



File specification

XML format

Unifaun Online, Pacsoft Online, Posti SmartShip

Contents

1	About XML format.....	3
2	Sender.....	4
3	Receiver.....	5
4	Party.....	5
5	Shipment and Container.....	6
5.1	Service.....	7
5.2	Add-on services.....	7
5.3	Container.....	7
6	Miscellaneous elements.....	9
6.1	Meta.....	9
6.2	Pre-notification by e-mail.....	9
6.3	Link to print.....	9
6.4	Consolidation.....	10
7	Valid values.....	10
7.1	Meta.....	10
7.2	Sender.....	11
7.3	Receiver.....	12
7.4	Party.....	13
7.5	Shipment.....	15
7.6	Service.....	17
7.7	Addon.....	17
7.8	Booking.....	19
7.9	Container.....	19
7.10	Options.....	21
7.11	Customs declaration documents UFPS.....	22
7.12	Customs declaration documents Unifaun/Pacsoft Online, Posti SmartShip.....	26
8	Error messages.....	29
8.1	UO and PO.....	30
8.2	UFPS.....	30

1 About XML format

Unifaun Online, Pacsoft Online, Unifaun PrintServer and Posti SmartShip (from now on referred to as UO, PO, UFPS and PSS) can read XML files, which is the recommended format due to scalability and versatility. Also, the XML format provides the best support for ERP-connection.

Acquisition of order files is done by polling a folder, defined by the user, for files with a certain file extension (e.g. *.xml). When a file is found, its structure and contents are validated. There is also the possibility of posting XML files directly to Unifaun Online, Pacsoft Online and Posti SmartShip without using an intermediate application. In case of structural errors, such as missing XML tags, the file will be ignored. If there are errors in the variable data, such as missing information and/or erroneous values, a warning label is printed (for UFPS) or the information is set as invalid in the online interface (for UO, PO and PSS). For help on interpreting these errors, please refer to chapter Error messages.

The order file in XML format normally consists of three basic elements with additional nested elements, all of them explained in detail in this document:

- Sender
 - Partner
- Receiver
 - Partner
- Shipment
 - Service
 - Container

Normally, the following XML declaration is used:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
```

If Unicode is used, one of following options is valid, depending on type:

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<?xml version="1.0" encoding="UTF-16"?>
```

Tip! You can create a sample file by copying lines with XML code and pasting them into an empty text document.

There are characters that can occur in variable data that need to be entered so that the XML parser can interpret them correctly:

& is entered as "&"

< is entered as "<"

> is entered as ">"

' is entered as "'"

" is entered as """

2 Sender

The `sender` element contains sender's information and its occurrence is not always mandatory in a file.

Sender's information can be defined both in the online systems and UFPS. If this is the case, only a referral is needed in order file, pointing to a sender's so called quick ID value. This referral is done in the `shipment` element.

In cases where there are several senders and a static structure is not possible, the information about the sender must be supplied in the order file. The `sender` element consists of `<sender sndid="...">` where `...` is to be replaced with an attribute of your choice with any data type and length. The element must end with `</sender>`.

Values are supplied as `<val n="value name">...</val>`. An index of valid values can be found in chapter Valid values.

Partner (carrier) information must be supplied in those cases senders are not predefined in the online systems or UFPS. This is done by a nested `partner` element with an attribute that specifies the carrier and the element's mandatory values. Index of valid partner attributes can be found in the **HELP > Code Lists > Services** menu in the online systems. For Unifaun PrintServer go to Unifaun Engine Administrator > Partner (general) > Select Partner.

An example of a `sender` element:

```
<sender sndid="1">
    <val n="name">Test Testström</val>
    <val n="address1">Testvägen 1</val>
    <val n="address2">Ingång C</val>
    <val n="zipcode">41118</val>
    <val n="city">GÖTEBORG</val>
    <val n="country">SE</val>
    <val n="contact">Test Testström</val>
    <val n="phone">031-000 00 00</val>
    <val n="fax">031-000 00 00</val>
    <val n="email">test@teststrom.se</val>
    <val n="sms">0700-00 00 00</val>
    <partner parid="PLAB">
        <val n="custno">0123465782</val>
    </partner>
</sender>
```

Tip! You can create a sample file by copying lines with XML code and pasting them into an empty text document.

3 Receiver

The element `receiver` contains receiver's information and its occurrence in a file is mandatory.

The `receiver` element consists of `<receiver rcvid="...">` where `...` is to be replaced with an attribute of your choice with any data type and length. The element must end with `</receiver>`.

Values are supplied as `<val n="value name">...</val>`. An index of valid values can be found in chapter Valid values.

In cases where the shipment contains add-on services that demand further information (e.g. receiver pays, which requires the receiver's customer number at the carriers'), the partner information must be supplied as well. This is done by a nested `partner` element with an attribute that specifies the carrier and the element's mandatory values. An index of valid partner attributes can be found in the **HELP > Code Lists > Services** menu in the online systems. For Unifaun PrintServer go to Unifaun Engine Administrator > Partner (general) > Select Partner.

An example of a `receiver` element:

```
<receiver rcvid="Customer_001">
    <val n="name">Kund Kundström</val>
    <val n="address1">Kundvägen 10A</val>
    <val n="address2">2 trappor</val>
    <val n="zipcode">41480</val>
    <val n="city">GÖTEBORG</val>
    <val n="country">SE</val>
    <val n="contact">Kund Kundström</val>
    <val n="phone">031-000 00 00</val>
    <val n="fax">031-000 00 00</val>
    <val n="email">kund@kundstrom.se</val>
    <val n="sms">0700-00 00 00</val>
</receiver>
```

Tip! You can create a sample file by copying lines with XML code and pasting them into an empty text document.

4 Party

The element `party` contains a neutral party's information and its occurrence in a file is optional.

The `party` element consists of `<party ptyid="...">` where `...` is to be replaced with an attribute of your choice with any data type and length. The element must end with `</party>`.

Values are supplied as `<val n="value name">...</val>`. An index of valid values can be found in chapter Valid values.

In cases where the shipment contains add-on services that demand further information (e.g. other payer, which requires the party's carrier customer number), the partner information must be supplied as well. This is done by a nested `partner` element with an attribute that specifies the carrier and the element's mandatory values. An index of valid partner attributes can be found in the **HELP > Code Lists > Services** menu in the online systems. For Unifaun PrintServer go to Unifaun Engine Administrator > Partner (general) > Select Partner.

An example of a `party` element:

```
<party ptyid="Party_001">
    <val n="name">Freight Payer Company</val>
    <val n="address1">Freight Avenue 10</val>
    <val n="address2">5th floor</val>
    <val n="zipcode">41480</val>
    <val n="city">GÖTEBORG</val>
    <val n="country">SE</val>
    <val n="contact">Some person</val>
    <val n="phone">031-000 00 00</val>
    <val n="fax">031-000 00 00</val>
    <val n="email">any@freightpayer.se</val>
    <val n="sms">0700-00 00 00</val>
</party>
```

Tip! You can create a sample file by copying lines with XML code and pasting them into an empty text document.

5 Shipment and Container

The shipment element defines the shipment itself and its presence in the file is mandatory. Sender, receiver, delivery terms, service type and add-ons are defined in this element.

The `shipment` element consists of `<shipment orderno="...">` where `...` is to be replaced with an attribute of your choice with any data type and length. This attribute is not printed on shipment documents; it is unique for every shipment in a file and is searchable in the online systems.

The two most important values in the shipment element are `<val n="from">...</val>` and `<val n="to">...</val>`.

These values refer to the sender's `sndid` and the receiver's `rcvid` (in our example `1` and `Customer_001`).

As mentioned earlier, sender information can be predefined in the online systems or UFPS. If this is the case, the value `from` should refer to the sender's quick ID in the online systems or UFPS.

Values are supplied as `<val n="value name">...</val>`. An index of valid values can be found in chapter Valid values.

The element must end with `</shipment>`.

5.1 Service

This element supplies the carrier's service code and its presence is mandatory. It's supplied as a nested element under `shipment` as `<service srvid="...">` where `...` corresponds to service code. An index of valid codes can be found in the **HELP > Code Lists > Services (ERP-connection XML)** menu in the online systems. For Unifaun PrintServer go to Unifaun Engine Administrator > Partner (general) > Select Partner.

The element must end with `</service>`.

5.2 Add-on services

As mentioned earlier, add-on services, such as Cash On Delivery and Receiver payment, can be supplied in this element. Add-ons are defined as nested elements to the `service` element as `<addon adnid="...">` where `...` corresponds to the add-on code. An index of valid add-on codes can be found in the **HELP > Code Lists > Services (ERP-connection XML)** menu in the online systems. For Unifaun PrintServer go to Unifaun Engine Administrator > Partner (general) > Select Partner.

Most of the add-ons require additional information to be executed. For instance, Cash On Delivery requires an amount and a payment reference to be supplied. An index of valid values can be found in chapter Valid values.

The element must end with `</addon>`.

5.3 Container

The `container` element defines contents of the shipment, e.g. number of parcels, weight and volume. This element is also nested in the `shipment` element. Values are supplied as `<val n="value name">...</val>`. An index of valid values can be found in chapter Valid values.

Parcel information can be defined in several different ways:

```
<container type="parcel">
```

Defines information for each parcel individually.

```
<container type="parcel" measure="totals">
```

Defines information for an entire row of parcels.

```
<container type="parcel" partorderno="Part_Ordernumber_1">
```

Defines an order number for the parcel. Any value.

```
<container type="totals" measure="totals">
```

Defines information for the entire shipment.

Example 1:

```
<container type="parcel">
    <val n="copies">2</val>
    <val n="weight">10</val>
    <val n="packagecode">PC</val>
</container>
```

This example shows that the shipment contains two parcels and the total weight is 20 kg (2 parcels x 10 kg).

Example 2:

```
<container type="parcel" measure="totals">
    <val n="copies">2</val>
    <val n="weight">10</val>
    <val n="packagecode">PC</val>
</container>
```

This example shows that the shipment contains two parcels and the total weight is 10 kg. The individual parcel weight is only assumed to be 5 kg each. Please note that the assumed individual parcel weight will not be printed on shipping documents.

An example of a shipment element:

```
<shipment orderno="Ordernumber_1">
    <val n="from">1</val>
    <val n="to">Customer_001</val>
    <val n="reference">Shipment reference</val>
    <val n="freetext1">Text of your choice</val>
    <service srvid="P15">
    </service>
    <container type="parcel">
        <val n="copies">1</val>
        <val n="weight">10</val>
        <val n="contents">Stuff</val>
    </container>
</shipment>
```

Tip! You can create a sample file by copying lines with XML code and pasting them into an empty text document.

6 Miscellaneous elements

There is several non-mandatory elements that can be supplied in if extended functionality is desired.

6.1 Meta

This element is usually inserted first in a file, directly after the declaration of the root element. Meta can contain printer mapping, mapping to a certain print favorite etc. Values are supplied as `<val n="value name">...</val>`. An index of valid values can be found in chapter Valid values.

6.2 Pre-notification by e-mail

Unifaun offers a service called **Pre-notification by e-mail**. When this service is used, an e-mail message is sent to a defined recipient with information about the shipment together with a tracking link containing shipment/parcel ID. An example of this e-mail is shown here:

<http://www.unifaun.se/download/prenot-en.jpg>

This service is declared as standalone element nested under the `shipment` element. Values are supplied as `<val n="value name">...</val>`. An index of valid values can be found in chapter Valid values.

```
<ufonline>
    <option optid="enot">
        <val n="value name">...</val>
    </option>
</ufonline>
```

6.3 Link to print

The service can send an e-mail message to a recipient containing a link from which the recipient can print shipping documents at his/hers convenience. Link to print can be used for both normal and return shipments.

This service is declared as standalone element nested under the `shipment` element. Link to print can also contain the attribute `linkprintkey` which is added in the `shipment` element's tag with a unique value of any data type and length. This is used if the mail to the recipient containing the link is sent from a third party system. The system will generate random key if none is specified. Example:

```
<shipment orderno="Ordernummer_1" linkprintkey="aFg3rA">
```

This key is a part of the link that is sent to the recipient. The link would look like this, according to the example above:

<https://www.unifaunonline.com/ext.uo.se.se.linkprint?key=<user-ID>&job=aFg3rA&mode=normal>

Due to the key being visible in the link, it is important that you do not use simple keys such as number sequence or similar. The recipient could change the link and print other recipient's shipping documents. Do not use empty spaces or underline characters in the link.

6.4 Consolidation

Consolidation feature permits continuous printing of shipping labels while the EDI message is held until the shipment is closed. A typical use case is if you have multiple orders to a certain receiver during the day and you wish to consolidate these into one single shipment.

The service is declared as a standalone element nested under the `shipment` element. Consolidation service requires the `mergeid` attribute in shipment element tag. Example:

```
<shipment orderno="Ordernummer_1" mergeid="MRG001">
```

`Mergeid` can be an alphanumeric value of your choice and should be unique for each shipment. Do not use empty spaces or underline characters. Consolidated shipments need to be closed, either manually in UO interface or by supplying a closing XML file containing a control element directly under the root element:

```
<control>
  <close>
    <val n="mergeid">...</val>
    <val n="printer">...</val>
  </close>
</control>
```

Closing a consolidated shipment triggers waybill and eventual customs declaration documents printout to preset laser printer if no printer is specified in the `<val n="printer">` tag.

Tip! You can create a sample file by copying lines with XML code and pasting them into an empty text document.

7 Valid values

7.1 Meta

Declared under the `root` element `<data>`.

<code><meta></code>	
<code><val n="printer"></code>	Sends the print job to defined name of printer mapping.
<code><val n="favorite"></code>	Defines the print favourite in the online systems which is used to auto-complete the order file if necessary.
<code><val n="autoprint"></code>	Determines if the order file is to be sent directly to the print queue. If using XML-Posting this can only be used with Link-to-print, not to print physical labels. Valid values: YES, NO

<val n="test">	Used for order file testing. Prints out TEST on label and disables EDI forwarding. NOTE: Does not disable pre-notification if generated. Valid values: YES, NO
<val n="partition">	Defines the profile group where the shipment should be stored in the online systems.
</meta>	

7.2 Sender

Declared under the root element <data>.

<sender sndid="...">	Attribute <code>sndid</code> corresponds to sender ID/quick ID. Any contents.
<val n="name">	Sender's name
<val n="address1">	Address line 1
<val n="address2">	Address line 2
<val n="zipcode">	Zip code
<val n="city">	City
<val n="country">	Country code according to ISO-standard
<val n="contact">	Contact person
<val n="phone">	Phone number
<val n="fax">	Fax number
<val n="orgno">	Organisation number (only for Sweden)
<val n="vatno">	VAT number
<val n="email">	E-mail
<val n="sms">	Mobile phone number. In Sweden the number must begin with 07 and contain 10 digits.
<partner parid="...">	Attribute <code>parid</code> corresponds to carrier's ID. See the HELP > Code lists > Services menu in the online systems. For Unifaun PrintServer go to Unifaun Engine Administrator > Partner (general) > Select Partner.
<val n="custno">	Customer number
<val n="sourcecode"></val>	Use depends on Carrier (<code>Partner parid=</code>).
<val n=" custno_international">	Customer number for international services

<val n="ediaddress">	EDI-address. UFPS only.
<val n="palletregno">	Pallet reg. number for EUR-pallets
<val n="terminal">	Carrier departure terminal
<val n="postgiro">	Number for PlusGiro. Used mostly by Cash On Delivery add-on.
<val n="bankgiro">	Number for BankGiro. Used mostly by Cash On Delivery add-on.
<val n="konto">	Specifies an offshore account
<val n="iban">	IBAN account number
<val n="bic">	BIC number
<val n="paymentmethod">	Payment method for certain mail services. Valid values: INVO = Credit without delivery note INVODN = Credit with delivery note METERED = Domestic franking
<val n="bookingid">	OPAL/booking ID-number
<val n="bookingoffice">	Booking office
<val n=" custnoissuercode">	Customer number issuer
</partner>	
</sender>	

7.3 Receiver

Declared under the root element <data>.

<receiver rcvid="...">	Any contents. Mandatory.
<val n="name">	Receiver's name
<val n="address1">	Address line 1
<val n="address2">	Address line 2
<val n="zipcode">	Zipcode
<val n="city">	City
<val n="state">	State
<val n="country">	Country code according to ISO standard
<val n="contact">	Contact person

<val n="phone">	Phone number
<val n="fax">	Fax number
<val n="orgno">	Organisation number (for Sweden only)
<val n="vatno">	VAT number
<val n="email">	E-mail
<val n="sms">	Mobile phone number. For Sweden, the number must begin with 07 and contain 10 characters.
<val n="doorcode">	Door code
<partner parid="...">	Attribute <code>parid</code> corresponds to carrier's ID. See the HELP > Code lists > Services menu in the online systems. For Unifaun PrintServer go to Unifaun Engine Administrator > Partner (general) > Select Partner.
<val n="custno">	Customer number
<val n="palletregno">	Pallet reg. number for EUR-pallets
<val n="terminal">	Terminal, used with delivery terms for international freights.
<val n="postgiro">	Number for PlusGiro. Used mostly by Cash On Delivery add-on.
<val n="bankgiro">	Number for BankGiro. Used mostly by Cash On Delivery add-on.
<val n="konto">	Specifies an offshore account
<val n="agentno">	Agent ID. Normally set by application.
<val n=" custnoissuercode">	Customer number issuer
</partner>	
</receiver>	

7.4 Party

<party ptyid="...">	Any contents. Mandatory.
<val n="name">	Party's name
<val n="address1">	Address line 1
<val n="address2">	Address line 2

<val n="zipcode">	Zipcode
<val n="city">	City
<val n="state">	State
<val n="country">	Country code according to ISO standard
<val n="contact">	Contact person
<val n="phone">	Phone number
<val n="fax">	Fax number
<val n="orgno">	Organisation number (for Sweden only)
<val n="vatno">	VAT number
<val n="email">	E-mail
<val n="sms">	Mobile phone number. For Sweden, the number must begin with 07 and contain 10 characters.
<val n="doorcode">	Door code
<partner parid="...">	Attribute parid corresponds to carrier's ID. See the HELP > Code lists > Services menu in the online system. For Unifaun PrintServer go to Unifaun Engine Administrator > Partner (general) > Select Partner.
<val n="custno">	Customer number
<val n="palletregno">	Pallet reg. number for EUR-pallets
<val n="terminal">	Terminal, used with delivery terms for international freights
<val n="postgiro">	Number for PlusGiro. Used mostly by Cash On Delivery add-on.
<val n="bankgiro">	Number for BankGiro. Used mostly by Cash On Delivery add-on.
<val n="konto">	Specifies an offshore account.
<val n="agentno">	Agent ID. Normally set by application.
<val n=" custnoissuercode">	Customer number issuer.
</partner>	
</party>	

7.5 Shipment

Declared under the root element `<data>`.

<code><shipment orderno="..." mergeid="..."></code>	Unique order number. Any contents. Mandatory. Order number is searchable in the systems but not printed on shipping documents. Merge ID is supplied only when consolidation is used.
<code><val n="from"></code>	Defines the sender. Refers to the <code>sndid</code> value for sender.
<code><val n="legalfrom"></code>	Defines the legal sender (not printed on shipping documents, could be set to be print).
<code><val n="to"></code>	Defines the receiver. Refers to <code>rcvid</code> value for receiver.
<code><val n="returnto"></code>	Defines the return address.
<code><val n="legalto"></code>	Defines the legal receiver (not printed on shipping documents).
<code><val n="agentto"></code>	Defines the agent's ID for recipient in shipment.
<code><val n="customsfrom"></code>	Exporter
<code><val n="customsto"></code>	Importer
<code><val n="shpid"></code>	Shipment ID
<code><val n="freetext1"></code>	Free text field with any contents. Can be used for delivery instructions, for example. It is printed on shipping documents. 4 lines available, <code>freetext1-4</code> . Max. 30 characters/line.
<code><val n="eurpallets"></code>	Number of EUR pallets in the shipment. Requires <code>palletregno</code> for sender and receiver.
<code><val n="halfpallets"></code>	Number of exchange half pallets in the shipment.
<code><val n="quarterpallets"></code>	Number of exchange quarter pallets in the shipment.
<code><val n="reference"></code>	Shipment reference. Any contents. Max. 17 characters.
<code><val n="referencebarcode"></code>	Shipment reference as barcode. Max. 17 numeric characters.
<code><val n="rcvreference"></code>	Receiver's reference. Any contents. Max. 17 characters.
<code><val n="goodsdescription"></code>	Description of contents, only used for some carriers.
<code><val n="sisfreetext1"></code>	Free text field with any contents. Can be used for delivery instructions, for example. It is printed on shipping

	documents. 5 lines available, <code>sisfreetext1-5</code> . Max. 30 characters/line.
<code><val n="cmrfreetext1"></code>	Free text field with any contents. Can be used for delivery instructions, for example. Only printed on CMR waybill. 5 lines available, <code>cmrfreetext1-5</code> . Max. 30 characters/line.
<code><val n="cmrdocuments1"></code>	Fields for additional documents. 2 lines available, <code>cmrdocuments1-2</code> . Max. 30 characters/line.
<code><val n="cmrspecialagreement"></code>	Specifies any special agreement. Max. 30 characters.
<code><val n="waybillcondition"></code>	Defines condition for the waybill. Max. 30 characters.
<code><val n="termcode"></code>	Delivery terms or freight payer information. See the HELP > Code lists > Services (ERP-connection XML) or Terms - UPS menu in the online systems. For Unifaun PrintServer go to Unifaun Engine Administrator > Partner (general) > Select Partner.
<code><val n="termlocation"></code>	Defines the location where takeover for the specified delivery term is done.
<code><val n="printset"></code>	Defines which documents to print. Pipe characters are mandatory. Valid values: label = Label only, sis = Waybill only, * = None
<code><val n="shipdate"></code>	Defines shipment date, supplied as YYYY-MM-DD. Printed on certain shipping documents. The default value is the current date. Please note that EDI is sent on this date.
<code><val n="deliverydate"></code>	Defines delivery date, supplied as YYYY-MM-DD.
<code><val n="deliverytimeearliest"></code>	Defines the earliest time the delivery is made, supplied as HH:MM.
<code><val n="deliverytimelatest"></code>	Defines the latest time the delivery is made, supplied as HH:MM.
<code><val n="deliveryinstruction"></code>	Defines delivery instructions.
<code><val n="customsunit"></code>	Customs currency unit
<code><val n="flags"></code>	Special instruction for address formatting. Valid value: 3plformat = Prints out sender's address (legalfrom) instead of pickup address (from) on label/shipping documents.
<code><val n="pdfinsert"></code>	Used to specify PDF filename that is to be used for OneDoc (integrated document) feature.

<code><val n="route"></code>	Defines internal route code for the shipment. Used for differentiating shipments.
<code><val n="unitcode"></code>	Unit code. Only given if instructed by carrier.
<code><val n="preparedid"></code>	Prepared ID for DeliveryCheckout shipment.
<code><val n="keeppreparedshipment"></code>	Keep prepared shipment. DeliveryCheckout only. Valid values: <i>yes, no,</i>

7.6 Service

Declared under the `shipment` element.

<code><service srvid="..."></code>	Corresponds to carrier's service. See the HELP > Code lists > Services (ERP-connection XML) menu in the online systems. For Unifaun PrintServer go to Unifaun Engine Administrator > Partner (general) > Select Partner.
<code><val n="returnlabel"></code>	Defines if the shipment is a return shipment or not. Valid values: <i>yes, no, both</i>
<code><val n="nondelivery"></code>	Defines action when the package is undeliverable. RETURN = Return to sender. ABANDON = Treat as abandoned in receiver's country.
<code><val n="slac"></code>	Defines Shipper's Load and Count (SLAC) = total shipment pieces (e.g. 3 boxes and 3 pallets of 100 pieces each = SLAC of 303).
<code><val n="bookingid"></code>	Defines Booking ID/Booking number.
<code><val n="waybillinvoice"></code>	Defines number of invoices.
<code><val n="waybilleurcertificate"></code>	Defines number of EUR-certificates.
<code><val n="waybillexportnotification"></code>	Defines number of export notifications.
<code><val n="waybillincludedinbulk"></code>	Set as "yes" if included in bulk.
<code><val n="sortpos"></code>	Receiving HUB.

7.7 Addon

Declared under the element `service`.

<code><addon adnid="..."></code>	Corresponds to add-on service. See the HELP > Code lists > Services (ERP-connection XML) menu in the online systems. For Unifaun PrintServer go to Unifaun Engine Administrator > Partner (general) > Select Partner.
--	--

<val n="amount">	Amount. Used only with add-on COD (Cash On Delivery).
<val n="custno">	Customer number for carrier. Used primarily with RPAY (receiver pays) and OPAY (other payer).
<val n="reference">	Payment reference. Used with add-on COD.
<val n="referencetype">	Type of reference. Valid values: TXT = Text, OCR = OCR number. Used only with add-on COD (Cash On Delivery).
<val n="misc">	Defines value for <code>misctype</code> .
<val n="misctype">	Used to define notification mode for add-on NOT. See the HELP > Code lists > Services (ERP-connection XML) menu in the online systems.
<val n="account">	Account number. Used only with add-on COD (Cash On Delivery).
<val n="accounttype">	Account type. Valid values: PGTXT: PlusGiro (Text reference) PGOCR: PlusGiro (OCR reference) BGTXT: BankGiro (Text reference) BGOOCR: BankGiro (OCR reference) ACCDKOCR: Account Denmark (OCR reference) ACCDKTXT: Account Denmark (Text reference) ACCSETXT: Account Sweden (Text reference) AVISSETXT: Avi account Sweden (Text reference) ACCFI: Account Finland ACCFO: Account Faroe Islands ACCIS: Account Iceland ACCNO: Account Norway (Text reference) ACCNOOCR: Account Norway (OCR reference) ACCPL: Account Poland ACCDE: Account Germany ACCAX: Account Åland Islands Konto = Offshore account Used only with add-on COD (Cash On Delivery). To specify if OCR or not, see the "referencetype" tag.
<val n="text1">	Defines name of collection point example: Posten add-on DLVNOT. Defines name of the bank example: DHL Germany.

<val n="text2">	Defines address for collection point example: Posten add-on DLVNOT.
<val n="text3">	Defines phone number for Posten example: add-ons PODNOT, DLVNOT and PRENOT.
<val n="text4">	Defines e-mail address for Posten example: add-ons PODNOT, DLVNOT and PRENOT.
<val n="bank">	Defines bank ID example: DHL Germany.
<val n="tempmax">	Max. temperature allowed.
<val n="tempmin">	Min. temperature allowed.
</addon>	

7.8 Booking

Declared under the `service` element. Used when booking of pickup should be done in conjunction with printing. Contact Unifaun Support for more information.

<booking>	
<val n="pickupbooking">	Valid values: yes, no.
<val n="pickupdate">	Pickup date. Supplied as YYYY-MM-DD.
<val n="pickuptimefrom">	Earliest pickup time. Supplied as HH:MM.
<val n="pickuptimeto">	Latest pickup time. Supplied as HH:MM.
<val n="pickuptext1">	Pickup instructions
<val n="pickupmisc">	Information about where to send pickup confirmation. Only e-mail can be supplied at this time. Must be supported by carrier.
</booking>	
</service>	

7.9 Container

Declared under the `shipment` element.

<container type="parcel">	Parcel information can be supplied in various ways. See chapter Container.
<val n="copies">	Number of parcels

<val n="cntid1">	Parcel ID. Used only for custom parcel ID. UFPS only. Cntid is to be incremented according to number of parcels.
<val n="prcid1">	Parcel ID. Used only for custom parcel ID. UO and PO only. Prcid is to be incremented according to number of parcels.
<val n="marking">	Goods marking
<val n="packagecode">	Package code. See the HELP > Code lists > Package codes menu in the online systems. For Unifaun PrintServer go to Unifaun Engine Administrator > Partner (general) > Select Partner.
<val n="weight">	Weight
<val n="volume">	Volume
<val n="area">	Load meter. Can only be specified for entire shipment.
<val n="length">	Length
<val n="width">	Width
<val n="height">	Height
<val n="itemno">	Item number
<val n="contents">	Contents
<val n="dnguncode">	UN-number for ADR. Supplied as a 4 digit code.
<val n="dngghzcode">	Label number for ADR (Subsidiary risk)
<val n="dngpkcode">	Packaging group/ADR-class. Supplied as I, II or III.
<val n="dngadrclass">	ADR-class (Main class)
<val n="dngdescr">	Official transport name for item regarding ADR (Proper Shipping Name)
<val n="dngtechnicaldescr">	Technical description
<val n="dngmpcode">	Defines if the contents contaminate the marine environment, ADR only. Valid values: 1 = Toxic and 2 = Non-toxic for the marine environment
<val n="dngnote">	Note for ADR goods
<val n="dngnetweight">	Net weight for ADR goods class I (usually explosive contents). Always mandatory for DBSchenker, regardless of class. Defined in kg. Can also be brutto for LQ.

<val n="dngtrcode">	Tunnel restriction code
<val n="dngpktype">	DNG Package type
<val n="dngquantity">	Quantity of DNG package type
<val n="dnglimitedquantities">	Defines if the parcel row contains Limited Quantities of dangerous goods. Valid values: YES/NO.
<val n="pallet">	Sets amount of pallets, only used for some services. Must be placed in a separate container node with type totals and measure totals.
<val n="piece">	Sets amount of pieces, only used for some services. Must be placed in a separate container node with type totals and measure totals.
</container>	
</shipment>	

7.10 Options

E-mail pre-notification

Declared under the `shipment` element.

<ufonline>	
<option optid="enot">	
<val n="from">	Sender of e-mail message
<val n="to">	Recipient
<val n="cc">	Recipient of a message copy
<val n="bcc">	Recipient of blind message copy
<val n="errorerto">	Recipient of error messages
<val n="message">	Text string with any contents. Line breaks can be entered with pipe character, .
<val n="mailtemplate">	Mailtemplate. Used for "Anpassad föravisering".
</option>	
</ufonline>	

Link to print

Declared under the `shipment` element.

<ufonline>	
<option optid="...">	The attribute's <code>optid</code> -value defines the type of shipment, normal and return. Valid values: LNKPRTN = Normal, LNKPRTR = Return
<val n="sendemail">	Send e-mail message. Valid values: YES, NO.
<val n="from">	Sender of e-mail message
<val n="to">	Recipient
<val n="bcc">	Recipient of blind message copy
<val n="message">	Text string with any contents. Line breaks can be entered with pipe character, .
</option>	
</ufonline>	

Consolidation

Declared under the `shipment` element.

<ufonline>	
<option optid="CONSOLIDATE">	
</option>	
</ufonline>	

7.11 Customs declaration documents UFPS

Declared under the `shipment` element.

ED document (edoc)

<edoc>	
<val n="senderorgno">	Sender's organisation number (Sweden only)
<val n="receiverorgno">	Receiver's organization number (Sweden only)
<val n="parcelcount">	Parcel count
<val n="reference">	Reference number
<val n="sourcecountry">	Country of origin
<val n="destcountry">	Destination country

<val n="shipcodeborder">	Mode of transport (at the border). Valid values: SEA, RAIL, ROAD, AIR, POST, FIXED = Fixed transport, WATER = Water transport on inner waterways, OWN = Own propulsion.
<val n="shipcodedomestic">	Mode of transport (domestic). Valid values: SEA, RAIL, ROAD, AIR, POST, FIXED = Fixed transport, WATER = Water transport on inner waterways, OWN = Own propulsion.
<val n="finance1">	Financial information. 2 lines available, finance1-2.
<val n="declcity">	City
<val n="decldate">	Date
<val n="declarant">	Declarant/agent
<val n="edocnormal">	yes = Export, normal procedure no = Export, commercial item list attached
<val n="repr1">	Declarant's representative. 2 lines available, repr1-2.
<val n="reprorgno">	Declarant's organization number
<container type="stat">	Container type, always stat
<val n="statno">	Stat.no./TARIC no
<val n="substatno1">	Item code/subcode. 2 lines available, substatno1-2.
<val n="procedure">	Procedure
<val n="totalweight">	Net weight in kg
<val n="statvalue">	Statistic value
<val n="goodsmark1">	Goods marking. 6 lines available, goodsmark1-6.
<val n="specinfo1">	Special information. 4 lines available, specinfo1-4.
</container>	
<edoc>	

Commercial invoice (invoice)

<invoice type="proforma">	
<val n="invoicetype">	PROFORMA = Proforma invoice, STANDARD = Commercial invoice
<val n="invoiceno">	Invoice number
<val n="from">	Sender's ID, refers to <i>sndid</i> .
<val n="to">	Receiver's ID, refers to <i>rcvid</i> .

<val n="customsunit">	Customs currency unit
<val n="impexptype">	Import/export type. Valid values: TEMPORARY = Temporary, REEXPORT = Re-export, GIFT = Gift, DOCUMENTS = Documents, SAMPLE = Sample, RETURN = Return, OTHER = Other, requires the explanation value, PERMANENT = Permanent, INTERNAL = Internal
<val n="freightcharges">	Freight charges
<val n="insurancecharges">	Insurance charges
<val n="othercharges">	Other charges
<val n="invoicedecl1">	Default: "The exporter of the products covered by this document declares that, except"
<val n="invoicedecl2">	Default: "where otherwise clearly indicated, these products are of EU preferential origin."
<val n="explicenseno">	Number of export license
<val n="expdecl1">	Default: "I declare that of the best of my knowledge the information on this invoice is"
<val n="expdecl2">	Default: "true and correct."
<val n="generalnotes1">	A general description of contents for customs. 4 lines available, generalnotes1-4.
<val n="supervisor">	Invoice supervisor
<val n="jobtitle">	Job title for invoice supervisor
<val n="date">	Invoice date. Shipment date is set by default. If supplied, Swedish date format must be used, YYYY-MM-DD.
<val n="termcode">	Delivery terms
<val n="termlocation">	Defines the place where goods are taken over according to set delivery terms.
<val n="reference">	Reference number
<val n="sourcecountry">	Country of origin. Sender's country is default.
<line measure="..." shipid="..." cntid="...">	The attribute's <code>measure</code> valid values: <code>unit</code> = Values are defined per article. <code>totals</code> = Values are defined for all articles. Attribute <code>shipid</code> defines the shipment that contains the article.

	Attribute <code>cntid</code> identifies the parcel that contains the article.
<code><val n="units"></code>	Unit/packaging type. Preset values are used for Posten: GRAM = Gram, M2 = Meters squared, LITER = Litre, PAIR = Pair, PIECE = Piece
<code><val n="quantity"></code>	Quantity, concerns number of units, e.g. 1 carton. Default value is 1.
<code><val n="netweight"></code>	Net weight in kg
<code><val n="otherquantity"></code>	Other quantity/unit, e.g. 30. The unit is set by <code>statno</code> e.g. piece.
<code><val n="customsvalue"></code>	Customs value
<code><val n="contents"></code>	Contents/item name
<code><val n="sourcecountry"></code>	Country of origin. Sender's country is set by default.
<code><val n="statno"></code>	Stat.no., for export. 8 digits.
<code><val n="substatno1"></code>	Item no/subcode. 2 lines available, <code>substatno1-2</code> .
<code></line></code>	
<code></invoice></code>	

Customs declaration (CN23)

<code><cn23></code>	
<code><val n="senderorgno"></code>	Sender's organisation number (Sweden only)
<code><val n="receiverorgno"></code>	Receiver's organization number (Sweden only)
<code><val n="parcelcount"></code>	Number of parcels
<code><val n="comlistcount"></code>	Goods specification
<code><val n="reference"></code>	Reference number
<code><val n="reprorgno"></code>	Declarant's organization number
<code><val n="repr1"></code>	Declarant's representative. 2 lines available, <code>repr1-2</code> .
<code><val n="exportcountry"></code>	Sending country, supplied as ISO country code.
<code><val n="sourcecountry"></code>	Country of origin, supplied as ISO country code.
<code><val n="destcountry"></code>	Destination country, supplied as ISO country code.
<code><val n="shipcodeborder"></code>	Mode of transport (at the border) Valid values: SEA, RAIL, ROAD, AIR, POST, FIXED = Fixed transport, WATER = Water transport on inner waterways, OWN = Own propulsion.

<val n="shipcodedomestic">	Mode of transport (at the border) Valid values: SEA, RAIL, ROAD, AIR, POST, FIXED = Fixed transport, WATER = Water transport on inner waterways, OWN = Own propulsion.
<val n="finance1">	Financial information. 2 lines available, finance1-2.
<val n="declcity">	City
<val n="decldate">	Date
<val n="declarant">	Declarant/Agent
<line>	
<val n="units">	Unit/packaging type. Preset values are used for Posten: GRAM = Gram, M2 = Meters squared, LITER = Litre, PAIR = Pair, PIECE = Piece
<val n="quantity">	Quantity, concerns number of units, e.g. 1 box. Default value is 1.
<val n="netweight">	Net weight in kg
<val n="otherquantity">	Other quantity/unit, e.g. 30. The unit is set by statno e.g. piece.
<val n="customsvalue">	Customs value
<val n="contents">	Contents/item name
<val n="sourcecountry">	Country of origin. Sender's country is set by default.
<val n="statno">	HS Tariff Code (stat no), for export. 8 digits.
<val n="substatno">	Additional codes. 2 lines available, substatno1-2.
</line>	
</cn23>	

7.12 Customs declaration documents Unifaun/Pacsoft Online, Posti SmartShip

Declared under the `shipment` element. Desired customs declaration for UO and PO is supplied in the `documents` attribute for `customsdeclaration`, separated with pipe characters. The values in parenthesis represent field numbers in the online systems.

<customsdeclaration documents="...">	PROFORMA = Proforma / commercial invoice EDOC = ED document PROFORMAPLABEDI = Proforma / commercial invoice by EDI and a printed copy PLABEDI = Proforma / commercial invoice by EDI only
--------------------------------------	--

	<p>PROFORMATNT = Proforma / commercial invoice for TNT</p> <p>NOTETNT = Note TNT</p> <p>PNLWAYBILLED1 = I will send a separat ecustoms declaration (EDI)</p> <p>CN23 = Customs declaration</p> <p>PROFORMAUPS = Proforma / commercial invoice for UPS</p> <p>UPSEDI = Paperless Invoice for UPS (Only for test customers at this stage)</p> <p>FEDEXP = Commercial Invoice for FEDEX</p> <p>FEDEXE = Electronic Trade Documents (ETD) for FEDEX</p> <p>TRADEINVOICEPDK = Customs Declaration for PODK</p> <p>DHLROADSTD = DHL Customs Standard</p> <p>DHLAIR = DHL Express</p> <p>DHLAIRPLT = DHL Paperless Trade</p> <p>CN22 = CN22</p> <p>CN22INC = CN22 used by Deutsche Post, PostNord Brev and Direct Link</p>
<val n="invoicetype">	<p>PROFORMA = Proforma invoice, STANDARD = Commercial invoice.</p> <p>Default is STANDARD.</p>
<val n="senderorgno">	Sender's organization number (Sweden only) (2)
<val n="receiverorgno">	Receiver's organization number (Sweden only) (8)
<val n="senderphone">	Sender's phone number
<val n="receiverphone">	Receiver's phone number
<val n="sendercustno">	Sender's customer number (PLAB only)
<val n="receivercustno">	Receiver's customer number (PLAB only)
<val n="parcelcount">	Parcel count (6)
<val n="reference">	Reference number (7)
<val n="sourcecountry">	Country of origin (16)
<val n="destcountry">	Destination country (17)
<val n="shipcodeborder">	Means of transport (at the border) (25) Valid values: SEA, RAIL, ROAD, AIR, POST, FIXED = Fixed transport, WATER = Water transport on inner waterways, OWN = Own propulsion.
<val n="shipcodedomestic">	Means of transport (domestic) (26) Valid values: SEA, RAIL, ROAD, AIR, POST, FIXED = Fixed transport, WATER = Water transport on inner waterways, OWN = Own propulsion.

<val n="finance1">	Financial information. 2 lines available, finance1-2 (28).
<val n="declcity">	City (54)
<val n="decldate">	Date (54)
<val n="edocnormal">	yes = Export, normal procedure no = Export, commercial item list attached
<val n="repr1">	Declarant's / agent's representative. 2 lines available, repr1-2. (14)
<val n="reprorgno">	Declarant's / agent's organization number (14)
<val n="invoiceno">	Invoice number
<val n="freightcharges">	Freight charges
<val n="insurancecharges">	Insurance charges
<val n="othercharges">	Other charges
<val n="explicenseno">	Export license number
<val n="expdecl1">	Default: "I declare that of the best of my knowledge the information on this invoice is"
<val n="expdecl2">	Default: "true and correct."
<val n="generalnotes1">	General description of the contents, for customs. 4 lines available, generalnotes1-4.
<val n="supervisor">	Name of invoice supervisor
<val n="jobtitle">	Job title for invoice supervisor
<val n="customsunit">	Customs currency unit
<val n="impexptype">	Import/export type. Valid values: TEMPORARY = Temporary, REEXPORT = Re-export, GIFT = Gift, DOCUMENTS = Documents, SAMPLE = Sample, RETURN = Return, OTHER = Other, requires the explanation value, PERMANENT = Permanent, INTERNAL = Internal
<val n="invoicedecl1">	Default: "The exporter of the products covered by this document declares that, except
<val n="invoicedecl2">	Default: "where otherwise clearly indicated, these products are of EU preferential origin."
<val n="implicenseno">	Import license number
<val n="certificate">	Certificate
<val n="explanation">	Explanation, only if impexptype set to OTHER

<val n="comment1">	4 lines available, comment1-4.
<val n="termcode">	Delivery terms
<line>	
<val n="statno">	HS Tariff Code (stat no) (33)
<val n="substatno1">	Additional codes. 2 lines available, substatno1-2. (33)
<val n="procedure">	Procedure (37)
<val n="grossweight">	Gross weight in kg
<val n="statvalue">	Stat value
<val n="goodsmark1">	Goods marking. 6 fields available, goodsmark1-6. (31)
<val n="specinfo1">	Special information. 4 fields available, specinfo1-4. (44)
<val n="contents">	Contents/item name
<val n="quantity">	Quantity, concerns number of units, e.g. 1 box. Default value is 1.
<val n="netweight">	Net weight in kg (38)
<val n="customsvalue">	Customs value (46)
<val n="sourcecountry">	Country of origin, supplied as ISO country code. (34)
<val n="units">	Unit/packaging type. Preset values are used for Posten: GRAM = Gram, M2 = Meters squared, LITER = Litre, PAIR = Pair, PIECE = Piece Preset values are used for UPS: CRATE
<val n="otherquantity">	Other quantity/unit, e.g. 30. The unit is set by statno e.g. piece.
</line>	
</customsdeclaration>	

8 Error messages

If the information in the order file is missing or erroneous it is communicated by the systems in different ways.

In UFPS a label is printed containing necessary information about the error. UFPS can also send an e-mail message containing the error information if desired.

No label is printed in UO. The order file will instead be stored under menu **SHIPMENTS** > Stored printings with status **Invalid**. Click on the  icon for details.

A list containing the most frequent errors is displayed below.

8.1 UO and PO

Error message	Explanation
Address/Name	Name is missing.
Address/Zipcode	Zip code is missing or incorrectly formatted.
Address/City	City is missing.
Address/Country	Country code is missing.
Addon	Conflict between values. Several add-on services have been supplied that cannot be combined.
Addon/CustNo	Customer number is missing for the service/add-on.
Addon/Account	Account number is missing for the add-on.
Addon/Amount	Amount is missing for the add-on.
Addon/Id	Contact information is required by add-on (phone, fax, sms or e-mail).
Addon/Misc	Missing information for add-on using type miscellaneous.
Addon/Reference	Reference for add-on is missing.
Party/Custno	Customer number for carrier is missing.
Service/Id	Service code is missing or is incorrect or the supplied service cannot be used to the destination.

8.2 UFPS

Error messages that are printed or sent as e-mail from UFPS have a certain logic to them, based on the same logic as the order files.

Example 1

```
data.receiver[rcvid=RCV1].receiver[rcvid=RCV1].name : missing value
```

The first part of the error message contains the faulty element's name and its attribute, if any.

```
data.receiver[rcvid=RCV1]
```

points out that the error is in the element `receiver`.

The second part of the error message refers to a value in the actual element.

```
data.receiver[rcvid=RCV1].name
```

points out that the error is in the value `name`.

The third and last part of the message contains the error.

```
: missing value
```

points out that the data is missing.

Hence, the error message can be interpreted as the name for receiver RCV1 is missing.

Example 2

`data.sender[sndid=1].sender[sndid=1].parid : missing partner.custno (SBTL)`

The first part of the error message contains the faulty element's name and its attribute, if any.

`data.sender[sndid=1]` points out that the error is in the element `sender`.

The second part of the error message refers to a value in the actual element.

`data.sender[sndid=1].parid` : points out that the error is in the value `parid`.

The third and last part of the message contains the error.

`: missing partner.custno (SBTL)` points out that the data is missing.

Hence, the error message can be interpreted as the customer number for carrier (in this case SBTL) is missing for sender 1.

Only the last parts of messages are explained in the table below.

Error message	Explanation
<code>missing value</code>	Data missing for supplied value
<code>missing partner.custno</code>	Missing partner (carrier) customer number
Country specific validation (YY) for zipcode value XXXX rules: (<code>'operator', 'C'</code>), (<code>'checkregexpr', '[0-9]{5,5}'</code>), (<code>'checkregmask', 'NNNNN'</code>) : <code>checkregexpr</code> (<code>rematch = None</code>)	Value for zipcode (XXXX) does not conform to allowed values for the country YY. 'NNNNN' defines correct formatting, in this case 5 digits.
Missing serviceroute <code>srvid:XXX</code> , <code>country(from):YY</code> , <code>country(to):ZZ</code>	The service XXX cannot be carried out from country YY to country ZZ.
<code>.Invalid combination : ,xxx,xxx</code>	The add-on services cannot be combined.
<code>missing mandatory value for: custno()</code>	Mandatory value for the add-on service is missing, in this case the customer number for carrier.
<code>Addon:xxx valuename:yyy value:zzz</code> <code>valuelength < minlength(6)</code>	The value ZZZ for valuename YYY does not conform to the lowest length (6) allowed for add-on service XXX.
<code>missing or invalid/box ZIP code for receiver</code>	The zip code is non-existent or belongs to a P.O. Box address.
<code>missing relation</code>	Values for from or to refer to an incorrect <code>sndid</code> or <code>rcvid</code> value.
<code>missing mandatory value for: shipment.service.addon[adnid=XXX]:YYY</code>	Value in YYY is missing for add-on service XXX.