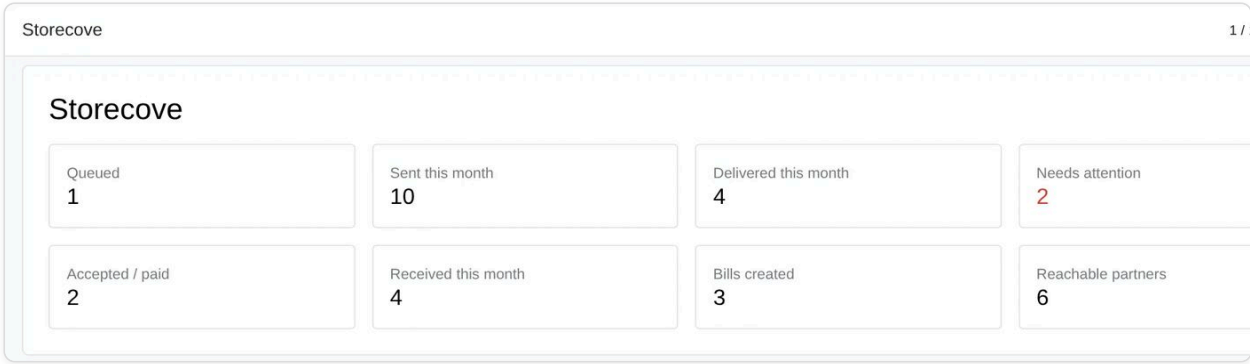
How long it is kept

The log is cleaned up automatically so it does not grow without limit: successful entries older than 90 days are deleted.

An end-to-end example

1 **Open the dashboard**

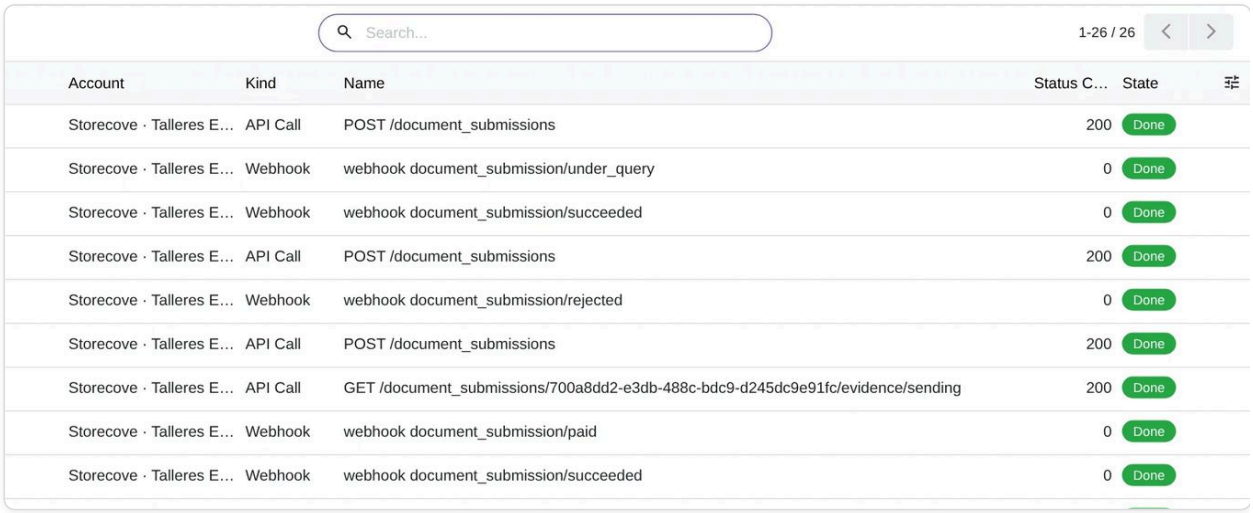
You see how many submissions are queued and how many need attention.



- **Real Odoo 20 screenshot.** The Storecove dashboard with its counters.

2 **Open the log**

Go to **Storecove › Configuration › Sync Log**. You will see the API calls and the notices received, all in **Done** state in this case.



- **Real Odoo 20 screenshot.** The sync log with API calls and webhooks.

3 **Filter the failures**

Click the **Errors** filter to keep only what failed.

4 **Reprocess if appropriate**

If the failure is a **Webhook** in **Error** state, open it and click **Reprocess Webhook**.

CONNECTOR GUIDE · STORECOVE

15 Data going out and coming back

FIELD	DIRECTION	WHEN
Customer e-invoices and credit notes	From Odoo to Storecove	Queued ones go out with **Storecove: send queued e-invoices** every 5 minutes, or straight away with **Send Now**. The **Send via Storecove** button sends them from the invoice.
Vendor bills for self-billing	From Odoo to Storecove	With the **Self-Bill via Storecove** button on vendor bills and vendor credit notes.
Purchase orders	From Odoo to Storecove	With the **Send via Storecove** button on the purchase order.
Raw documents (UBL, CII or IDOC)	From Odoo to Storecove	When you send them from **Storecove › Tools › Send Raw Document (UBL/CII/IDOC)**.
Test invoices	From Odoo to Storecove	With **Send Test**, to the Storecove sandbox test participants.
Legal entities and Peppol identifiers	From Odoo to Storecove	With **Register on Storecove**, **Update on Storecove** and **Provision**; they are withdrawn with **Unregister** and **De-provision**.
Responses to received invoices	From Odoo to Storecove	With **Accept (AP)** and **Reject (RE)** on the received document.
A partner's reachability	From Storecove to Odoo	With **Check Reachability**, on the partner or in the **Storecove › Tools › Check Partner Reachability** wizard, which checks several partners at once.

FIELD	DIRECTION	WHEN
Status of each submission	From Storecove to Odoo	Through Storecove notices, which **Storecove: poll webhook queue (pull mode)** collects every 5 minutes.
Evidence of a submission	From Storecove to Odoo	With **Fetch Evidence** on the submission.
Received documents	From Storecove to Odoo	When their notice arrives. With **Fetch** you retry the download of one that could not be fetched.

Calls to Storecove do not go straight from your Odoo: they go through FlexigoTech Cloud. The account's API key is stored only once, encrypted, in FlexigoTech Cloud, and it never stays in Odoo. FlexigoTech Cloud forwards each of the account's calls to the Storecove API. Polling the notice queue also goes through FlexigoTech Cloud. An account in **Demo (offline simulator)** mode calls nobody. Without a FlexigoTech Cloud subscription, the connector does not talk to Storecove: the sending queue waits, nothing is marked as failed and nothing is deleted.

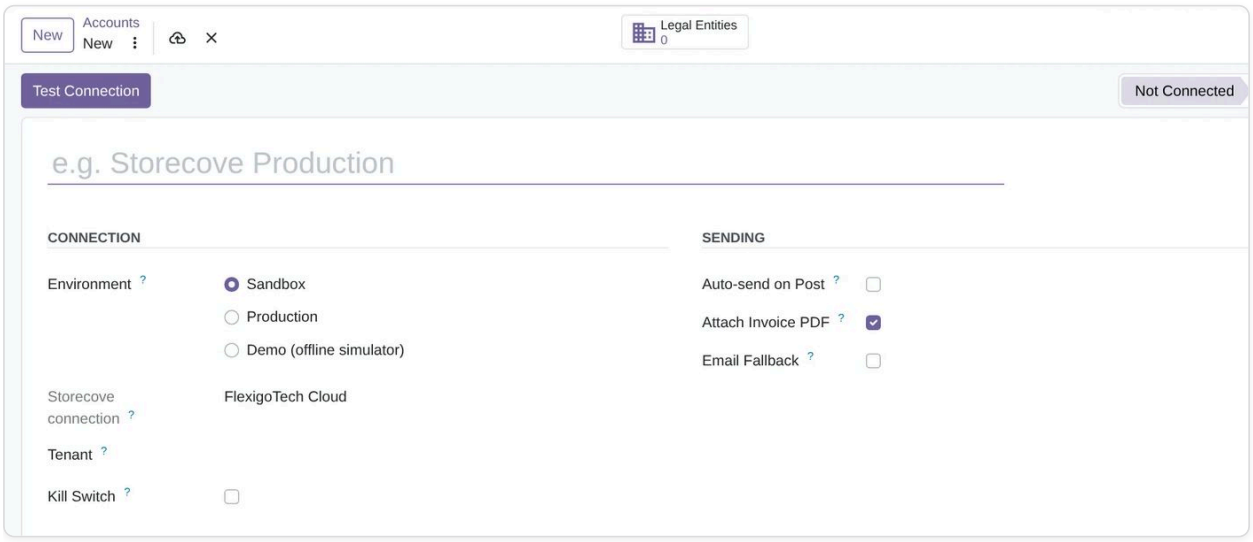
CONNECTOR GUIDE · STORECOVE

16 What happens on its own

TASK	FREQUENCY	WHAT IT DOES
Storecove: send queued e-invoices	Every 5 minutes, enabled	Sends the queued submissions to Storecove.
Storecove: poll webhook queue (pull mode)	Every 5 minutes, enabled	Polls Storecove for pending notices, saves them in the sync log and processes them one by one.

You change the frequencies in **Settings › Technical › Automation › Scheduled Actions**, with developer mode on.

17 The account options and their defaults



- **Real Odoo 20 screenshot.** The form of a new account, with the **Connection** and **Sending** groups.

OPTION	ON CREATION	WHAT IT DOES
Name	Empty, required	The name you use to recognise the account in lists.
Environment	**Sandbox**	The other options are **Production** and **Demo (offline simulator)**. Sandbox and production are separate Storecove accounts, each with its own keys. Demo does not contact Storecove and simulates responses.
Storecove connection	**FlexigoTech Cloud**	Read-only, it cannot be changed: every account you create connects through FlexigoTech Cloud.
Tenant	Empty	Optional identifier passed to the Storecove legal entities. Storecove returns it in each notice, to tell tenants apart.
Kill Switch	Off	Suspends all outgoing traffic to Storecove for this account. Queued documents stay queued and nothing is sent or polled.
Main Company	The active company	The company the account belongs to. It is required.
Auto-send on Post	Off	When on, every customer invoice or credit note you post, from a company with a connected legal entity, goes into the sending queue automatically.
Attach Invoice PDF	On	Embeds the Odoo invoice PDF in the e-invoice as the main attachment.
Email Fallback	Off	If the recipient has no reachable electronic identifier, Storecove delivers the invoice by email to the contact's billing address.

You do not type the API key into Odoo. It is stored encrypted in FlexigoTech Cloud, from the **FlexigoTech Cloud (requires FlexigoTech Cloud subscription)** section of the account, and only an administrator can do it. If you try to save it any other way, Odoo blocks you with a message. An account's connection type also cannot be changed after it is created.

What you do by hand

These account actions do not happen on their own: you trigger them with a button.

TAB OR GROUP	BUTTONS
Account header	**Test Connection**
FlexigoTech Cloud (requires FlexigoTech Cloud subscription)	**Check FlexigoTech Cloud** shows since when the key has been stored. **Delete from FlexigoTech Cloud** deletes it. Both only appear once the key is stored, and only an Odoo administrator can see the whole group.
Receiving	**Regenerate Push Webhook Token**, only for the connector's **Manager** role

The **FlexigoTech Cloud (requires FlexigoTech Cloud subscription)** group holds the account's API key. The key is sent once, encrypted, to FlexigoTech Cloud and is never kept in Odoo. Before saving it, a manager has to tick the checkbox accepting that the key and the calls to Storecove go through FlexigoTech Cloud. The **Saved in FlexigoTech Cloud** field shows when it was saved. If the key is not saved yet, nothing is sent and Odoo warns you to save it first.

The **Receiving** group shows **Last Pull At**, the date when Odoo last fetched documents from Storecove.

18 Security and personal data

- The Storecove API key is not stored in your Odoo. The key goes to FlexigoTech Cloud once, encrypted, and stays there.
- If you try to save the key on a new account by any other route, Odoo rejects it with the message "A new Storecove account keeps its API key in FlexigoTech Cloud, never in Odoo: save it in the FlexigoTech Cloud section of the account.".
- The person who gives consent is an Odoo manager. They tick the **I accept that my Storecove API key and my calls to Storecove go through FlexigoTech Cloud** checkbox on the account. Without that acceptance, the key is not sent and you get the warning "Before saving the API key, an administrator has to accept that the key and the calls to Storecove go through FlexigoTech Cloud. Nothing was sent.".
- The **API key (FlexigoTech Cloud)** field, the consent checkbox and **Saved in FlexigoTech Cloud** are visible only to the manager. Deleting the key from FlexigoTech Cloud with **Delete from FlexigoTech Cloud** is also a manager task, and if another user tries it, they get "Only an administrator can delete the Storecove API key from FlexigoTech Cloud.".
- The invoice and order buttons (**Send via Storecove**, **Self-Bill via Storecove**, **Preview Storecove Route**) are visible to the connector's **User** and **Manager** roles. **Regenerate Push Webhook Token** and **Reprocess Webhook** are only for the **Manager** role.
- Each account's calls to Storecove go through FlexigoTech Cloud: the invoices and orders you send with their PDF, the legal entities and Peppol identifiers, reachability checks, evidence and received documents.
- Your customers, orders, invoices and products are stored in your Odoo. FlexigoTech Cloud passes on to Storecove what each call carries and keeps it only for as long as needed to deliver it.
- If you delete an account that had the key saved, the key is also removed from FlexigoTech Cloud.
- With a paused subscription or without permission, the connector does not call Storecove: the queue waits and nothing is marked as failed or deleted.

Personal data and deletion

FlexigoTech Cloud keeps the encrypted API key on servers in Germany. In your Odoo, the connector does not delete invoices, contacts or received documents. From the sync log it only deletes, once a day, old successful entries.

To set up the subscription, the company name and the email of the person who accepts are stored, together with the date and the version of the text accepted.

19 Multiple accounts and multiple companies

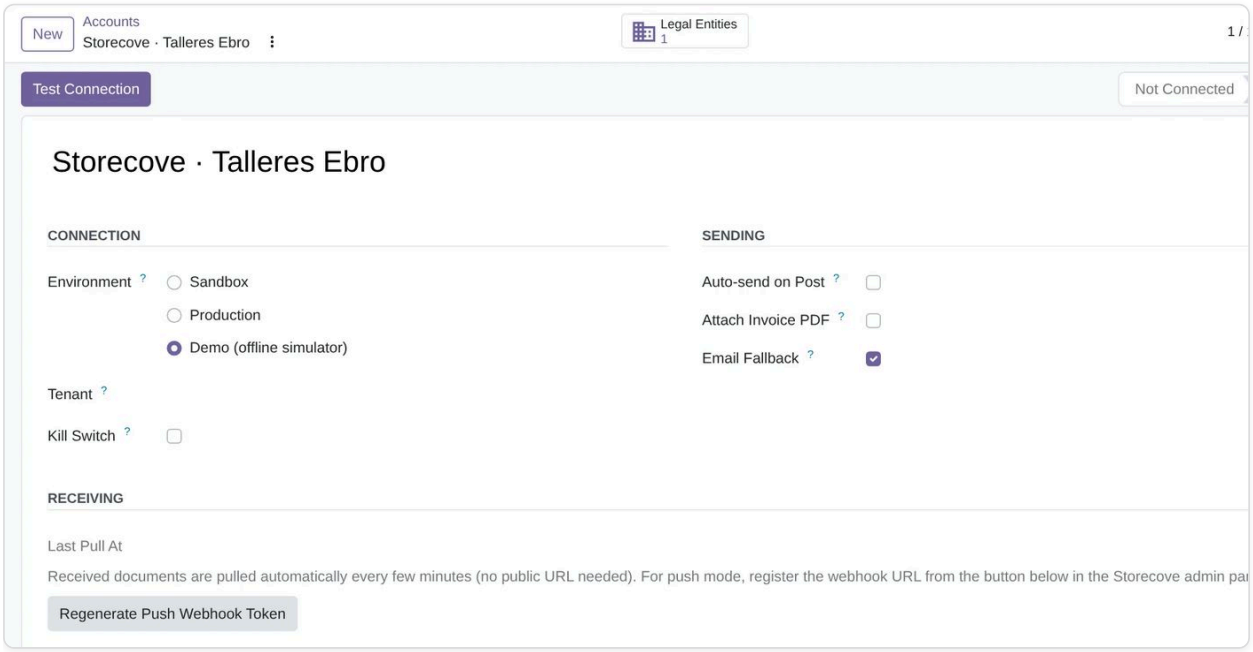
Each account has its own options: environment (sandbox, production or demo), automatic sending, attaching the PDF and email fallback. Sandbox and production are separate accounts, each with its own key, and an account never calls with another account's key.

The Standard plan supports up to 3 accounts per connector and the Pro plan up to 10. The first support response arrives within 1 working day with Standard and within 4 working hours with Pro.

Each legal entity, submission, received document and sync log entry belongs to a company, and a user sees only those of their own companies. Accounts and the scheme catalogue are shared across the whole database,

and each company has at most one legal entity per account. When an account serves several companies, each call is logged against the company of the legal entity, submission or received document that originates it, not that of the account.

If a company has legal entities registered in more than one Storecove account, the connector does not choose on its own and warns you so that you can deactivate the one you do not use. If a company has no legal entity in the account, you get «Company "..." has no registered legal entity on account "...".».



- **Real Odoo 20 screenshot.** One Storecove account with its environment, sending and receiving.

CONNECTOR GUIDE · STORECOVE

20 If something fails

The key is not saved in the account	Odoo does not store the API key on the account. You enter it in the FlexigoTech Cloud section of the account, after ticking **I accept that my Storecove API key and my calls to Storecove go through FlexigoTech Cloud**. If it is not saved, you get "Save the API key of ... in FlexigoTech Cloud first: nothing was sent."
There is no FlexigoTech Cloud service	Without a trial or subscription, the connector sends nothing. You get "Storecove has no FlexigoTech Cloud service on this database: nothing was sent, the queue waits and nothing is marked as failed or deleted. See Settings > FlexigoTech Cloud." Activate the trial or the subscription and the queue carries on by itself.
Only an administrator can do this	Accepting the consent and saving the key are administrator tasks. Other users receive a permissions warning.
The invoice does not reach Storecove	Check its submission in **Submissions**: the invoice may have an active submission, the customer's country may be missing, or a line may have more than one tax. If it was confirmed with automatic sending and has no submission, something was missing when it was confirmed: fix it and press **Send via Storecove**.
A received document does not create the supplier bill	Open the document in **Received Documents** and check its status. If a bill has already been created, another one will not be created. If the purchase journal or the legal entity is missing, they need to be set up.
You want to see what Storecove replied	Look at the **Sync Log**: every call and every webhook is recorded with its status code.

If FlexigoTech Cloud pauses, the submission waits. Nothing is marked as failed or deleted, and the task **Storecove: send queued e-invoices** retries it every 5 minutes. A received document that Storecove cannot serve is left as **New** and requested again on the next run. If a day has passed since it arrived, it is marked as failed and you can try again with the fetch button.

Other messages you may see

"Bill ... already has an active Storecove submission (...)."	That invoice is already in progress. Wait for it to finish or check its status in **Submissions**.
"Post the invoice before sending it."	Only confirmed invoices are sent. Confirm it and send again.
«No route to "...": set a Storecove identifier on the partner (Accounting tab) or enable the email fallback on both the partner and the account.»	Complete the contact's identifier or enable the email fallback on the contact and the account. The message refers to the Accounting tab, but those fields are on the contact's **E-invoicing** tab.
«Customer "..." has no country; it is required for e-invoicing.»	Set the country on the customer record.
«Sending via FACe/FACeB2B (Spain) requires a due date on invoice "..." (docs §6.1 ES: "mandatory dueDate").»	Set the due date on the invoice.
«Only queued submissions can be cancelled; "..." was already handed to Storecove.»	A submission that has already gone cannot be cancelled.
«Register the legal entity "..." on Storecove first.»	Register the legal entity in Storecove before sending or responding to a document.
"Select at least one partner."	Choose a contact in the **Check Partner Reachability** wizard before pressing **Check**.

CONNECTOR GUIDE · STOREOVE

21 Other ways of doing this

METHOD	WHAT IT INVOLVES
Manually from the Storecove dashboard	Someone prepares each document in the Storecove dashboard, outside Odoo. The invoice details have to be re-entered and what was sent has to be noted by hand.
Odoo alone	Odoo does not include anything for Storecove. Its Peppol module, in Community from 17 to 20, registers your company as a Peppol participant and sends and receives in Peppol BIS Billing 3.0, only over Peppol and only for the countries on its list.
This connector	It takes customer invoices and credit notes from Odoo to Storecove and brings in the documents sent to you. It needs your own Storecove account and a FlexigoTech Cloud subscription, from €49 a month excluding VAT, with a 14-day trial and no card required.

If you are coming from another tool, there is nothing to import: automatic sending only acts on invoices you confirm after switching it on, and you send the earlier ones yourself with **Send via Storecove**.

CONNECTOR GUIDE · STORECOVE

22 Plans and pricing

€9

License per Odoo version.

+ FlexigoTech Cloud

STANDARD

€49/month

or €490 a year

- First response within 1 working day
- Up to 3 accounts

PRO

€98/month

or €980 a year

- First response within 4 working hours
- Up to 10 accounts
- Video call if needed

14-day trial, no card **Prices exclude VAT**

The module license is per Odoo version. FlexigoTech Cloud is paid monthly or yearly per production database, and the 14-day trial needs no card. The subscription also covers the test and pre-production copies of that database. With Standard, up to 3 accounts; with Pro, up to 10.

23 Frequently asked questions

Does it work without FlexigoTech Cloud?

No.

Does it work on Odoo Online?

No. Odoo Online does not allow third-party modules to be installed. It does work on Odoo.sh or on your own server.

Which Odoo versions are supported?

Odoo 17, 18, 19 and 20, with a branch of the module for each.

How has it been tested?

With automated tests on Odoo 17, 18, 19 and 20 that simulate Storecove's responses according to its documentation. An earlier version of the connector sent a real invoice and a real order through Storecove's sandbox; with this one, the pass through FlexigoTech Cloud has been tested with simulated responses.

Does it need Odoo Enterprise?

No. It works with Odoo Community: it only depends on Invoicing, Purchase, Sales and FlexigoTech Cloud.

Do I need a Storecove account?

Yes. The account and its API key are yours, and Storecove's fees are not included in the connector.

How much does the connector cost?

The module license costs €9 on Odoo Apps, per Odoo version. The FlexigoTech Cloud subscription is separate: the Standard plan costs €49 a month or €490 a year, and the Pro plan €98 a month or €980 a year. Prices exclude VAT. There is a 14-day trial with no card required.

Is there a free trial?

Yes, 14 days with no card required. If you do not subscribe at the end, the service stops and nothing is charged.

How many Storecove accounts does each plan allow?

The Standard plan allows up to 3 accounts per connector and the Pro plan up to 10.

What is the difference in support between plans?

The first reply arrives within 1 working day with Standard and within 4 working hours with Pro, which includes a video call if needed.

Where is the Storecove API key stored?

Encrypted in FlexigoTech Cloud, never in your Odoo. The key is entered once from the FlexigoTech Cloud section of the account.

Who can save the key?

Only an Odoo administrator, who must first accept that the key and the calls to Storecove go through FlexigoTech Cloud.

What documents can I send?

Customer invoices and credit notes, supplier invoices for self-billing and purchase orders, as well as raw documents in UBL, CII or IDOC.

Which networks does it reach?

Peppol and other networks Storecove delivers through, such as SDI, KSeF or FACeB2B. The network depends on the country and the customer's identifier.

What about documents sent to me?

They appear in **Received Documents**. With the legal entity's automatic creation tick boxes switched on, as they come by default, a draft supplier bill or a sales order is created, and someone has to approve them in Odoo.

How often are queued invoices sent?

Every 5 minutes, with the task **Storecove: send queued e-invoices**, which is switched on by default. With **Send Now** you can send a queued submission straight away.

How often do documents and Storecove notifications arrive?

Every 5 minutes, with the task **Storecove: poll webhook queue (pull mode)**, which is switched on by default and needs no public URL.

Can I receive notifications instantly instead of polling?

The code has a push mode, but the account form does not show its URL, so out of the box notifications are collected every 5 minutes.

Can the frequencies be changed?

Yes, in **Settings › Technical › Automation › Scheduled Actions**, with developer mode enabled.

How many scheduled tasks does it include?

Two, both active on installation.

Can I test without sending anything to a real customer?

Yes. **Send Test Invoice** immediately sends a confirmed invoice to a test participant in the sandbox, instead of to your customer. It only works from accounts on **Sandbox** or **Demo (offline simulator)**, never from **Production**.

Is there an offline demo mode?

Yes. The **Demo (offline simulator)** environment never contacts Storecove and simulates the responses.

Can I check before sending whether a customer receives over the network?

Yes. **Check Reachability** on the contact record, or the **Check Partner Reachability** tool for several at once, asks Storecove and stores the result and the date.

Is it updated automatically if a customer becomes reachable?

No. There is no scheduled task that updates it: reachability and its date are written when you press the button.

What happens to invoices I had before installing it?

The connector does not import history or migrate data in your database. Confirmed invoices that already existed are not sent automatically: you queue them yourself with **Send via Storecove**.

Can I have several Odoo companies?

Yes. Each legal entity, submission, received document and log entry belongs to a company, and a user only sees those of their companies.

What is a legal entity?

It is the Odoo company registered with Storecove. Outgoing invoices from that company are sent through it.

What happens if FlexigoTech Cloud is unavailable?

The submission waits: nothing is marked as failed or deleted, and the task retries it every 5 minutes.

What happens if the trial or subscription expires?

The connector stops talking to Storecove. The send queue waits, nothing is marked as failed and nothing is deleted.

Which taxes are supported on invoices?

Each line needs exactly one percentage VAT tax, with no additional fixed charges. The tax category is derived from the tax and you can override it in **Tax Mappings**.

How many identifier schemes does it include?

The catalogue is loaded on installation with 86 schemes.

Does it support fiscal reporting for invoices, such as Verifactu or KSeF?

If the legal entity's country has a documented reporting network that Storecove does not use by default, **Force CTC Reporting** requests it on all issued invoices: es-aeat-verifactu, pl-ksef, ro-anaf or my-lhdnm. In Poland and Romania, Storecove has to enable it first.

How long is personal data kept?

The connector doesn't delete anything from your Odoo, except the old correct entries in the sync log. FlexigoTech Cloud passes each call to Storecove and keeps it only as long as needed to deliver it.

Does it include custom configuration or data migration?

No. It does not include custom developments, setting up the module for you, importing historical data or migrating your installation (your data and your customizations) to another version of Odoo. Help with configuration and fixing errors is included, and the subscription also covers test and pre-production copies of that database.

How big is the connector?

It brings 15 menus of its own and works with 19 data models, either its own or extended.

24 Glossary

Storecove	Electronic invoicing platform and Peppol access point. It receives your invoice, converts it to the format each destination network requires and delivers it.
FlexigoTech Cloud	FlexigoTech service that stores each Storecove account's API key encrypted and passes each call from the connector to Storecove. Requires a subscription, with a 14-day trial and no card needed.
Account	Each connection between Odoo and Storecove, with its environment, sending options and API key. They are created under **Storecove › Configuration › Accounts**.
API key	Credential for your Storecove account. It is provided once, encrypted, to FlexigoTech Cloud and is not stored in Odoo.
Sandbox	Storecove's test environment, with its own keys. It is a separate account from production.
Production	Storecove's live environment, with its own keys.
Demo (offline simulator)	Account environment that simulates responses and never contacts Storecove.
Legal entity	Odoo company registered with Storecove. Outgoing invoices from that company are sent through it.
Identifier	Data used to locate a company on an exchange network, for example the tax ID or a Peppol identifier.
Identifier scheme	Type of identifier for a country, such as ES:VAT or NL:KVK. The catalogue is loaded at installation.
Provision	Register an identifier of the entity on the exchange network so it can receive documents.
Peppol	International network for exchanging electronic invoices, one of the networks the connector reaches.
Destination network	Network a document travels through, such as Peppol, SDI, KSeF or FACeB2B.
Reachability	Status showing whether Storecove finds a contact on any network: **Not checked**, **Reachable** or **Not reachable**.
Submission	Record of each document you send to Storecove, with its status. It is viewed under **Storecove › Documents › Submissions**.
Evidence	Proof of transmission of a submission. Requested with **Fetch Evidence**.
Received document	Document sent to you by a third party that arrives in Odoo. It is viewed under **Storecove › Documents › Received Documents**.
Self-billing invoice	Supplier invoice generated by the buyer on behalf of the supplier, with **Self-Bill via Storecove**.
Webhook	Notification Storecove sends to your Odoo when something changes, such as the status of a submission or the arrival of a document.
Pull and push mode	In pull mode, Odoo queries Storecove for pending notifications every 5 minutes. In push mode, Storecove sends them to a URL on your Odoo, which the account form does not show: by default the connector works in pull.

Tax mapping	Rule that forces the tax category of a specific tax. Without a mapping, the connector deduces it from the tax itself.
CTC / eReporting	Documented tax reporting network of a country, such as es-aeat-verifactu. It is enabled with **Force CTC Reporting**.

CONNECTOR GUIDE · STORECOVE

25 Checklist and contact

- [ ] You have your own Storecove account and its API key, in sandbox or production.
- [ ] Your Odoo is version 17, 18, 19 or 20, on Odoo.sh or on your own server, not on Odoo Online.
- [ ] The Invoicing, Purchase and Sales apps are installed.
- [ ] You have installed the module and can see the Storecove app.
- [ ] An administrator has activated the 14-day FlexigoTech Cloud trial or the subscription.
- [ ] You have created the account under **Storecove › Configuration › Accounts**, with the correct environment, and saved the API key in the FlexigoTech Cloud section.
- [ ] You have pressed **Test Connection** and the account shows as **Connected**.
- [ ] You have reviewed the account's automatic sending, attaching the PDF and the email fallback.
- [ ] You have registered your company as a legal entity with **Register on Storecove** and it has moved to **Registered**.
- [ ] You have checked a contact's reachability with **Check Reachability**.
- [ ] You have sent a test invoice from sandbox and can see its status under **Submissions**.

If you would like us to set it up with you, write to us. With the Pro plan, if needed, we can go through it by video call.

Email

To install it, set it up or for any question.
comercial@flexigotech.com

On the web

The connector page, with the guide and the prices.
flexigotech.com/en/storecove-connector-odoo.html

On Odoo Apps

The module page on the Odoo store.
apps.odoo.com

Storecove is a trademark of its owner. FlexigoTech publishes this connector on its own, with no endorsement or sponsorship from the trademark owner. FlexigoTech is a brand of Flexibles y Accesorios Gobe, S.L. (Barcelona, Spain).