d.velop smart invoice API

Table of Contents

1. d.velop smart invoice API

1.1. Basic information

1.1.1. About the d.velop smart invoice API

This documentation is available online to development partners of d.velop AG in the Service Portal. Passing this documentation or parts of it on to someone else is not permitted. For requests as part of the development partnership, only the online documentation applies.

Please note that via this interface your software also accesses data that has been stored and configured in d.velop smart invoice by your customers and also affects the processes in the d.velop smart invoice system. Please use it with care and make sure that your application is part of an existing collaboration of several applications. The improper use of this interface may lead to changed application procedures and the loss of data.

For any questions about the prerequisites and about software development with d.velop smart invoice, please contact d.velop AG Technology Partner Management.

1.1.2. Scope of functions of d.velop smart invoice

d.velop smart invoice is the central component in the (automatic) verification and approval of incoming invoices.

1.1.3. Required knowledge

This documentation is intended for d.velop invoices system administrators and developers of apps and enhancements.

You should be familiar with the following subjects to use the API:

- They should have a good understanding of the d.velop invoices architecture.
- They should be familiar with authentication via the d.ecs identity provider API in the context of d.velop invoices.
- You should have knowledge in the development of web-based apps and be confident in dealing with the following detailed topics:
  - Hypertext Transfer Protocol (HTTP) (RFC 7230)
  - RESTful HTTP interfaces
  - JavaScript Object Notation JSON) (RFC 7159)
  - Hypertext Application Language in connection with the JavaScript Object Notation (HAL+JSON)
- Additionally, knowledge in administering a web server is a benefit.

1.2. Notes about support for API programming

Please note that via this interface your software also accesses data that has been stored and configured in d.velop invoices by your customers and also affects the processes in d.velop invoices. Please use it with care and make sure that your application is part of an existing collaboration of several applications. The improper use of this interface can lead to changed application procedures and the loss of data.

Software developments with this programming interface are individual developments. The program code generated by you is not subject to the maintenance and support conditions of the products of d.velop AG. Our support team is happy to help you, but your requests are subject to additional charges if the issues are not caused by an error in our products.

For any questions about the prerequisites and about software development with d.velop invoices, please contact d.velop AG Technology Partner Management.

1.3. Using the API functions

Below you can learn about the various options for using the d.velop smart invoice API user interfaces to meet your requirements.

1.3.1. Receiving document data (d.velop smart invoice)

You can receive document data from d.velop invoices for further processing in two ways: via webhook or via the API for retrieving outstanding transfers. If you choose to process the data via a webhook, d.velop invoices calls your implemented web service with the data of the document to be transferred. If you process the data by retrieving the transfers, your application is responsible for the regular retrieval of the outstanding documents within a configurable time frame, processing them and reporting whether or not processing was successful to d.velop invoices.

When is a webhook the better option and when is retrieval via API?

The retrieval of data via API and receipt of the data via webhook basically offer the same scope of functions. You get the same document data with both variants, and both variants require you to report whether or not processing was completed. So when should you choose to retrieve the data via API and when should you choose to receive it via webhook?

A webhook lets you respond to document data more quickly, because d.velop invoices actively contacts your application as soon as there is data to be processed. You can immediately check the data for validity and provide a reply, for instance. If you retrieve the data via API, your application checks whether new data is available at regular intervals. If this check is performed once an hour, for example, your application only transfers data to the external system once an hour as well. If errors occur and you report the process to be unsuccessful, the users only find out about this with this delay too.

To retrieve document data via API, your application must be able to access the d.velop invoices API. If you are using the cloud version of d.velop invoices, your application must therefore have internet access. To ensure that you can receive data via a webhook, d.velop invoices must be able to reach your application. If you are using the cloud version of d.velop invoices, your application must be accessible from the internet.

Format of the document data

Both methods deliver the document data to you in a standardized format.

```
{
  "event_type": "integration.export",
  "_links": {
    "dmsobject": {
      "href": "https://example.d-velop.cloud/dms/r/21b9b52f-19e3-4b3d-8ac6-
b32d991f83e0/o2/P000000001"
    }
  },
  "workflow": {
    "voucher": {
      "doc_id": "P000000001",
      "company": {
        "nr": "01",
        "name": "docures AG"
      },
      "vendor": {
        "nr": "50001",
        "name": "Schrauben Meier GmbH"
      },
      "vendor_bank_account":  {
        "id":  "",
       "iban":  ""
      },
      "currency": {
        "id": "EUR",
        "name": "Euro",
        "code": "EUR"
      },
      "net_amount": 100.0,
      "gross_amount": 119.0,
      "pay_amount":  null,
      "vat_amount": 19.0,
      "document_date": "2020-05-05T00:00:00+00:00",
      "internal_number": "INT010291",
      "external_number": "INV12310",
      "payment_date": "2020-05-10T00:00:00+00:00",
      "date_of_supply": "2020-05-05T00:00:00+00:00"
      "financially_correct":  false,
      "document_type": {
        "id": "inv",
        "name": "Invoice",
        "credit_note": false
      },
      "payment_terms_type": "manual",
      "payment_terms": {
        "id": "14-3-90",
        "name": "14 Tage 3%, 90 Tage netto"
        "net_days": 90,
        "cashback_days1": 14,
        "cashback_percentage1": 3
      },
      "manual_payment_terms": {
        "net_due_date": "2025-09-05T00:00:00+00:00",
        "cashback_due_date1": "2025-05-27T00:00:00+00:00",
        "cashback_percentage1": 3.75
      },
      "due_dates": {
        "net_due_date": "2025-09-05T00:00:00.000+00:00",
        "cashback_due_date1": "2025-05-27T00:00:00.000+00:00",
        "cashback_amount1": 9.01
      }
      "posting_period": "2020-05-09",
      "posting_date":  "2020-05-09T00:00:00+00:00",
      "posting_text": "M3x3mm screws",
      "barcode": "BC1234",
      "custom1": "Sales"
      "line_items": {
        "5afe1329-83cd-4ccc-b806-f9ee458522b7": {
          "internal_id": "5afe1329-83cd-4ccc-b806-f9ee458522b7",
          "line_no": 1,
          "verified": true,
          "verifier": {
            "type": "user",
            "name": "mmus",
            "display_name": "Max Mustermann"
          },
          "verified_by": {
            "type": "user",
            "id": "257659",
            "name": "mmus",
            "display_name": "Max Mustermann"
          },
          "verified_at": "2020-05-05T10:25:00+01:00",
          "gl_account": {
            "nr": "6300"
          },
          "cost_center": {
            "nr": "1000"
          },
          "cost_unit": {
            "nr": "PJ0001"
          },
          "net_amount": 100.0,
          "gross_amount": 119.0,
          "pay_amount": null,
          "vat_amount": 19.0,
          "tax_code": {
            "id": "DE_S",
            "name": "Vorsteuer 19%",
            "percentage": 19.0
          },
          "order_number": "PO001",
          "order_line": 1,
          "order_id": "PO_001",
          "order_line_id": "PO_001_1",
          "quantity": {
            "invoiced": 2
          },
          "unit": "Pcs.",
          "unit_price": 50.0,
          "description": "Schraubendreher",
          "item_number": "4711",
          "discount_absolute": 0,
          "discount_per_unit": 0,
          "discount_percent": 0,
          "discount2_percent": 0,
          "discount3_percent": 0,
          "discount4_percent": 0,
          "discount5_percent": 0,
          "custom1": "Neubau"
        }
      }
    },
    "step": {
      "id": "64576949-ff8e-4e45-b0ec-7b4e25436331",
      "title": "Verification"
    }
  },
  "connection": {
    "from_step": {
      "id": "9080a8d4-6698-4613-b5c1-6c1ca103daab",
      "title": "Verification"
    },
    "to_step": {
      "id": "fd51ab72-aceb-4c1d-b775-ea3d411c9284",
      "title": "Approval"
    },
    "end_mode": null
  }
}
```

At the highest level, you receive the following data:

Property	Description
event_type	The type of the transferred event. At present, only the event **integration.export** is transferred.
_links	Links to related resources. At present, only the following types are supported: • **dmsobject**: Link to the invoice in the underlying document management system (d.velop documents or SharePoint). • **report_results_async**: The URL for reporting the processing result in the case of asynchronous processing. Asynchronous processing is explained in more detail in the sections on webhooks and the API for retrieving the data.
workflow	The workflow object. This object contains document data and status information about the workflow. It is described in more detail below.
connection	Information about the current step connection in the workflow. Always filled for events of the type **integration.export**.

Information about the step connection

An export (event type **integration.export**) is always triggered in the context of a connection between two workflow steps. This information helps you to determine, for example, whether the workflow ends with the transfer.

Property	Description
from_step	The starting step in the connection. If the workflow begins immediately, this property is **zero**.
to_step	The target step in the connection. If the workflow ends immediately, this property is **zero**.
end_mode	If the workflow ends with this connection, then **end_mode** contains information about the mode in which the workflow ends. Possible options: • **finished**: The workflow ends in the regular way. • **aborted**: The workflow is aborted or ended by admin.

Information about the current workflow status (workflow object)

Property	Description
voucher	The document data for the workflow.
step	The current step in the workflow. Caution: Since exports are usually triggered as part of step connections, the step changes when the workflow is completed. You can find the next step using the **connection** property on the highest level of the JSON structure.

Document header data (voucher object)

All the document header data is grouped together in this object.

Property	Description
doc_id	The ID of the document in the underlying document management system (d.velop documents or SharePoint).
company	The company (invoice recipient) for this document. • **nr**: The ID or number of the company. • **name**: The name of the company.
vendor	The vendor (supplier/invoice issuer) for this document. • **nr**: The ID/number of the vendor. • **name**: The name of the vendor.
vendor_bank_account	The vendor bank account for this document. • **id**: The ID of the vendor bank account from the master data (usually the ID from the ERP system). • **iban**: The IBAN of the bank account. • **bic**: The BIC of the bank account.
currency	The currency in which the invoice was issued. • **id**: The ID of the currency from the master data (usually the ID from the ERP system). • **name**: The name of the currency. • **code**: The currency code as per ISO 4217.
net_amount	The total net amount of the document.
vat_amount	The total tax amount of the document.
gross_amount	The total gross amount of the document.
document_date	The document/invoice date in the format yyyy-MM-ddTHH:mm:ssZ.
internal_number	The internal document number.
external_number	The external document number. The vendor's invoice number.
date_of_supply	The date of supply in the format yyyy-MM-ddTHH:mm:ssZ.
document_type	The document type. • **id**: The ID of the document type from the master data (usually the ID from the ERP system). • **name**: The name of the document type. • **credit_note**: Is the document a credit note?
payment_terms_type	Defines whether the payment terms of the document or manually entered payment terms apply for the workflow: • **manual**: The manually entered payment terms apply. • **voucher**: The payment terms of the document apply.
payment_terms	The payment terms for the invoice. • **id**: The ID of the payment terms from the master data (usually the ID from the ERP system). • **name**: The name of the payment term from the master data. • **net_days**: The net payment period in days. • **cashback_days1**: The cash discount period in days. • **cashback_percentage1**: The cash discount percentage.
manual_payment_terms	The manually entered payment terms for the workflow. If no values are entered, an empty object is transferred. • **net_due_date**: The manually entered net due date. • **cashback_due_date1**: The manually entered cash discount due date. • **cashback_percentage1**: The manually entered cash discount percentage as a decimal value.

Property	Description
due_dates	The net and cash discount due dates as well as the cash discount amount of the document. The due date must be interpreted depending on **payment_terms_type**. If **voucher** is entered as the value for **payment_terms_type**, the calculation of the values is based on the payment terms of the document. If **manual** is entered as the value for **payment_terms_type** , the due dates correspond to the manually entered due dates, and the calculation of the cash discount amount is based on the manually entered cash discount percentage. • **net_due_date**: The net due date. • **cashback_due_date1**: The cash discount due date. • **cashback_amount1**: The calculated cash discount amount (gross) as a decimal value.
payment_date	The payment date in the format yyyy-MM-ddTHH:mm:ssZ.
posting_period	The posting period for the invoice. The format is not specified. If the external system works with a posting date, we recommend using the field **posting_date** as an alternative.
posting_date	The posting date for the invoice in the format yyyy-MM-ddTHH:mm:ssZ. If the external system works with a posting period instead of a date, we recommend using the field **posting_period** as an alternative.
posting_text	The posting text for the invoice.
barcode	The barcode for the invoice.
regional_ch	Regional fields for customers from Switzerland. • **qr_code**: The Swiss QR code in raw format. • **qr_type**: The reference type from the Swiss QR code. • **qr_reference**: The reference from the Swiss QR code. • **qr_amount**: The amount from the Swiss QR code. • **qr_currency**: The currency information from the Swiss QR code. • **qr_iban**: The IBAN from the Swiss QR code. • **qr_info**: The information field from the Swiss QR code. • **qr_message**: The message field from the Swiss QR code. • **esr_line**: The ESR line in raw format.
custom1 to custom20	Custom fields at header level.

Information about document line items (line item object)

Each individual document line item supports the following properties:

Property	Description
internal_id	The unique ID of the document line item.
line_no	The line number of the document line item.
verified	Indicates whether the line item is marked as factually correct. Caution: This property is unavailable if you are using the “Advanced verification” preview feature.
verifier	Indicates the person who is authorized to mark this line item as factually correct. Caution: This is not necessarily the person that actually marked the line item as factually correct. This person can be found in the property **verified_by**. Attention: This property is unavailable if you are using the “Advanced verification” preview feature.
verified_by	Indicates the person who marked this line item as factually correct. Caution: This property is unavailable if you are using the “Advanced verification” preview feature.

Property	Description
verified_at	The time at which this line item was marked as factually correct. In the format yyyy-MM-dd HH:mm:ss, time zone UTC. Caution: This property is unavailable if you are using the **Advanced verification** preview feature.
gl_account	The G/L account of the document line item. • **nr**: The number of the G/L account.
cost_center	The cost center of the document line item. • **nr**: The number of the cost center.
cost_unit	The cost unit of the document line item. • **nr**: The number of the cost unit.
procurement_category	The procurement category of the document line item. • **nr**: The number of the procurement category.
net_amount	The net amount of the document line item.
tax_code	The tax code of the document line item. • **id**: The ID of the tax code from the master data (usually the ID from the ERP system). • **name**: The name of the tax code. • **percentage**: The tax rate.
vat_amount	The tax amount for the document line item. Caution: If the field is a writable field in d.velop invoices, the tax amount may differ from the tax rate of the selected tax code.
gross_amount	The gross amount of the document line item.
unit	The unit of measure of the document line item.
quantity	The quantity of the document line item. • **invoiced**: The quantity invoiced using this line item.
unit_price	The unit price of the document line item.
price_unit	The price unit of the document line item. The price unit is relevant if the supplier and company do not work with the same unit of measure for the quantity. The unit price is divided by the price unit before it is multiplied by the quantity.
item_number	The item number of the document line item.
description	The description of the document line item.
order_number	The number of the order belonging to this document line item.
order_line	The number of the order line belonging to this document line item.
order_id	The ID of the order belonging to this document line item (corresponds to the ID transferred in the master data).
order_line_id	The ID of the order line belonging to this document line item (corresponds to the ID transferred in the master data).
goods_receipt_nr	The number of the goods receipt belonging to this document line item.
goods_receipt_line_no	The number of the goods receipt line item belonging to this document line item.
goods_receipt_id	The ID of the goods receipt belonging to this document line item (corresponds to the ID transferred in the master data).
goods_receipt_line_item_id	The ID of the goods receipt line belonging to this document line item (corresponds to the ID transferred in the master data).
discount_absolute	Absolute discount on the net amount of the line item.
discount_per_unit	Absolute discount on the unit price of the line item.
discount_percent	First percentage discount on the net amount of the line item.
discount2_percent	Second percentage discount on the net amount of the line item.
discount3_percent	Third percentage discount on the net amount of the line item.

Property	Description
discount4_percent	Fourth percentage discount on the net amount of the line item.
discount5_percent	Fifth percentage discount on the net amount of the line item.
custom1 to custom20	Custom fields at line item level.

Receiving data via webhook

You can use webhooks to receive document data from d.velop smart invoice in an external system via HTTP. To do so, you can use an integrating application where you store an HTTP endpoint.

What are webhooks?

d.velop smart invoice can use webhooks to inform an external application (an ERP system, for example) when certain events occur. The external application can respond to this event and create a posting proposal, for example.

Webhooks can be created using any technology. To do this, the application has to provide an HTTP endpoint that receives POST requests in a format which is specified by d.velop smart invoice.

Implementing a webhook endpoint

To implement a webhook endpoint, you must make a new endpoint available in your application. The deployment of the endpoint is dependent on the technology you use: in a PHP application, for example, through a new .php file, in Microsoft .NET ASP MVC through a new route.

The webhook endpoint extracts and processes the JSON from the POST request. You must then send an HTTP status code to signal the successful processing to d.velop smart invoice.

> ### Important
> You must acknowledge the webhook within 30 seconds. Otherwise, d.velop smart invoice interprets the transfer as failed and continues the review and approval workflow in the defined error step.

The data that d.velop smart invoice sends to the endpoint corresponds to the event object from the API.

You can additionally authenticate incoming requests to ensure that the requests actually originate from d.velop smart invoice.

Status codes and return values

If the webhook request was processed correctly, respond with HTTP status code **200**. If you encountered errors due to a program error, respond with status code **500**.

If invalid information, such as a non-existent general ledger account, is the cause, respond with status code **400**. In this case, you can also include an error message in the response. The error message is displayed for your users in d.velop smart invoice. You must enter the error message in German and English.

Example:

```
{
  "error": {
    "de": "Die Buchungsperiode wurde bereits geschlossen.",
    "en": "The posting period has already been closed."
  }
}
```

Authenticating requests

When you implement a webhook endpoint, you must ensure that incoming requests actually originate from d.velop smart invoice. For this purpose, each request includes a signature in the header **X-Smart-Invoice-Signature**. To verify the signature, you require the secret from the webhook integration settings. The secret differs in each integrating application.

The header contains the signature and a timestamp. This timestamp is specified as a Unix timestamp in the key **t**. You can find the signature in the key **v1**.

```
X-Smart-Invoice-Signature:
t=12839891289,v1=389413894189038901289 30jaskldjl221
```

The signature is created by d.velop smart invoice in accordance with the HMAC-SHA256 method. You can now review the signature.

This is how it works

1. Compare the timestamp from the header **X-Smart-Invoice-Signature** with the actual time. If the difference is more than five minutes, discard the request as not authenticated and respond with an appropriate HTTP status code.
2. Link the timestamp from the header **X-Smart-Invoice-Signature** and the body of the request with the following information:

   ```
   timestamp.body
   ```
3. Use the secret from the integrating webhook application to calculate an HMAC-SHA256 value in a hexadecimal representation from the character string which you created.
4. Now compare the value that you calculated with the value from the header **X-Smart-Invoice-Signature**. To do this, use a constant checking comparison function to prevent a timing attack. If the values differ from each other, discard the request as not authenticated. Otherwise, you can continue processing the request.

Retrieving transfers via REST API

You can retrieve documents that are ready for transfer from an external application via the REST API of d.velop smart invoice. To do this, you can access the interfaces in order to retrieve documents and to confirm a successful transfer.

The processing sequence

When document data is made available for retrieval by d.velop smart invoice, the document data is available for two days or 2,879 minutes by default. You must retrieve the document data within this time slot using an external application, and report whether the processing was successful. If the time runs out without any response from the external application, the transfer is marked as failed. The document is then diverted to the error location which was configured in d.velop smart invoice.

> ### Note
> The two-day time slot can be changed by the administration. The time slot can be enhanced by up to 28 days or 40,319 minutes.
>
> We recommend that you keep the time slot as small as possible.

If the external application reports that the transfer was successful, the document is further processed based on the d.velop smart invoice configuration. This often means that the workflow is ended by the successful transfer.

If a problem has occurred and the external application reports an error, the document is diverted to the configured error location. The error message from the external application is displayed to the user.

Integration key

Each external application requires an integration key to request document data. This integration key is automatically generated by d.velop smart invoice and can be viewed in the administration interface. The integration key is not an authentication feature; instead, it serves to differentiate between transfers for different applications.

Retrieving transfers

You can regularly retrieve a list of all the outstanding transfers for your integration key as follows:

Request

```
GET /smartinvoice/api/v1/transfers?integration_key=abc
Accept: application/json
```

Response

```
{
  "_links": {
    "next": {
      "href": "https://test.d-velop.cloud/smartinvoice/api/v1/transfers/..."
    },
    "previous": {
      "href": "https://test.d-velop.cloud/smartinvoice/api/v1/transfers/..."
    },
  },
  "transfers": [
    {
      "_links": { ... },
      "workflow": {
        ...
      }
    }
  ]
}
```

Property	Description
_links	Transfers may be returned in multiple pages. Use the specified URLs to retrieve additional pages. • **next**: Reference to the next page. • **previous**: Reference to the previous page.
transfers	The workflows that are available for transfer. The format of each transfer corresponds to the document data format described.

Reporting the results

After you have processed the data from a transfer, you have to report a result. If you do not return a result on time, the transfer is classed as failed.

To report a result, send a POST request to the URL specified under **_links** and **report_results_async** in the transfer.

Request

```
POST /smartinvoice/api/v1/transfers/:id
Content-Type: application/json

{
  "successful": false,
  "error": {
    "de": "Transfer fehlgeschlagen.",
    "en": "Transfer failed."
  }
}
```

Property	Description
successful	Boolean. Mandatory. Indicates whether the transfer was successful. If you enter **false** here, you must also send the error object (**error**). With **true** this is not required.
error	You can save error messages in various languages. The languages German and English are mandatory. Messages entered here are displayed for the end user.

1.3.2. Transferring an approval matrix (d.velop smart invoice)

You can use an API to transfer data from any data source to a configured approval matrix.

Jobs

You can transfer multiple records at the same time. d.velop smart invoice processes the data asynchronously. It always generates exactly one job, whose job ID you receive as the response.

Response

```
{
  "job_id": "65866efa-e8a6-4c9c-ae30-60121b6d039f"
}
```

Retrieving the current status of a job

You can retrieve the status of the job as follows based on the job ID returned to you by the endpoint and the matrix ID:

Request

```
   GET
/smartinvoice/api/v1/approval_matrices/:matrix_id/rows/batch/jobs/:job_id
Accept: application/json
```

As a response, you receive a JSON object with information about the job.

Response

```
{
  "status": "finished",
  "issues": []
}
```

Property	Description
status:	String. The current status of the job. The possible options are as follows: • **waiting**: The job is in the queue. • **processing**: The job is being processed. • **successful**: The job was processed successfully. • **failed**: An error occurred.
more_issues	Boolean. Indicates whether more errors occurred than indicated under **issues**.

Property	Description
issues	Object. Contains issues that occurred during processing. **message**: String. An error message describing the issue that occurred. **record number**: Number. Data record from the JSON that was sent to the endpoint as the request payload.

Approval matrices

An approval matrix lets you automatically define suitable assignees for an invoice based on the data entered for the invoice. You can transfer data records for an approval matrix configured in d.velop smart invoice as follows:

Request

```
   POST /smartinvoice/api/v1/approval_matrices/:matrix_id/rows/batch
Content-Type: application/json

{
  "rows": [
    {
      "user": {
        "type": "idp",
        "name": "username@domain.de"
      },
      "limit": {
        "amount": 10000.00,
        "currency": "EUR"
      },
      "column1": "01",
      "column2": "1300",
      "column3": {
        "user": {
          "type": "idp",
          "name": "username@domain.de"
        }
      },
      "column4": {
        "group": {
          "type": "idp",
          "name": "groupname"
        }
      }
    }
  ]
}
```

Each row of an approval matrix supports the following properties:

Property	Description
user	Object. Mandatory. User object that applies as the approver for this matrix row.
type	String. Mandatory. The user object type. **idp**: The user provided via IDP. **aad**: User from Azure AD.
name	String. Mandatory. Name of the user.

Property	Description
limit	Object. Mandatory. The user's approval limit for this matrix row in the specified currency. **amount**: Number. Mandatory. The amount of the approval limit. **currency**: String. Mandatory. Currency code for the approval limit according to ISO 4217.
column1-20	String/object. Optional. Value for the column associated with **column<1–20>** for this matrix row. You can find the mapping of **column<1–20>** for the matrix columns in the matrix configuration. In an approval matrix, you can reference columns that contain data for users or user groups. If you add a value for a matrix row for such a column, the type of the value must be **Object** and the value must represent a valid user object or user group object. **Value has to be a user:** **user**: Object. Mandatory. User object to be set as the value for the matrix row column associated with **column<1–20>**. • **type**: String. Mandatory. The user object type. • • **idp**: The user provided via IDP. • **aad**: User from Azure AD. • **name**: String. Mandatory. Name of the user. **Value has to be a user group:** **group**: Object. Mandatory. User group object to be set as the value for the matrix row column associated with **column<1–20>**. • **type**: String. Mandatory. The user group object type. • **idp**: The user group provided via IDP. • **aad**: User group from SharePoint. • **name**: String. Mandatory. Name of the user group.

1.3.3. Transferring a verification matrix (d.velop smart invoice)

You can use an API to transfer data from any data source to a configured verification matrix.

Jobs

You can transfer multiple records at the same time. d.velop smart invoice processes the data asynchronously. It always generates exactly one job, whose job ID you receive as the response.

Response

```
{
  "job_id": "65866efa-e8a6-4c9c-ae30-60121b6d039f"
}
```

Retrieving the current status of a job

You can retrieve the status of the job as follows based on the job ID returned to you by the endpoint and the matrix ID:

Request

```
   GET
/smartinvoice/api/v1/verification_matrices/:matrix_id/rows/batch/
jobs/:job_id
Accept: application/json
```

As a response, you receive a JSON object with information about the job.

Response

```
{
  "status": "finished",
  "issues": []
}
```

Property	Description
status:	String. The current status of the job. The possible options are as follows: • **waiting**: The job is in the queue. • **processing**: The job is being processed. • **successful**: The job was processed successfully. • **failed**: An error occurred.
more_issues	Boolean. Indicates whether more errors occurred than indicated under **issues**.
issues	Object. Contains issues that occurred during processing. **message**: String. An error message describing the issue that occurred. **record number**: Number. Data record from the JSON that was sent to the endpoint as the request payload.

Verification matrices

You can use a verification matrix to define suitable verifiers for an invoice as part of the extended verification process. This is done based on the data recorded for the invoice. You can transfer data records for a verification matrix configured in d.velop smart invoice as follows:

```
   POST /smartinvoice/api/v1/verification_matrices/:matrix_id/rows/batch
Content-Type: application/json
{
  "rows": [
    {
      "verifiers": [
        {
          "id": "fa2461f6-f112-408c-aaad-4f35f996bd66",
          "user": {
            "type": "idp",
            "name": "username@domain.de"
          },
        },
        {
          "id": "bc7748fc-a73a-401d-ab1d-4f3111a6be12",
          "group": {
            "type": "idp",
            "name": "groupname"
          },
        }
      ],
      "limit": {
        "amount": 10000.00,
        "currency": "EUR"
      },
      "column1": "01",
      "column2": "1300"
    }
  ]
}
```

Each row of a verification matrix supports the following properties:

Property	Description
verifiers	Object. Mandatory. Contains the verifier or verifier groups for the matrix row verification types.

Property	Description
id	String. Mandatory. The ID of the verification type from the workflow definition.
user|group	Object. Mandatory. The responsible user or user group for the specified verification type. **Value has to be a user:** **user**: Object. Mandatory. Object for the user that, as the verifier, is responsible for the specified verification type. • **type**: String. Mandatory. The user object type. • **idp**: The user provided via IDP. • **aad**: User from Azure AD. • **name**: String. Mandatory. Name of the user. **Value has to be a user group:** **group**: Object. Mandatory. Object for the user group that, as the verifier group, is responsible for the specified verification type. • **type**: String. Mandatory. The user group object type. • **idp**: The user group provided via IDP. • **aad**: User group from SharePoint. • **name**: String. Mandatory. Name of the user group.
limit	Object. Mandatory. The amount limit for the reviewer of this matrix row in the specified currency. **amount**: Number. Mandatory. The amount of the amount limit. **currency**: String. Mandatory. Currency code for the amount limit according to ISO 4217.
column1-20	String. Optional. Value for the column associated with **column<1–20>** for this matrix row. You can find the mapping of **column<1–20>** for the matrix columns in the matrix configuration.

1.3.4. Transferring master data (d.velop smart invoice)

In d.velop smart invoice, you can use the master data API to transfer data from an external ERP system to a bucket. You have two options:

1. Batch transfer: With the batch transfer of master data, you regularly synchronize all relevant master data from an external ERP system with d.velop smart invoice.
2. Transfer individual master data records: You can add or update individual master data records synchronously to a bucket.

Batch transfer of master data

With the batch transfer of master data, you can regularly synchronize your data from external ERP systems with d.velop smart invoice. The synchronization takes place asynchronously.

Determining the bucket ID

You need at least one master data bucket in order to synchronize master data with d.velop smart invoice. You require the bucket ID to synchronize the master data with the corresponding bucket. You can find the bucket ID in the d.velop smart invoice administration interface. The available buckets are shown under **Buckets**. The bucket IDs are shown in the **ID** column.

Jobs

For every entity, there are endpoints for transferring multiple data records at the same time (these endpoints end in **/batch**). d.velop smart invoice processes the data sent to these endpoints asynchronously. Each of the endpoints listed below for transferring data returns the ID of the generated job. Exactly one job is always returned.

Response

```
{
  "jobs": [
    {
      "job_id": "65866efa-e8a6-4c9c-ae30-60121b6d039f"
    }
  ]
}
```

Retrieving the current status of a job

You can use the job ID returned to you by the relevant endpoint to retrieve the status of the job as follows:

Request

```
GET /smartinvoice/api/v1/masterdata/import_jobs/65866efa-e8a6-4c9c-
ae30-60121b6d039f
Accept: application/json
```

As a response, you receive a JSON object with information about the job.

Response

```
{
  "job_id": "65866efa-e8a6-4c9c-ae30-60121b6d039f",
  "status": "successful"
}
```

Property	Description
job_id	String. The ID of the job.
status	String. The current status of the job. The possible options are as follows: • **queued**: The job is still being processed. • **successful**: The job was processed successfully. • **failed**: An error occurred.

Companies

Companies are the recipients of invoices. A company in d.velop smart invoice usually corresponds to a company or company code in the ERP system. Every invoice requires a company. You transfer companies as follows:

Request

```
POST /smartinvoice/api/v1/buckets/:bucket_id/companies/batch
Content-Type: application/json

{
  "companies": [
    {
      "id": "01",
      "name": "docures AG",
      "parent_id": null,
      "address": "Musterstr. 23",
      "city": "Musterstadt",
      "zip_code": "12345",
      "local_currency": "EUR",
      "country": "DE"
    }
  ]
}
```

Each individual company supports the following properties:

Property	Description
id	String. Mandatory. The ID or number of the company. This value can be seen on the user interface.
name	String. Mandatory. The name of the company. This value can be seen on the user interface. The name of the company is an important criterion for automatically identifying the company in d.velop invoices. Where possible, it should be entered identically to how it is shown on the invoices.
address	String. The company's address information, e.g street and house number. The address of the company is an important criterion for automatically identifying the company in d.velop invoices.
city	String. The location or city of the company.
zip_code	String. The zip code of the location or city.
local_currency	String. The local currency of the company. This value is used to convert invoice amounts into the local currency for verification in conditions. You must enter this value as a code in accordance with ISO 4217 (for example: EUR or USD).
country	String. The country of the company. This value is used to select the rule set for the automatic check of the local national regulations. You must enter this value as an ISO 3316-1 alpha-2 code (for example, DE or CH).

Vendors

Vendors are the senders of invoices. You transfer vendors as follows:

Request

```
POST /smartinvoice/api/v1/buckets/:bucket_id/vendors/batch
Content-Type: application/json

{
  "vendors": [
    {
      "company_id": "01",
      "id": "50001",
      "name": "Schrauben Meier GmbH",
      "address": "Teststr. 6",
      "city": "Kiel",
      "zip_code": "24145",
      "country": "DE",
      "email": "schraubenmeier@example.com",
      "vat_id": "DE99999999",
      "registration_id": "1028502",
      "payment_terms_id": "10",
      "tax_category_1": "NATIONAL"
    }
  ]
}
```

Each individual vendor supports the following properties:

Property	Description
company_id	String. Mandatory. The company to which this vendor belongs.
id	String. Mandatory. The ID or number of the vendor in the ERP system. This ID can be viewed by users in d.velop invoices. Must be unique for each **company_id**.

Property	Description
name	String. Mandatory. The name of the vendor. If possible, with the same spelling used on the invoices for this vendor.
address	String. Mandatory. The address line for the vendor, e.g. street and house number.
city	String. Mandatory. The location or city of the vendor.
zip_code	String. Mandatory. The zip code of the location or city of the vendor.
country	String. Mandatory. The country of the vendor. This value is used to select the rule set for the automatic check of the local national regulations. You must enter this value as an ISO 3316-1 alpha-2 code (for example, DE or CH).
email	String. The e-mail address of the vendor. Where available, the address the vendor enters on his or her invoices.
vat_id	String. The VAT ID number of the vendor. This property is important for automatically determining the vendor in d.velop invoices.
registration_id	String. The tax ID of the vendor. This may be a national tax ID in a non-EU country, for example.
payment_terms_id	String. The ID of the primary payment terms for this vendor.
tax_category_1	String. The first part of the category for automatically determining the correct tax code. You can find a detailed description of this behavior in the section on tax codes.

Vendor Bank Accounts

You can save multiple bank accounts for each vendor. The bank accounts are important for both identifying the vendor while reading the invoice and comparing the information entered on the invoice with the data from the ERP system. You transfer bank accounts as follows:

Request

```
POST /smartinvoice/api/v1/buckets/:bucket_id/vendor_bank_accounts/batch
Content-Type: application/json

{
  "vendor_bank_accounts": [
    {
      "company_id": "01",
      "vendor_id": "50001",
      "iban": "DE02100100100006820101",
      "bic": "PBNKDEFF",
      "primary": true
    }
  ]
}
```

Each individual vendor bank account supports the following properties:

Property	Description
company_id	String. Mandatory. The company to which this vendor bank account belongs.
vendor_id	String. Mandatory. The vendor to which this bank account belongs.
id	String. Mandatory. The ID of this bank account in the ERP system.
iban	String. Mandatory. The IBAN of the relevant bank account.
bic	String. The BIC of the relevant bank account.

Property	Description
primary	Boolean. Mandatory. Is this bank account the vendor’s primary bank account?

Payment Terms

d.velop smart invoice uses payment terms to calculate escalation times for net payment periods and discount periods. You transfer payment terms as follows:

Request

```
POST /smartinvoice/api/v1/buckets/:bucket_id/payment_terms/batch
Content-Type: application/json

{
   "payment_terms":[
      {
         "company_id":"01",
         "id":"NET30",
         "net_days":30,
         "cashback_days1":10,
         "cashback_percentage1":2.5,
         "name":"30 days net, 10 days 2.5% discount"
      }
   ]
}
```

Each individual payment term supports the following properties:

Property	Description
company_id	String. Mandatory. The company to which these payment terms belong.
id	String. Mandatory. The ID of the payment terms in the ERP system.
net_days	Number. Mandatory. The net payment period in days. If you enter 0, this is interpreted as “no period.”
cashback_days1	Number. Mandatory. The cash discount period in days. If you enter 0, this is interpreted as “no period.”
cashback_percentage1	Number. Mandatory. The cash discount rate as a percentage.
name	String. The name for these payment terms.

Document Types

The document type identifies the type of the invoice. At minimum, the two document types invoice and credit note should be available. You transfer document types as follows:

Request

```
POST /smartinvoice/api/v1/buckets/:bucket_id/document_types/batch
Content-Type: application/json

{
  "document_types": [
    {
      "company_id": "01",
      "id": "inv",
      "name": "Invoice",
      "credit_note": false
    }
  ]
}
```

Each individual document type supports the following properties:

Property	Description
company_id	String. The company to which this document type belongs. If you do not make an entry, the document type is valid for all companies.
id	String. Mandatory. The ID of the document type. This value can be viewed by users in d.velop invoices.
name	String. Mandatory. The name of the document type.
credit_note	Boolean. Mandatory. Is the document a credit note?

Currencies

Each document in d.velop smart invoice is assigned to a currency. You transfer the currencies that are transferred by your ERP system. You transfer currencies as follows:

Request

```
POST /smartinvoice/api/v1/buckets/:bucket_id/currencies/batch
Content-Type: application/json

{
  "currencies": [
    {
      "company_id": "01",
      "id": "EUR",
      "name": "Euro",
      "code": "EUR"
    }
  ]
}
```

Each individual currency supports the following properties:

Each individual currency supports the following properties:

Property	Description
company_id	String. The company to which this currency belongs. If you do not make an entry, the currency is valid for all companies.
id	String. Mandatory. The ID of the currency in the ERP system. This value can be viewed by users in d.velop invoices.
name	String. Mandatory. The name of the currency.
code	String. Mandatory. The currency code as per ISO 4217.

Tax Codes

A tax code can be assigned to each document line in d.velop smart invoice. You transfer tax codes as follows:

Request

```
POST /smartinvoice/api/v1/buckets/:bucket_id/tax_codes/batch
Content-Type: application/json

```

```
{
  "tax_codes": [
    {
      "company_id": "01",
      "id": "S-DE",
      "name": "VSt 19%",
      "percentage": 19.0,
      "tax_category_1": "NATIONAL",
      "tax_category_2": "FULL"
    }
  ]
}
```

Each individual tax code supports the following properties:

Property	Description
company_id	String. Mandatory. The company to which this tax code belongs. If you do not make an entry, the tax code is valid for all companies.
id	String. Mandatory. The ID of the tax code in the ERP system. This value is displayed for users in d.velop invoices.
name	String. Mandatory. The name of the tax code.
percentage	Number. Mandatory. The percentage of the tax code.
tax_category_1	String. The first part of the tax category for automatically determining the tax code.
tax_category_2	String. The second part of the tax category for automatically determining the tax code.

Automatically determining the tax code

Using the values **tax_category_1** and **tax_category_2**, d.velop smart invoice can automatically determine a suitable tax code based on the detected and/or selected document data. To do so, it compares the **tax_category_1** and **tax_category_2** values of the vendor and of the G/L account or procurement category. If there is a clear match, the corresponding tax code is entered automatically.

G/L Accounts

You transfer G/L accounts as follows:

Request

```
POST /smartinvoice/api/v1/buckets/:bucket_id/gl_accounts/batch
Content-Type: application/json

{
  "gl_accounts": [
    {
      "company_id": "01",
      "nr": "6300",
      "name": "Sonstige betriebliche Aufwendungen"
    }
  ]
}
```

Each individual G/L account supports the following properties:

Property	Description
company_id	String. The company to which this G/L account belongs. If you do not make an entry, the G/L account is valid for all companies.
nr	String. Mandatory. The number of the G/L account.
name	String. Mandatory. The name of the G/L account.
tax_category_2	String. The second part of the tax category for automatically determining the tax.

Cost Centers

You transfer cost centers as follows:

Request

```
POST /smartinvoice/api/v1/buckets/:bucket_id/cost_centers/batch
Content-Type: application/json

{
  "cost_centers": [
    {
      "company_id": "01",
      "nr": "1000",
      "name": "Verwaltung"
    }
  ]
}
```

Each individual cost center supports the following properties:

Property	Description
company_id	String. The company to which this cost center belongs. If you do not make an entry, the cost center is valid for all companies.
nr	String. Mandatory. The number of the cost center.
name	String. Mandatory. The name of the cost center.

Cost Units

You transfer cost units as follows:

Request

```
POST /smartinvoice/api/v1/buckets/:bucket_id/cost_units/batch
Content-Type: application/json

{
  "cost_units": [
    {
      "company_id": "01",
      "nr": "PJ001",
      "name": "Projekt Neubau"
    }
  ]
}
```

Each individual cost unit supports the following properties:

Property	Description
company_id	String. The company to which this cost unit belongs. If you do not make an entry, the cost unit is valid for all companies.
nr	String. Mandatory. The number of the cost unit.
name	String. Mandatory. The name of the cost unit.

Other Dimensions

Other dimensions are data records that can be used in custom fields in document lines (for custom fields in the header data, see “Custom Entities”). Each additional dimension is identified using a freely definable type (**type** field). All data records with the same type count as one additional dimension. You can assign an additional dimension to each custom field in d.velop smart invoice. You transfer additional dimensions as follows:

Request

```
POST /smartinvoice/api/v1/buckets/:bucket_id/other_dimensions/batch
Content-Type: application/json

{
  "other_dimensions": [
    {
      "company_id": "01",
      "type": "project",
      "nr": "PJ001",
      "name": "Projekt Neubau",
      "parent_dimension_type": "Projektleiter",
      "parent_dimension_id": "1000"
    }
  ]
}
```

Each individual dimension value supports the following properties:

Property	Description
company_id	The company to which this dimension value belongs. If you do not make an entry, the dimension value is valid for all companies.
type	String. Mandatory. The type of this dimension value. Types are used to keep all the data for an additional dimension together.
nr	String. Mandatory. The ID or number of the dimension value.
name	String. Mandatory. The name of the dimension value.
parent_dimension_type	String. The type of the additional dimension, which determines this dimension value.
parent_dimension_id	String. The ID of the additional dimension value of the type **parent_dimension_type**, which determines this dimension value.

Custom Entities

Custom entities are data records that can be used in custom fields in the document header (for custom fields in document lines, see “Other Dimensions”). Each custom entity is identified using a freely definable type (**type** field). All data records with the same type count as one custom entity. You can assign a custom entity to each custom field in d.velop smart invoice. You transfer custom entities as follows:

Request

```
POST /smartinvoice/api/v1/buckets/:bucket_id/custom_entities/batch
Content-Type: application/json
```

```
{
  "custom_entities": [
    {
      "company_id": "01",
      "type": "project",
      "id": "PJ001",
      "name": "Projekt Neubau"
    }
  ]
}
```

Each individual entity value supports the following properties:

Property	Description
company_id	The company to which this entity value belongs. If you do not make an entry, the entity value is valid for all companies.
type	String. Mandatory. The type of the entity value. Types are used to keep all the data for a custom entity together.
id	String. Mandatory. The ID of this entity value.
name	String. Mandatory. The name of this entity value.

Purchase Orders

Purchase orders and purchase order line items are used to compare invoices and purchase orders in d.velop smart invoice. You transfer purchase orders and purchase order lines as follows:

Request

```
POST /smartinvoice/api/v1/buckets/:bucket_id/purchase_orders/batch
Content-Type: application/json

{
  "purchase_orders": [
    {
      "company_id": "01",
      "id": "PO001",
      "nr": "PO001",
      "name": "Office Supplies",
      "vendor_id": "50001",
      "status": 1,
      "custom1": null,
      "custom2": null,
      "custom3": null,
      "custom4": null,
      "custom5": null,
      "custom6": null,
      "custom7": null,
      "custom8": null,
      "custom9": null,
      "custom10": null,
      "custom11": null,
      "custom12": null,
      "custom13": null,
      "custom14": null,
      "custom15": null,
      "custom16": null,
      "custom17": null,
      "custom18": null,
      "custom19": null,
      "custom20": null,
      "line_items": [
        {
          "company_id": "01",
          "purchase_order_id": "PO001",
          "id": "PO001_1",
          "line_no": 1,
          "quantity_ordered": 10.0,
          "quantity_received": 5.0,
          "quantity_not_invoiced": 5.0,
          "description": "Printer Paper, A4",
          "subtotal": 200.0,
          "unit": "Box",
          "unit_price": 20.0,
          "price_unit": 1.0,
          "gl_account": "6300",
          "cost_center": "1000",
          "cost_unit": null,
          "item": "102912",
          "tax_code_id": "DE_S",
          "discount_absolute": 0.0,
          "discount_per_unit": 0.0,
          "discount_percent": 0.0,
          "discount2_percent": 0.0,
          "discount3_percent": 0.0,
          "discount4_percent": 0.0,
          "discount5_percent": 0.0,
          "custom1": null,
          "custom2": null,
          "custom3": null,
          "custom4": null,
          "custom5": null,
          "custom6": null,
          "custom7": null,
          "custom8": null,
          "custom9": null,
          "custom10": null,
          "custom11": null,
          "custom12": null,
          "custom13": null,
          "custom14": null,
          "custom15": null,
          "custom16": null,
          "custom17": null,
          "custom18": null,
          "custom19": null,
          "custom20": null,
          "procurement_category": null,
          "erp_id": "100",
          "type": "header_surcharge",
          "surcharged_line_item_id": "1",
          "surcharge_category": "pcs",
          "surcharge_value": "20.00",
          "responsible": "email@domain.de",
          "goods_receipt_reference_mandatory": true
        }
      ]
    }
  ]
}
```

Each individual purchase order supports the following properties:

Property	Description
company_id	String. Mandatory. The company to which this purchase order belongs.
id	String. Mandatory. The ID of this purchase order. This is not displayed for users in d.velop invoices.
nr	String. Mandatory. The number of this purchase order.
name	String. Mandatory. A description of the purchase order.
vendor_id	String. Mandatory. The ID of the vendor to which this purchase order belongs.
status	Number. The following values and mapped statuses are possible: **1**: Open **2**: Approved **3**: Confirmed **4**: Partially received **5**: Received **6**: Partially invoiced **7**: Invoiced **8**: Cancelled
custom1 to custom20	String. Custom fields at header level The values of these fields are transferred to the relevant custom fields in the invoice header as long as the fields are empty on the invoice.

Each individual purchase order line item supports the following properties:

Property	Description
company_id	String. Mandatory. The company to which this purchase order line belongs.
id	String. Mandatory. The ID of the purchase order line. This value must be unique across all the purchase orders. That means that even two different purchase orders must not share the same purchase order line ID.
line_no	Number. Mandatory. The number of the line within the purchase order.
quantity_ordered	Number. Mandatory. The total quantity ordered.
quantity_received	Number. Mandatory. The total quantity received.
quantity_not_invoiced	Number. Mandatory. The quantity that has not been invoiced yet. Caution: d.velop invoices does not use this value for quantity checks.
item	String. Mandatory. The item number of the purchase order line.
procurement_category	String. The procurement category of the purchase order line.
description	String. Mandatory. A text describing the purchase order line, such as the name of the item.

Property	Description
unit	String. Mandatory. The unit for the ordered quantity. For example, pieces.
unit_price	Number. Mandatory. The unit price for the ordered item.
price_unit	Number. Mandatory. The price unit or packaging unit if the supplier bills in a different quantity unit than the one managed in the ERP system. In cases of doubt, enter 1.0.
subtotal	Number. Mandatory. Overall (net) amount of the purchase order line.
gl_account	String. The G/L account for the purchase order line.
cost_center	String. The cost center for the purchase order line.
cost_unit	String. The cost unit for the purchase order line.
tax_code_id	String. The ID of the associated tax code.
discount_absolute	Number. Absolute discount on the overall (net) amount of the purchase order line.
discount_per_unit	Number. Absolute discount on the unit price for the purchase order line.
discount_percent	Number. Percentage discount on the overall (net) amount of the purchase order line.
discount2_percent	Number. Second percentage discount on the overall (net) amount of the purchase order line.
discount3_percent	Number. Third percentage discount on the overall (net) amount of the purchase order line.
discount4_percent	Number. Fourth percentage discount on the overall (net) amount of the purchase order line.
discount5_percent	Number. Fifth percentage discount on the overall (net) amount of the purchase order line.
custom1 to custom20	String. Custom fields. The values in the fields are transferred to the corresponding fields in the invoice line created from the purchase order.
type	String. Mandatory. **header_surcharge** or **line_item_surcharge**.
surcharged_line_item_id	String. The ID of the purchase order line item for which the surcharge applies.
surcharge_category	String. **pcs** (surcharge per piece) or **fixed** (fixed surcharge) or **percent** (percentage surcharge).
surcharge_value	String. Value of the surcharge.
responsible	String. User name of the person responsible for the purchase order line item. Note this is case sensitive.
goods_receipt_reference_mandatory	Boolean. Indicates whether the assignment of a goods receipt line to this order line item is mandatory. You can find further information on the use of this feature in the administration manual of d.velop smart invoice in the section Working with goods receipts.

Note

Surcharge line items are separate order line items

Note that a surcharge line item in a purchase order represents a separate order line item. The reference to the purchase order line item for which the surcharge applies is established via the **surcharged_line_item_id** property.

Goods Receipt

Note

Goods receipts are still being previewed at the moment. Please contact d.velop if you would like to use goods receipts.

Goods receipts and goods receipt lines are used by d.velop invoices to compare purchase orders, goods receipts and invoices with each other. You transfer goods receipts and goods receipt lines as follows:

Request

```
POST /smartinvoice/api/v1/buckets/:bucket_id/goods_receipts/batch
Content-Type: application/json

{
  "goods_receipts": [
    {
      "company_id": "01",
      "vendor_id": "50001",
      "id": "GR001",
      "nr": "GR001",
      "creation_date": "2022-04-07",
      "delivery_slip_nr": "LS001",
      "status": 0,
      "line_items": [
        {
          "company_id": "01",
          "id": "GR001_1",
          "line_no": 1,
          "goods_receipt_date": "2022-04-07",
          "quantity": 2.7,
          "purchase_order_line_id": "PO001_1"
        }
      ]
    }
  ]
}
```

Each individual goods receipt supports the following properties:

Property	Description
company_id	String. Mandatory. The company to which this goods receipt belongs.
vendor_id	String. Mandatory. The vendor to which this goods receipt belongs.
id	String. Mandatory. The ID of the goods receipt. This value is not displayed for users in d.velop invoices.
nr	String. Mandatory. The number of the goods receipt.
creation_date	Date. Mandatory. The date on which the goods receipt was created. Format YYYY-MM-DD.
delivery_slip_nr	String. Mandatory. The vendor delivery slip number for which this goods receipt was posted.

Each individual goods receipt line item supports the following properties:

Property	Description
company_id	String. Mandatory. The company to which this goods receipt line belongs.
id	String. Mandatory. The ID of this goods receipt line. This value must be unique across all the goods receipts. That means that two different goods receipts must not share the same goods receipt line ID.
line_no	Number. Mandatory. The number of this goods receipt line within the goods receipt.

Property	Description
goods_receipt_date	Date. Mandatory. The date on which this goods receipt line was posted. Format YYYY-MM-DD.
quantity	Number. Mandatory. The quantity received.
purchase_order_line_id	String. The ID of the purchase order line belonging to this goods receipt line.

Surcharge types

Surcharge types are used by d.velop invoices to manually add surcharges to an invoice via the dialog. You can transfer surcharge types as follows:

Request

```
POST /smartinvoice/api/v1/buckets/:bucket_id/surcharge_types /batch
Content-Type: application/json

{
    "surcharge_types": [
        {
            "company_id": "01",
            "nr": "TEST",
            "name": "Test_surcharge",
            "applies_to": "line_item_surcharge",
            "erp_id": "",
            "tenant_id": "",
            "category": "pcs",
            "quantity": "",
            "unit_price": "",
            "gl_account": "",
            "cost_center": "",
            "cost_unit": "",
            "item": "",
            "tax_code_id": "",
            "custom1":"",
            "custom2":"",
            "custom3":"",
            "custom4":"",
            "custom5":"",
            "custom6":"",
            "custom7":"",
            "custom8":"",
            "custom9":"",
            "custom10":"",
            "custom11":"",
            "custom12":"",
            "custom13":"",
            "custom14":"",
            "custom15":"",
            "custom16":"",
            "custom17":"",
            "custom18":"",
            "custom19":"",
            "custom20":"",
            "tax_category":"",
            "procurement_category":""
        }
    ]
}
```

Each individual surcharge type supports the following properties:

Property	Description
company_id	String. The company to which the surcharge type belongs. If you do not specify a company, the cost unit applies for all companies.
nr	String. Mandatory. The ID of the surcharge. The ID must be unique and cannot be left blank.
name	String. Name of the surcharge type.
applies_to	String. Mandatory. Possible values: **header_surcharge**, **line_item_surcharge**
erp_id	String.
tenant_id	String. Mandatory.
category	String. Use the values **pcs**, **fixed**, **percent** if the surcharges are added per piece, at a fixed rate or as a percentage.
quantity	String. Number of the surcharge type.
unit_price	String. Unit price of the surcharge type.
gl_account	String. The G/L account of the surcharge type.
cost_center	String. The cost center of the surcharge type.
cost_unit	String. The cost object of the surcharge type.
item	String. The item number of the surcharge type.
tax_code_id	String. The tax code of the surcharge type.
custom1-20	String. Custom fields of the surcharge type.
tax_category	String. The tax category of the surcharge type.
procurement_category	String. The procurement category of the surcharge type.

Adding and updating individual master data records

You can create and update individual entities via API. This option supplements the existing batch endpoints. Individual master data records are transferred synchronously. This means that master data that changes frequently can be updated in a targeted manner in d.velop smart invoice.

> ### Warning
> API requests must only be made as part of an application interaction (e.g. via a button in the external ERP system). Do not misuse the API, e.g. to synchronously update all accounts payable. Make sure that mass actions in the ERP system do not lead to a large number of API requests. The API also has a rate limit. The HTTP status **429 "Too Many Requests"** means that the rate limit has been reached.

Request

Example of a request to update a single company:

```
PUT /smartinvoice/api/v1/buckets/:bucket_id/companies
Content-Type: application/json

{
    "id": "01",
    "name": "docures AG",
    "parent_id": null,
    "address": "Musterstr. 23",
    "city": "Musterstadt",
    "zip_code": "12345",
    "local_currency": "EUR",
    "country": "DE"
}
```

You can find more information about the properties of the entity under Batch transfer of master data.

Differences compared to the batch endpoints

- You obtain the API endpoint for updating or adding individual master data records of the desired entity by omitting the last path segment **/batch**.
- The HTTP verb is **PUT**.
- In contrast to the batch endpoints, a request adds or updates only one master data record of an entity. The body of the request is not an array and contains the master data record as an object.
- The transferred master data record is created or updated synchronously. Unlike the existing batch endpoints, the response does not return an asynchronously processed job object. You can determine the success or failure of the request directly from the status code of the HTTP response.
- The endpoints for updating individual master data records have a rate limit that must be taken into account when using the API.

Response

```
HTTP/1.1 201 Created
Content-Type: application/json

{
  "status": "successful"
}
```

1.3.5. Transferring mappings between d.velop smart invoice and a SharePoint repository

You want to use a SharePoint repository with d.velop smart invoice. To do so, assign the fields for the sources **Document** and **Workflow protocol** from d.velop smart invoice to the corresponding fields in the configured SharePoint repository.

You can transfer mappings to the SharePoint repositories as follows via API:

Request

```
POST /smartinvoice/api/v1/repositories/sharepoint/:repository_id/mappings

Content-Type: application/json

{
    "voucher_mapping": {
        "attributimport_id": "string",
        "barcode": "string",
        "document_type": "string",
        "order_numbers": "string",
        "order_number": "string",
        "bic": "string",
        "gross_amount": "string",
        "posting_period": "string",
        "posting_text": "string",
        "custom1-20": "string",
        "date_of_supply": "string",
        "delivery_slip_numbers": "string",
        "doc_id": "string",
        "first_step_recipient": "string",
        "external_invoice_nr": "string",
        "responsible": "string",
        "iban": "string",
        "import_calculative_check": "string",
        "internal_invoice_nr": "string",
        "account_holder": "string",
        "vendor_name": "string",
        "vendor_nr": "string",
        "line_item_cost_centers": "string",
        "line_item_cost_units": "string",
        "line_item_custom1-20": "string",
        "line_item_gl_accounts": "string",
        "company_name": "string",
        "company_nr": "string",
        "vat": "string",
        "net_amount": "string",
        "city": "string",
        "posting_date": "string",
        "posting_date_german_string": "string",
        "invoice_date": "string",
        "sp_title": "string",
        "vba_bic": "string",
        "vba_iban": "string",
        "vba_id": "string",
        "currency": "string",
        "workflow_status": "string",
        "pay_amount": "string"
    },
    "summary_mapping": {
        "company_name": "string",
        "company_nr": "string",
        "custom1-20": "string",
        "external_invoice_nr": "string",
        "internal_invoice_nr": "string",
        "vendor_name": "string",
        "vendor_nr": "string",
        "workflow_doc_id": "string"
    }
}
```

You can transfer the following mapping objects:

Property	Description
voucher_mapping	Object. Mapping for the document.
summary_mapping	Object. Mapping for the workflow protocol.

For the properties of the mapping objects, enter the SharePoint field to which you want to assign the field from d.velop smart invoice.

Response

Status code
204 – Successfully updated mappings

Status code
400 – Bad request
403 – Forbidden
404 – Not found

If the mapping could not be updated, you will receive the following response:

Response

```
{
    "code": 400,
    "id": "invalid_format",
    "message": {
        "type": "SmartInvoice::Error",
        "title": "Error",
        "detail": "Mapping could not be updated",
        "metadata": {
            "details": {
                "voucher_mapping": {
                    "non_customizable": [
                        "doc_id",
                        "sp_title"
                    ],
                    "missing_fields": [
                        "brutto_betrag"
                    ],
                    "unknown_columns": [],
                    "mismatching_types": [
                        "barcode",
                        "belegart",
                        "bestellnummer",
                        "bestellnummern",
                        "bic"
                    ]
                }
            },
            "timestamp": "2024-12-05 13:23:50 UTC"
        }
    }
}
```

The response contains the following properties:

Property	Description
metadata.details.<mapping object>	Object. Mapping objects with missing information.
non_customizable	Array. Missing, predefined mappings between d.velop smart invoice and SharePoint.
missing_fields	Array. Missing mandatory fields.
unknown_columns	Array. Fields for which the corresponding field mapping does not exist in SharePoint.
mismatching_types	Array. Fields for which the data type of the field in d.velop smart invoice does not match the data type in SharePoint.

1.3.6. Determining available master data buckets

You can determine the master data buckets that are available in d.velop smart invoice via API.

Request

```
GET /smartinvoice/api/v1/buckets
Accept: application/json
```

Response

```
{
    "_links": {
        "next": {
            "href": "https://test.d-velop.cloud/smartinvoice/api/v1/
buckets/..."
        },
        "previous": {
            "href": "https://test.d-velop.cloud/smartinvoice/api/v1/
buckets/..."
        }
    },
    "buckets": [
        {
            "id": 1473,
            "name": "Stammdaten"
        },
        {
            "id": 1548,
            "name": "Zusatzdaten"
        }
    ]
}
```

Property	Description
_links	Buckets may be returned in multiple pages. Use the specified URLs to retrieve additional pages. • **next**: Reference to the next page. • **previous**: Reference to the previous page.
buckets	Array with the available buckets in d.velop smart invoice • **id**: ID of the bucket. • **name**: Description of the bucket.