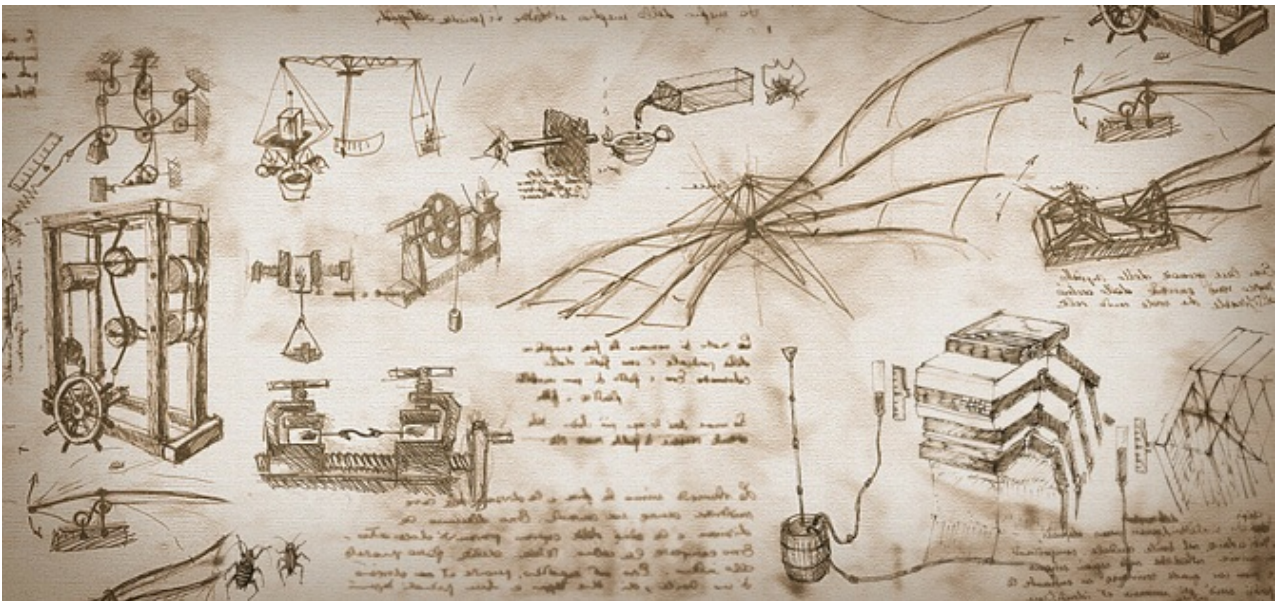


für software, die passt



## Projekt: Vetdata

## Interface Specification Supplier invoices



Version:	1.5
Autor:	Roger Pfister + Markus Studer, nowhow solutions AG
Datum:	11.02.2021
Generiert am:	11 Feb 2021 - 07:19:47
Generiert von:	<a href="http://kmudata.nowhow.ch/">http://kmudata.nowhow.ch/</a>
Copyright:	© 2021 nowhow solutions AG

# Inhalt

<b>1</b>	<b>Specification of invoice delivery to the Vetdata platform</b>	<b>3</b>
1.1	Change Log	3
1.2	Introduction	4
1.3	Intended purpose of this document	4
1.4	Schedule	4
<b>2</b>	<b>Usage of webservice</b>	<b>5</b>
2.1	Scenarios	5
2.2	How to send documents (supplier invoices) to the Vetdata platform	5
<b>3</b>	<b>Technical information</b>	<b>7</b>
3.1	WSDL	7
3.2	XML schema (XSD)	7
3.3	Sample file	7
3.4	Authentication	7
3.5	Available systems	7
<b>4</b>	<b>Methods</b>	<b>8</b>
4.1	Method SubmitInvoice	8
4.2	Method SubmitInvoicePdf	9
4.3	Method SubmitInvoiceXml	10
4.4	Error/Exceptionhandling	11
<b>5</b>	<b>Supplier invoice format "Vetdata"</b>	<b>13</b>
5.1	Logical content of an invoice to Vetdata	13
5.2	Information on customer	14
5.3	General information	14
5.4	Payment information	15
5.5	VAT information per VAT rate	15
5.6	Total invoice amounts	16
5.7	Line item (detail per article)	17
<b>6</b>	<b>Supported units of measurements</b>	<b>18</b>
<b>7</b>	<b>Appendix: SAP/IDoc format</b>	<b>21</b>

# 1 Specification of invoice delivery to the Vetdata platform

## 1.1 Change Log

Version	Date	Changes	Author
1.5	11.02.2021	<ul style="list-style-type: none"> <li>- Additional SAP/IDoc format available (separate appendix to this document)</li> <li>- Attribute "invoiced quantity" (LI10) as decimal (instead of integer)</li> <li>- minor enhancements in schema <code>vetdata_1.0.xsd</code></li> <li>- Validation enhancements: <ul style="list-style-type: none"> <li>- <code>error.invoice.schema_not_supported</code></li> <li>- <code>error.invoice.tvs_customer_is_prohibited_or_deleted</code></li> <li>- <code>error.invoice.document_issue_date_older_than_two_months</code></li> </ul> </li> </ul>	Roger Pfister, Theo Pfaff
1.4	02.12.2020	Documentation corrections and enhancements: <ul style="list-style-type: none"> <li>- new endpoints</li> <li>- Hash values (of XML and PDF data) has to be provided as <u>hexadecimal</u> string</li> <li>- the attributes LI11, LI13, LI14, LI15, TI04 can have a positive or negative value</li> <li>- the attribute TI03 <code>total tax amount payable</code> is optional</li> <li>- Rounding explanations in "Total invoice amounts"</li> <li>- Error Code <code>error.invoice.supplier_number_mismatch</code> removed</li> <li>- new sample file</li> <li>- new schema <code>vetdata_1.0.xsd</code></li> </ul>	Roger Pfister, Theo Pfaff
1.2 - 1.3		nowhow-internal revisions	nowhow
1.1	21.08.2020	Documentation enhancements: <ul style="list-style-type: none"> <li>- Authentication and webservice systems / endpoints</li> <li>- WSDL, XSD, sample file</li> <li>- additional ID per attribute of the XML file</li> <li>- additional annotations in the according XSD file</li> <li>- Error handling new list of "SOAP fault codes"</li> </ul>	Markus Studer, Roger Pfister, Theo Pfaff
1.0	10.06.2020	Valid version (sent to suppliers)	Roger Pfister
0.9	05.06.2020	PDF/A, Formattings	Roger Pfister
0.1	26.05.2020	Initial version created	Markus Studer

---

## 1.2 Introduction

Vetdata is a platform for veterinarians providing online space for document storage. Documents are stored supporting archive requirements required by Swiss law.

The Vetdata application will contain supplier invoices which will be available to the veterinarians with all the detail data.

## 1.3 Intended purpose of this document

This specification defines the technical interface (SOAP) to be used by suppliers to transmit invoice data (PDF and XML data) to the Vetdata platform.

- PDF file: must be the (printable) invoice document using the PDF/A format
- XML file: must contain detailed invoice data according the attached XSD schema, described in detail in
  - chap 5 - Supplier invoice format "VETdata": this is an own, generally usable format
  - Appendix - Supplier invoice format "SAP/IDoc": this format is based on the SAP/IDoc format

## 1.4 Schedule

- June 2020: Interface specification (1st version of this document) sent to suppliers [done]
- August 2020: WSDL (of this interface) will be available [done]
- September 2020: a test system will be available [done]
- March 2021: Start of operation of "Vetdata" [according to further individual information to suppliers ready to use this interface]

## 2 Usage of webservice

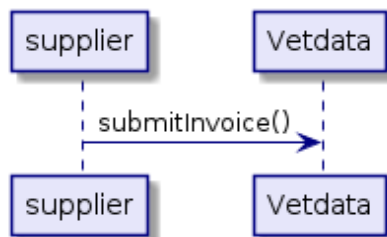
### 2.1 Scenarios

There are two usage scenarios to transmit the invoice data:

- Both XML data and the PDF document in one SOAP request
- Separate SOAP request for XML data and the PDF document

#### Both XML data and the PDF document in one SOAP request

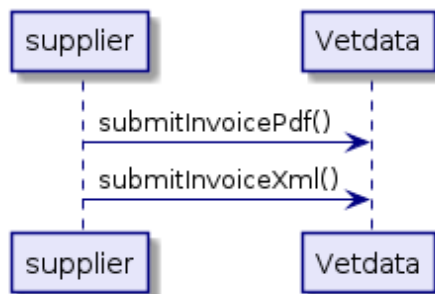
If both the XML data and the PDF document are available the data is transmitted with the method `submitInvoice`



An invoice is considered "accepted" by the Vetdata platform if this call returns a `resultInfo` of `completed`.

#### Separate SOAP request for XML data and the PDF document

If XML data and corresponding PDF document are created in different stages of a billing process, those information can be transmitted with separate calls to `submitInvoicePdf` and `submitInvoiceXml`



The order of those two calls doesn't matter. Important: An invoice is only considered "accepted" by the Vetdata platform if the second of those calls returns a `resultInfo` of `completed`.

To verify the transfers of the documents (supplier invoices), the transfer of the documents is secured by SHA-256 hashes, i.e. the supplier calculates a SHA-256 hash of the document (PDF or XML data) and transmits this information together with the corresponding document. The Vetdata platform calculates a SHA-256 hash of the document received and verifies the SHA-256 hash. If a mismatch is detected the invoice is rejected (Important: Hashes are calculated on the original files, not the base64 encoded string representation of those files).

### 2.2 How to send documents (supplier invoices) to the Vetdata platform

PDF file:

1. create the file in PDF/A format
2. calculate the SHA-256 hash (hex value) of this (unencoded) PDF/A file
3. create the base64 encoded byte string of this PDF/A file
4. submit the data using the SOAP request method "submitInvoice" (if you sent PDF and XML files together) or "submitInvoicePdf" (if you want to send the PDF file separately)

XML file:

1. create the file in XML format according the rules of the respective schema (see attached Vetdata XSD file)

- 
2. calculate the SHA-256 hash (hex value) of this (unencoded) XML file
  3. create the base64 encoded byte string of this XML file
  4. submit the data using the SOAP request method "submitInvoice" (if you sent XML and PDF files together) or "submitInvoiceXml" (if you want to send the XML file separately)

---

## 3 Technical information

### 3.1 WSDL

The attached WSDL has to be used to call this webservice using the methods described in chap. 4 "Methods"

- [SupplierInvoice.wsdl](#)

### 3.2 XML schema (XSD)

The attached schema (XSD) can be used to validate the XML representation of the supplier invoices / credit notes to be sent to Vetdata. The annotations in this schema refer to the attribute description you can find in chap 5 "Supplier invoice format".

- [vetdata\\_1.0.xsd](#)

### 3.3 Sample file

The attached sample file represents a (fictive) supplier invoice (XML format) and matches the above schema (XSD).

- [Sample-Vetdata-Invoice.xml](#)

### 3.4 Authentication

Access to the Soap Methods is protected by "HTTP basic authentication". Accounts for suppliers (User names and passwords) are provided by Vetdata upon request.

Technical limitation: The "Authorization" HTTP Header has to be present with every request (Authenticate pre-emptively). A missing "Authorization" header will result in a SoapFault and not a HTTP Status Code "401 - Unauthorized".

### 3.5 Available systems

#### 3.5.1 Test system

- Purpose: testing environment, none of the received files will be processed further (will not be sent to or visible for the customers defined on this system, no financial transactions will be triggered)
- Data: all actually known TVS suppliers and TVS customers will be addressable with their real identifications (TVS numbers) and names, so you can test with real customers (and real or fictive invoices / credit notes)
- Endpoint: <https://apptest.vetdata.ch/SupplierInvoice.asmx>
- Availability: operating since October 2020; the system will be up and running 7 x 24 hours but interruptions / downtimes (e.g. for new deployments of the application) can occur at any time without further notice

#### 3.5.2 Production system

- Purpose: productive operation, all received files will be processed by the Vetdata and TVS application
- Data: all known and active TVS suppliers and TVS customers will be online when starting this new Vetdata application
- Endpoint: <https://app.vetdata.ch/SupplierInvoice.asmx>
- Availability: planned start end of January 2021; the system will be up and running 7 x 24 hours (except announced down time for maintenance work)

## 4 Methods

### 4.1 Method SubmitInvoice

#### SubmitInvoice

Submits an invoice to Vetdata. Vetdata validates the information received. Soap faults are used to report problems. An invoice is considered successfully transferred if both documents (xml and pdf) are transferred and no soap faults are encountered.

#### 4.1.1 Input / Request

Parameter	Description	Datatype	Example / Remarks
TVSCustomerNumber	The four digit TVS customer number (in range 1000-9999)	xs:integer	e.g. 1234
documentIdentifier	Sequential number assigned by the biller, which uniquely identifies the invoice / credit note. Max 12 digits. (in range 0-999999999999)	xs:nonNegativeInteger	e.g. 986454321
documentIssueDate	ISO 8601 (YYYY-MM-DD)	xs:date	e.g. 2019-06-20
xmlDataSchema	Vetdata schema used to transfer the billing information	xs:string	e.g. 'vetdata_1.0.xsd'
xmlData	Full XML data with the billing information (UTF-8) encoded as base64 string	xs:string	1)
xmlDataSha256Hash	SHA-256 hash of Full XML data, represented as hexadecimal string	xs:string	1)
pdfData	PDF/A (at least PDF/A-1b) document encoded as base64 string	xs:string	1)
pdfDataSha256Hash	SHA-256 hash of PDF/A document, represented as hexadecimal string	xs:string	1)

<sup>1)</sup> Vetdata needs both the xml document and the PDF document of an invoice. It is possible to submit the PDF and XML documents with separate calls to [SubmitInvoicePdf](#) and [SubmitInvoiceXml](#). Important: an invoice is only deemed successfully transferred if both documents (xml, pdf) are successfully transferred to Vetdata.

#### 4.1.2 Output / Response

Parameter	Description	Datatype	Example / Remarks
resultInfo	Result of submit	xs:string	Always 'completed' (as xml and pdf are transmitted in one call)



## 4.2 Method SubmitInvoicePdf

### SubmitInvoicePdf

Submits the PDF document of an invoice.

#### 4.2.1 Input / Request

Parameter	Description	Datatype	Example / Remarks
TVSCustomerNumber	The four digit TVS customer number (in range 1000-9999)	xs:integer	e.g. 1234
documentIdentifier	Sequential number assigned by the biller, which uniquely identifies the invoice / credit note. Max 12 digits. (in range 0-999999999999)	xs:nonNegativeInteger	e.g. 986454321
documentIssueDate	ISO 8601 (YYYY-MM-DD)	xs:date	e.g. 2019-06-20
pdfData	PDF/A (at least PDF/A-1b) document encoded as base64 string	xs:string	
pdfDataSha256Hash	SHA-256 hash of PDF/A document, represented as hexadecimal string	xs:string	

#### 4.2.2 Output / Response

Parameter	Description	Datatype	Example / Remarks
resultInfo	Result of submit	xs:string	'partially submitted' or 'completed'

### 4.3 Method SubmitInvoiceXml

#### SubmitInvoiceXml

Submits the XML document of an invoice.

#### 4.3.1 Input / Request

Parameter	Description	Datatype	Example / Remarks
TVSCustomerNumber	The four digit TVS customer number (in range 1000-9999)	xs:integer	e.g. 1234
documentIdentifier	Sequential number assigned by the biller, which uniquely identifies the invoice / credit note (max 12 digits in range 0-999999999999)	xs:nonNegativeInteger	e.g. 986454321
documentIssueDate	ISO 8601 (YYYY-MM-DD)	xs:date	e.g. 2019-06-20
xmlDataSchema	Vetdata schema used to transfer the billing information	xs:string	e.g. 'vetdata_1.0.xsd'
xmlData	Full XML data with the billing information (UTF-8) encoded as base64 string	xs:string	
xmlDataSha256Hash	SHA-256 hash of full XML data, represented as hexadecimal string	xs:string	

#### Remarks

Vetdata needs both the xml document and the PDF document of an invoice. It is possible to submit the PDF and XML documents with separate calls to [SubmitInvoicePdf](#) and [SubmitInvoiceXml](#). Important: an invoice is only deemed successfully transferred if both documents (xml, pdf) are successfully submitted to Vetdata.

#### 4.3.2 Output / Response

Parameter	Description	Datatype	Example / Remarks
resultInfo	Result of submit	xs:string	'partially submitted' or 'completed'

#### 4.4 Error/Exceptionhandling

Error/Exceptionhandling is implemented according to SOAP standards. The following types of errors are defined:

Issuer	Description
.net Framework	Errors generated from the .net plattform itself, e.g. request validation fails according to the defined WSDL
Application	Errors through invalid data in requests or application faults result in a soap fault with error information

##### 4.4.1 SOAP fault codes

faultcode	Description
error.invoice.could_not_process_xml	Could not process invoice xml data.
error.invoice.error_parsing_root_data	Problem parsing invoice root item in xml data.
error.invoice.error_processing_line_item	Problem parsing invoice line item in xml data.
error.invoice.error_processing_vat_information	Problem parsing invoice vat information in xml data.
error.invoice.invoice_already_accepted	Invoice was already accepted by the vetdata system. It is not possible to submit an invoice multiple times.
error.invoice.document_issue_is_in_future	The invoice issue date must not be in the future. <b>NEW</b>
error.invoice.document_issue_date_older_than_twelve_months	The invoice issue date must not be older than 12 months. <b>NEW</b>
error.invoice.mismatch_document_identifier_in_call_and_xml	Document identifier in soap call and info in invoice xml do not match.
error.invoice.mismatch_sum_amount_exclusive_vat_plus_tax	Mismatch between VATInformation charges and TotalInvoiceAmounts.
error.invoice.mismatch_sum_line_items_and_total	
error.invoice.mismatch_tvs_customer_number_in_call_and_xml	TVS customer number in soap call and info in invoice xml do not match.
error.invoice.mismatch_tvs_supplier_number_in_call_and_xml	TVS supplier number in soap call and info in invoice xml do not match.
error.invoice.only_swiss_francs_as_currency_supported	
error.invoice.pdf_document_already_accepted	PDF document was already accepted by the vetdata system. It is not possible to submit a pdf document multiple times.
error.invoice.problem_adding_to_search_index	
error.invoice.problem_with_search_index_deletion	
error.invoice.problem_with_storage_deletion	
error.invoice.schema_not_supported	Specified schema is not supported by Vetdata. <b>NEW</b>

faultcode	Description
error.invoice.schema_validation_failed	Schema validation of received invoice xml failed.
error.invoice.storage_and_sha_256_hash_mismatch_problem	Received SHA-256 signature does not match received data.
error.invoice.supplier_marked_as_deleted	
error.invoice.supplier_not_found	
error.invoice.tvb_customer_is_prohibited_or_deleted	TVS Customer is not allowed to receive invoices payable by TVS or is deleted. <b>NEW</b>
error.invoice.tvb_customer_not_found	
error.invoice.tvb_customer_zip_code_not_allowed_for_invoice	Received zipcode of customer in xml is not allowed for invoices. Each customer in TVS has one or more zipcodes that are allowed for invoices.
error.invoice.unknown_document_type_received	
error.invoice.xml_document_already_accepted	XML document was already accepted by the vetdata system. It is not possible to submit an xml document multiple times.


## 5 Supplier invoice format "Vetdata"

Invoices are provided by suppliers. The data must be delivered in a structured form (XML) as well as a PDF/A (for display).


### 5.1 Logical content of an invoice to Vetdata

The content of a structured invoice for Vetdata is described below. Ideas of the content come from <https://swissdigin.gs1.ch/home/standard/inhaltsstandard> This defines the content, but not the form.

Fields marked with  are fields which are not provided in Swissdigin but are required for Vetdata.

ID	Attribute	Format	opt/mand	Description
IB01	document type	380 / 381	mandatory	Describes the document type: '380' for invoice, '381' for credit note
IB02	document identifier	Integer (max 999'999'999'999)	mandatory	Sequential number assigned by the biller, which uniquely identifies the invoice / credit note (max 12 digits)
IB03	document issue date	Date	mandatory	ISO 8601 (YYYY-MM-DD, e.g. 2019-06-20), must not be older than 12 months and must not be in the future <span style="color: green;">UPDATED</span>
IB04	document currency	String(3)	mandatory	currency in ISO-Standard 4217, at the moment only 'CHF' is accepted
IB05	 TVS supplier number	Integer	mandatory	three-digit number
IB06	tax identification number	String(20)	mandatory	VAT number under which the biller has carried out the delivery resp. has rendered the service. The format for the electronic data transmission is without suffix (example): 'CHE123456789'
IB07	legal name of company	String(255)	mandatory	Name and adress have to comply with the data in the commercial or tax register.
IB08	address1	String(255)	mandatory	Freetext
IB08	address2	String(255)	optional	Freetext
IB08	address3	String(255)	optional	Freetext
IB09	city name	String(255)	mandatory	Freetext
IB10	zip/postal code	Integer	mandatory	Number
IB11	country	String(2)	mandatory	country code in ISO-Standard 3166-1 (e.g. 'CH', 'DE', etc.).


## 5.2 Information on customer

ID	Attribute	Format	opt/mand	Description
IC01	 TVS customer number	Integer	mandatory	The four-digit TVS customer number
IC02	customer number in seller system	String(255)	mandatory	Freetext
IC03	legal name of company	String(255)	mandatory	Name and address have to comply with the data in the commercial or tax register
IC04	address1	String(255)	mandatory	Freetext
IC04	address2	String(255)	optional	Freetext
IC04	address3	String(255)	optional	Freetext
IC05	city name	String(255)	mandatory	Freetext
IC06	zip/postal code	Integer	mandatory	Number
IC07	country	String(2)	mandatory	country code in ISO-Standard 3166-1 (e.g. 'CH', 'DE', etc.).

## 5.3 General information

ID	Attribute	Format	opt/mand	Description
GI01	date of delivery	Date	mandatory	ISO 8601 (YYYY-MM-DD, e.g. 2019-06-20)
GI02	order reference of customer	String(255)	optional	Freetext, e.g. Vetpoint order number
GI03	order date	Date	optional	ISO 8601 (YYYY-MM-DD, e.g. 2019-06-20)

#### 5.4 Payment information

ID	Attribute	Format	opt/mand	Description
PI01	 payable by TVS	Boolean	mandatory	'True' if the invoice has to be paid by TVS, 'False' otherwise
PI02	payment terms	String(600)	optional	Free text element
PI03	payment period	Integer	optional	Number of days within which the invoice has to be settled
PI04	due date	Date	optional	ISO 8601 (YYYY-MM-DD, e.g. 2019-06-20)
PI05	settlement discount rate	Decimal	optional	Percentage rate of settlement discount (between 0.00 and 100.00)
PI06	due date of settlement discount	Date	optional	Date until which it is allowed to deduct the stipulated settlement discount rate. Format: ISO 8601 (YYYY-MM-DD, e.g. 2019-06-20)
PI07	payment identifier	String(255)	optional	In Switzerland, normally the ESR reference number is used. It is provided by the supplier to automatically assign the incoming payment to the corresponding invoice. In the environment of SEPA, this reference is called "Structured Creditor Reference"
PI08	ESR party identifier	String(20)	optional	ESR party number used for Swiss transactions for the steering of payments over the Swiss Post
PI09	financial institution identifier	String(20)	optional	Usually BIC (Bank Identifier Code)
PI10	financial institution name	String(255)	optional	Additional information in case financial institution identifier is not correct
PI11	IBAN number	String(255)	optional	Standard number used to uniquely identify the account to be credited

If "payable by TVS" is true, all other attributes are ignored.

If "payable by TVS" is false, other attributes must be filled so that a payment can be made.

#### 5.5 VAT information per VAT rate

ID	Attribute	Format	opt/mand	Description
VI01	VAT percentage rate	Decimal	mandatory	Applied VAT percentage rates, multiple rates per invoice are allowed (amount of different VAT percentage rates = n)
VI02	taxable amount	Decimal	mandatory	Equivalent to the total of all line items "total amounts exclusive VAT" (sum LI13) referring to the VI01 value
VI03	tax amount payable	Decimal	mandatory	Equivalent to the (not rounded) total of all line items "VAT amount payable" (sum LI15) referring to the VI01 value

## 5.6 Total invoice amounts

ID	Attribute	Format	opt/mand	Description
TI01	total amount of invoice excl. VAT	Decimal	mandatory	Equivalent to the total of all line items "total amounts exclusive VAT" (sum LI13, or sum VI02)
TI02	total amount of invoice incl. VAT	Decimal	mandatory	Equivalent to the total of all line items "total amounts inclusive VAT" (sum LI14, or sum VI03) <sup>2)</sup>
TI03	total tax amount payable	Decimal	optional	Equivalent to the total of all line items "tax amount payable" (sum LI15). also equivalent of all "tax amount payable" (sum VI03)
TI04	rounding amount	Decimal <sup>1)</sup>	mandatory	Rounding difference caused by the rounding to the next 5 cents <sup>2)</sup>






<sup>1)</sup> also a negative value is allowed

### <sup>2)</sup> Rounding explanations

- in the line items the values "total amount incl. VAT" (LI14), "VAT amount payable" (LI15) and probably also total amount excl. VAT (LI13) should represent an amount with two digits after the decimal point (rounding to the second digit after the decimal point, but no rounding to the next 5 cents)
- "total amount of invoice excl. VAT" (TI01, sum of all LI13), "total tax amount payable" (TI03, sum of all LI15) and "tax amount payable" (VI03, the sum of all LI15 with the same VAT rate) should not be rounded to the next 5 cents
- "total amount of invoice incl. VAT" (TI02, sum of all LI14) is the only amount that can be rounded to the next 5 cents; but no such rounding is also acceptable at all
- "rounding amount" (TI04) must represent the difference between the sum of TI01 + TI03 and TI02, if TI03 is not provided (optional attribute), the sum of all VI03 has to be taken); the application is performing the following check: "total amount of invoice incl. VAT" (TI02) - "total amount of invoice excl. VAT" (TI01) - sum of "tax amount payable" (VI03) = "rounding amount" (TI04) ?
- see also the attached sample file for an example of correct rounding



## 5.7 Line item (detail per article)

ID	Attribute	Format	opt/mand	Description
LI01	article number of supplier	String(255)	mandatory	Freetext
LI02	 article number Vetpoint	String(20)	mandatory if article exists in Vetpoint	Format: [3 digit Vetpoint supplier number]-[article number vetpoint] e.g. '999-VET988756'
LI03	GTIN/EAN	String(255)	optional	<a href="https://de.wikipedia.org/wiki/Global_Trade_Item_Number">https://de.wikipedia.org/wiki/Global_Trade_Item_Number</a>
LI04	description of goods supplied or service rendered	String(600)	mandatory	Freetext
LI05	unit of measurement	String(3)	mandatory	ISO codes for units of measurement. See the separate chapter with list of supported values
LI06	 expiry date	Date	mandatory, if applicable on article	The expiry date of this article ISO 8601 (YYYY-MM-DD, e.g. 2019-06-20)
LI07	 batch# / LOT#	String(255)	optional	Freetext
LI08	 Swissmedic Number	String(20)	mandatory, if applicable on article	Swissmedic-Nr. mandatory for a drug with format [999999-99-999] or [VAK-999999-00-999]
LI09	 ATCVet Code	String(20)	mandatory, if applicable on article	The ATCVet code of the article, mandatory for a drug. A veterinary medicine article starts with 'Q' followed by 1 capital letter, 2 digits and optional 1-2 capital letters or 2 capital letters and two digits, human medicine article without the starting 'Q' <span style="color: green;">UPDATED</span>
LI10	invoiced quantity	Decimal <span style="color: green;">UPDATED</span>	mandatory	number or decimal value
LI11	price amount excl. VAT	Decimal <sup>1)</sup>	mandatory	price per unit of measurement
LI12	VAT percentage rate	Decimal	mandatory	
LI13	total amount excl. VAT	Decimal <sup>1)</sup>	mandatory	Invoiced total amount exclusive VAT applying to the line item
LI14	total amount incl. VAT	Decimal <sup>1)</sup>	optional	Invoiced total amount of the line item inclusive VAT after settlement of all allowances and charges applying to the line item
LI15	VAT amount payable	Decimal <sup>1)</sup>	mandatory	Total VAT amount payable of the line item

<sup>1)</sup> also negative values are allowed if the line item represents a discount, a prepayment or returned / refunded item

## 6 Supported units of measurements

4G	Mikroliter	microlitre
AA	Knäuel	ball
BE	Gebinde	bundle
BG	Beutel	bag
BJ	Eimer	bucket
BL	Ballen	bale
BLI	Blister	blister
BND	Bündel	bundle
BO	Flasche	bottle
BX	Box	box
C4	Karte	card
C5	Dienstleistung	service
C62	Stück	piece
CA	Dose	can
CG	Karton	carton
CH	Container	container
CL	Bund	coil
CLT	Zentiliter	centilitre
CMK	Quadratcentimeter	square centimetre
CMQ	Kubikcentimeter	cubic centimetre
CMT	Zentimeter	centimetre
CQ	Kartusche	cartridge
CS	Etui	case
CT	Kartonage	carton
CV	Hülle	cover
D64	Block	block
D97	Palette	pallet
DAY	Tag	day
DI	Behälter	dispenser
DPC	Dutzend	dozen
DZN	Dutzend	dozen
EA	Stück	piece
FOI	Folie	foil
GRM	Gramm	gram
HUR	Stunde	hour

JR	Glas	jar
KGM	Kilogramm	kilogram
KIS	Kiste	crate
KMT	Kilometer	kilometre
KST	Kanister	jerrycan
KT	Kit	kit
LBR	Pfund	pound
LTR	Liter	liter
M4	Wert	value
MC	Mikrogramm	microgram
MGM	Milligramm	milligram
MLT	Milliliter	millilitre
MMT	Millimeter	millimetre
MTK	Quadratmeter	square metre
MTQ	Kubikmeter	cubic metre
MTR	Meter	metre
NMO	Nanomol	nanomole
OP	2er Pack	pack of 2
PA	Packung	packet
PAD	Tupfer	pad
PCE <small>NEW</small>	Stück	piece
PD	Block	block
PF	Palette	pallet
PK	Paket	pack
PR	Paar	pair
QR	Bogen	quire
RD	Spindel	rod
RL	Spule	reel
RM	Ries	ream
RO	Rolle	roll
SA	Sack	sack
SCH	Schachtel	box
SET	Satz	set
SPD	Spender	dispenser
SR	Streifen	strip
ST	Blatt	sheet
TAB	Tab	tab

---

TC	LKW-Ladung	truckload
TN	Dose	tin
TNE	Tonne (metrisch)	ton
TU	Tube	tube
TUC	Tuch	cloth
WEE	Woche	week
Z3	Fass	cask

---

## 7 Appendix: SAP/IDoc format

This appendix (an analogous description such as chapter 5) is available as a separate document (PDF file).