



**AGETOR®**

AXT EDI

## Table of contents

|     |  |    |
|-----|--|----|
| 1   | Description .....                                    | 3  |
| 2   | Dependencies .....                                   | 3  |
| 3   | Installation .....                                   | 3  |
| 4   | Conversion .....                                     | 4  |
| 5   | AXT configuration.....                               | 5  |
| 6   | EDI2XML configuration .....                          | 8  |
| 7   | Segments and elements.....                           | 11 |
| 7.1 | Record specification from a FIXED position file..... | 11 |
| 7.2 | Record specification from an EDI format file .....   | 12 |
| 8   | Examples.....  | 13 |
| 8.1 | Example INPUTfile FIXED position. ....               | 13 |
| 8.2 | Example INPUTfile EDI Invoice .....                  | 14 |

## 1 Description

The Edi filter converts EDI formats to XML.

The Edi filter can take


1. EDI data input and convert this input to XML, for instance Hancom or SFTI.
2. Fixed Position file, also with different row specifications.

## 2 Dependencies

You need to have AXT Basic 2.0.0 installed to use this Filter.

## 3 Installation

This section gives you step-by-step information on how to install AXT EDI.

 The newest version of AXT EDI is always available at the Bording Data download center at <http://www.agetor.dk>.

Before you begin the installation of the AXT EDI, make sure you have the following prerequisites installed:

- Java Development Kit (JDK) version 1.3 or newer.
- AGETOR Development Kit (ADK) version 2.0.0 or newer.
- AXT version 2.0.0 or newer.
- Webserver with servlet runner e.g. Jakarta-Tomcat or Apache with Apache JServ.

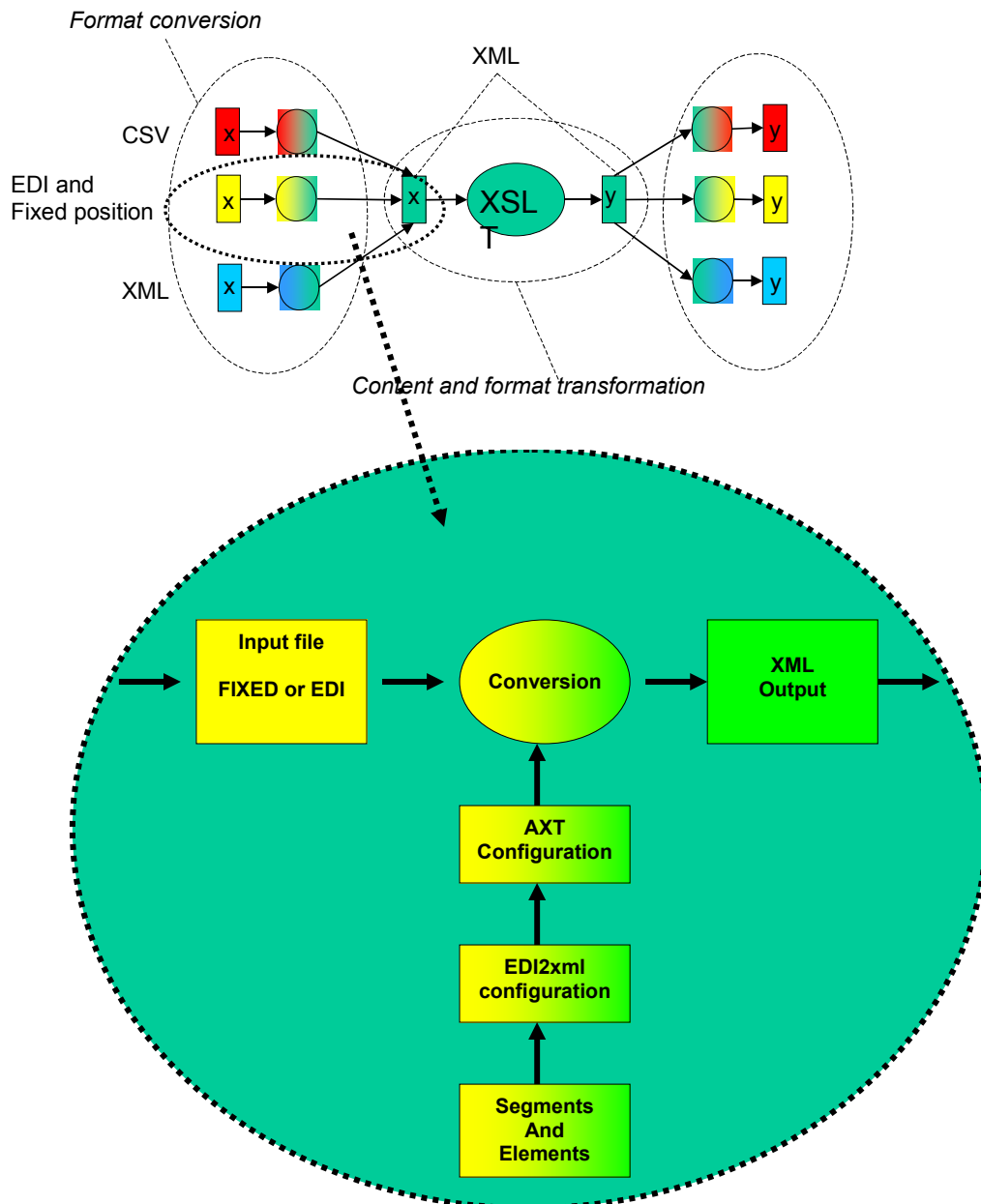
Download the newest version of the AXT EDI and unzip the package into your "AGETOR\_HOME/install/packages" directory. If you are upgrading an existing installation the existing configuration will be retained.

- Open a command window with "AGETOR\_HOME/bin/prompt.bat" and run "installer.bat".
- After the AGETOR® installtool has started open a Internet browser and point to "http://localhost:8020".
- If prompted for login and password, please type in your login and password.
- Under "Product(s) ready to install" click on "AXT EDI 2.0.0" and answer the few questions.

Please note, that new properties might have been added and you should always consult the release.txt for any changes you might need to incorporate.

## 4 Conversion

The figure 1 below shows the overall AXT conversion and transformation process. The big circle shows the conversion process for a Fixed position or EDI format file to XML and which elements the conversion uses. These elements will be explained in the following chapters.



## 5 AXT configuration

To use the Edi2XML filter:

### AXTConfiguration.xml

```
<doc name="EDIinvoice-document">
  <key name="ip" value="*" />
  <key name="content-type" value="text/plain" />
  <key name="url" value="edi-invoice" />
  <!-- map input to basic xml -->
  <filter class="dk.bording.axt.tc.edixml.Edi2XML">
    <param name="confname" value="EDIinvoice-example" />
    <param name="una" value="true" />
  </filter>
</doc>

<doc name="Fixed-document">
  <key name="ip" value="*" />
  <key name="content-type" value="text/plain" />
  <key name="url" value="fixed-orders" />
  <!-- map input to basic xml -->
  <filter class="dk.bording.axt.tc.edixml.Edi2XML">
    <param name="confname" value="fixedpos-example" />
  </filter>
</doc>
```

Parameters given to the filter can be all of the following:

| Parameters    | Description   | Optional/default   |
|---------------|---|--|
| confname      | The configuration entrance defined by the name attribute on the document tag in the main Edi2XML configuration e.g. The main configuration is defined by the "axt.transformation.edi.configuration" property or by the parameter "configuration".   | Required.  |
| xmlencoding   | The xml encoding is the encoding that is in the xml prologue at the very top of the produced xml document. E.g. if the "xmlencoding"-parameter is "MyEncoding" then the following string will appear at the very beginning of the produced document;<br><?xml version="1.0" encoding="MyEncoding"?>   | Optional.<br>Defaults to ISO-8859-1  |
| roottag       | The root tag is the xml root element that is used to surround the produced xml. E.g. when this parameter have the value "document" the produced xml will be surrounded by <document> and </document>. If root tag is wanted the value must be set to "noroottag".   | Optional.<br>By default the root tag is the name of the document.  |
| buffer        | With this parameter you can determine if the output should be buffered before it is written, thereby making sure that an eventual error is the first thing in the output. Do only use this parameter on small documents and if Edi2XML is the last or only filter in the filter chain.  | Optional.<br>True or false, defaults to false.   |
| indent        | Determines if the output should be nicely indented or not. This parameter should normally be set to true if Edi2XML is the last filter in a document chain and if the output should be seen and read by a person. If the Edi2XML filter is not the last filter in the document chain and other filters can change the final output, then this parameter should be false (default).  | Optional.<br>True or false, defaults to false.   |
| una           | If this parameter value is set to true the first 9 characters of the input stream would be read as the UNA segment and the separator and terminator characters in the Edi2XML configuration are ignored. E.g. UNA:+.? '(9 characters)<br>UNChar1-3 UNA<br>UNChar4 - Subelement-separator<br>UNChar5 - Element-separator<br>UNChar6 - Decimal-notation<br>UNChar7 - Release-character<br>UNChar8 - ReservedForFutureUse<br>UNChar9 - Segment-terminator<br>All 9 characters must be at the very beginning of the input stream. If the characters are not at the beginning of the input but the una parameter is true then parameter is set to false. | Optional.<br>True or false, defaults to false.<br><br>If set to true but the first characters is not a UNA segment then the una parameter automatically becomes false. |
| configuration | The configuration parameter is used to override the axt.transformation.edi.configuration property thereby making it possible to have more than one configuration entry. By using this parameter you can state which main configuration each   | Optional.<br>Overrides the axt.transformation.edi.configuration  |

|                           | configured filter must have.   | property.                                      |
|---------------------------|--|--|
| ignore-undefined-segments | When this parameter is set to true all undefined segments are ignored during conversion of the EDI file. Undefined segments are segments that have not been defined within the current document configuration. When an undefined segment is encountered it is ignored and the filter will proceed with the next defined segment and the output will be exactly the same as the output from an EDI file without the undefined segments. | Optional.<br>True or false, defaults to false. |

## 6 EDI2XML configuration

A definition of an Edi conversion definition can look like the following:

Notice that the same 'segmentdef' file can be reused at different levels; which will minimize the definition phase of a document.

### Configuration file: e.g. edi2xml.xml

```
<document name="EDIinvoice-example" fixed="false" releasechar="?" decimalchar="." subelementseparator="."
elementseparator="+"
segmentterminator="" roottag="EDI-invoice-doc">
  <segment name="UNB" minrow="1" maxrow="1" segmentdef="{AGETOR_HOME}/conf/axt/edi/ex/EDlunbSegment.xml"/>
  <segment name="UNH" minrow="1" maxrow="1" segmentdef="{AGETOR_HOME}/conf/axt/edi/ex/EDlunhSegment.xml"/>
  <segment name="BGM" minrow="1" maxrow="1" segmentdef="{AGETOR_HOME}/conf/axt/edi/ex/EDlbgmSegment.xml"/>
  <segment name="DTM" minrow="1" maxrow="100" segmentdef="{AGETOR_HOME}/conf/axt/edi/ex/EDldtmSegment.xml"/>
  <segment name="RFF" minrow="1" maxrow="3" segmentdef="{AGETOR_HOME}/conf/axt/edi/ex/EDlrrffSegment.xml">
    <segment name="DTM" minrow="0" maxrow="3" segmentdef="{AGETOR_HOME}/conf/axt/edi/ex/EDldtmSegment.xml"/>
  </segment>
  <segment name="NAD" minrow="1" maxrow="6" segmentdef="{AGETOR_HOME}/conf/axt/edi/ex/EDlnadSegment.xml">
  <segment name="RFF" minrow="0" maxrow="1" segmentdef="{AGETOR_HOME}/conf/axt/edi/ex/EDlrrffSegment.xml"/>
  </segment>
  <segment name="PAT" minrow="0" maxrow="3" segmentdef="{AGETOR_HOME}/conf/axt/edi/ex/EDlpatSegment.xml">
  <segment name="DTM" minrow="1" maxrow="100" segmentdef="{AGETOR_HOME}/conf/axt/edi/ex/EDldtmSegment.xml"/>
  <segment name="MOA" minrow="0" maxrow="1" segmentdef="{AGETOR_HOME}/conf/axt/edi/ex/EDlmoaSegment.xml"/>
  </segment>
  <segment name="LIN" minrow="0" maxrow="100" segmentdef="{AGETOR_HOME}/conf/axt/edi/ex/EDllinSegment.xml">
  <segment name="PIA" minrow="0" maxrow="1" segmentdef="{AGETOR_HOME}/conf/axt/edi/ex/EDlpiaSegment.xml"/>
  <segment name="IMD" minrow="0" maxrow="1" segmentdef="{AGETOR_HOME}/conf/axt/edi/ex/EDlimdSegment.xml"/>
  <segment name="QTY" minrow="0" maxrow="5" segmentdef="{AGETOR_HOME}/conf/axt/edi/ex/EDlqtySegment.xml"/>
  <segment name="DTM" minrow="0" maxrow="1" segmentdef="{AGETOR_HOME}/conf/axt/edi/ex/EDldtmSegment.xml"/>
  <segment name="QVR" minrow="0" maxrow="1" segmentdef="{AGETOR_HOME}/conf/axt/edi/ex/EDlqvrSegment.xml"/>
  <segment name="MOA" minrow="0" maxrow="1" segmentdef="{AGETOR_HOME}/conf/axt/edi/ex/EDlmoaSegment.xml"/>
  <segment name="PRI" minrow="0" maxrow="3" segmentdef="{AGETOR_HOME}/conf/axt/edi/ex/EDlpriSegment.xml"/>
  <segment name="RFF" minrow="0" maxrow="1" segmentdef="{AGETOR_HOME}/conf/axt/edi/ex/EDlrrffSegment.xml">
    <segment name="DTM" minrow="0" maxrow="1" segmentdef="{AGETOR_HOME}/conf/axt/edi/ex/EDldtmSegment.xml"/>
  </segment>
  <segment name="TAX" minrow="0" maxrow="3" segmentdef="{AGETOR_HOME}/conf/axt/edi/ex/EDluntSegment.xml">
  <segment name="MOA" minrow="0" maxrow="100"
segmentdef="{AGETOR_HOME}/conf/axt/edi/ex/EDlmoaSegment.xml"/>
  </segment>
  <segment name="NAD" minrow="0" maxrow="6" segmentdef="{AGETOR_HOME}/conf/axt/edi/ex/EDlnadSegment.xml">
  <segment name="RFF" minrow="0" maxrow="1" segmentdef="{AGETOR_HOME}/conf/axt/edi/ex/EDlrrffSegment.xml"/>
  </segment>
  </segment>
  <segment name="UNS" minrow="1" maxrow="1" segmentdef="{AGETOR_HOME}/conf/axt/edi/ex/EDlunsSegment.xml"/>
  <segment name="CNT" minrow="1" maxrow="2" segmentdef="{AGETOR_HOME}/conf/axt/edi/ex/EDlcntSegment.xml"/>
  <segment name="MOA" minrow="0" maxrow="100" segmentdef="{AGETOR_HOME}/conf/axt/edi/ex/EDlmoaSegment.xml">
  <segment name="RFF" minrow="0" maxrow="1" segmentdef="{AGETOR_HOME}/conf/axt/edi/ex/EDlrrffSegment.xml">
    <segment name="DTM" minrow="0" maxrow="1" segmentdef="{AGETOR_HOME}/conf/axt/edi/ex/EDldtmSegment.xml"/>
  </segment>
  </segment>
  <segment name="TAX" minrow="0" maxrow="3" segmentdef="{AGETOR_HOME}/conf/axt/edi/ex/EDltaxSegment.xml">
  <segment name="MOA" minrow="0" maxrow="100"
segmentdef="{AGETOR_HOME}/conf/axt/edi/ex/EDlmoaSegment.xml"/>
  </segment>
  <segment name="UNT" minrow="1" maxrow="3" segmentdef="{AGETOR_HOME}/conf/axt/edi/ex/EDluntSegment.xml"/>
  <segment name="UNZ" minrow="1" maxrow="3" segmentdef="{AGETOR_HOME}/conf/axt/edi/ex/EDlunzSegment.xml"/>
</document>

<document name="fixedpos-example" fixed="true" roottag="Order-DOCUMENT">
  <segment name="EDIMGR" xmlns="top" segmentdef="{AGETOR_HOME}/conf/axt/edi/ex/TopSegment.xml"/>
  <segment name="H" xmlns="head" segmentdef="{AGETOR_HOME}/conf/axt/edi/ex/HeadSegment.xml"/>
  <segment name="R" xmlns="line" maxrow="500" segmentdef="{AGETOR_HOME}/conf/axt/edi/ex/LineSegment.xml"/>

```

---

---

</document>

In the configuration file any given number of definitions like the above can be present. The only limitation is that the **name** attribute in the **document** tag has to be unambiguous.

The Edi conversion definition is used for describing the structure of the XML output.

| Tag             | Attribute           | Meaning  | Required  |          | Possible values       | Default value |
|-----------------|---------------------|--|-----------|----------|-----------------------|---------------|
|                 |                     |  | Non-fixed | Fixed    |                       |               |
| <b>Document</b> | name                | Unambiguous identification of the document. This is referred to in the AXT configuration using the parameter <b>confname</b> e.g. <code>&lt;param name="confname" value="fixedpos-example"/&gt;</code> | Yes       | Yes      |                       | N/A           |
|                 | fixed               | Used for parsing fixed-position documents  | No        | Yes      | true or false         | false         |
|                 | releasechar         | The char used as the release character e.g. the char "?". All characters in the input at the right side of a release character are ignored.  | Yes       | Not used |                       | N/A           |
|                 | decimalchar         | The char used to separate numbers in an integer part from the fractional part. Default value is "." But it is possible to use "," instead.   | No        | No       | "," Or "." and others | ."            |
|                 | subelementseparator | The char used as sub-element separator e.g. the char ":".  | Yes       | Not used |                       | N/A           |
|                 | elementseparator    | The char used as element separator e.g. the char "+".  | Yes       | Not used |                       | N/A           |
|                 | segmentterminator   | The char used as segment terminator e.g. the char "'" (apostrophe).  | Yes       | Not used |                       | N/A           |
|                 | roottag             | The root XML tag used as the very first tag in the output e.g. "ORDER" or "DOCUMENT". If the root tag is defined in the AXT configuration, it will override this setting.                              | No        | No       | Valid XML tag name    | Document name |

| Tag            | Attribute  | Meaning  | Required  |       | Possible values    | Default value |
|----------------|------------|--|-----------|-------|--------------------|---------------|
|                |            |  | Non-fixed | Fixed |                    |               |
| <b>Segment</b> | name       | The name of the document definition e.g. UNA. The parser matches the name against in the input.  | Yes       | Yes   |                    | N/A           |
|                | xmlname    | A xml name can be given as the XML markup for the output. If none is given the name of the segment name will be used as the xml-tag.                               | No        | No    | Valid XML tag name | Segment name  |
|                | minrow     | This attribute informs the parsing that there at least shall occur the specified number of these segments in the input.  | No        | No    | 0, 1, 2,...        | 0             |
|                | maxrow     | This attribute informs the parsing that there at most must occur the specified number of these segments in the input number '*' means an unlimited amount of rows. | Yes       | Yes   | 1, 2, 3,... or *   | N/A           |
|                | segmentdef | Filename of the current segment definition.  | Yes       | Yes   | A file name        | N/A           |

In the above table the **segmentdef** attribute has to be a valid file containing segment definitions.

It is possible to use environment variables in the file naming. Notice that file names are platform specific.

## 7 Segments and elements

### 7.1 Record specification from a FIXED position file

The following is an example of 'Line' record specification from a fixed position file. It shows the segment and element definition.

```

<EDI_SEGMENT>
  <segment name="R" start="1" length="1" >
    <element name="type" start="1" length="2" mandatory="true"/>
    <element name="lineno" start="3" length="5" mandatory="true"/>
    <element name="itemno" start="8" length="20" mandatory="true"/>
    <element name="description" start="28" length="40" mandatory="true"/>
    <element name="quantity" start="68" length="15" mandatory="true"/>
    <element name="unit" start="83" length="3" mandatory="false"/>
    <element name="deliverydate" start="86" length="8" mandatory="true"/>
    <element name="price" start="94" length="15" mandatory="true"/>
    <element name="currency" start="109" length="3" mandatory="false"/>
  </segment>
</EDI_SEGMENT>

```

A segment definition describes how to read a single segment. The definition is used for both reading Fixed-position and in-house EDI input format.

## 7.2 Record specification from an EDI format file

The following is an example of 'QTY' record specification and show the segment definition.

```
<EDI_SEGMENT>
  <segment name='QTY' fixed="false">
    <element name="C186" type="String" mandatory="true">
      <element name="C6063" type="String" mandatory="true"/>
      <element name="C6060" type="String" mandatory="true"/>
      <element name="C6411" type="String" mandatory="false"/>
    </element>
  </segment>
</EDI_SEGMENT>
```

The following table describes the different tags and the attributes. The root tag EDI\_SEGMENT is not described in the following table since its only use is for enclosing the different segment descriptions, which may exist in the file.

| Tag            | Attribute | Meaning  | Required  |       | Default value |
|----------------|-----------|--|-----------|-------|---------------|
|                |           |  | Non-fixed | Fixed |               |
| <b>segment</b> | name      | The name of the segment e.g. QTY. The name is not used during execution but can be specified for readability.  | No        | No    | N/A           |
|                | start     | Used when reading fixed-position format.<br>Tells the parser when to begin reading the segment name.<br>See <b>element start</b> for exact instructions.   | Not used  | No    | 1             |
|                | length    | The number of characters to read   | Not used  | No    | 1             |
| <b>element</b> | name      | The name of the XML output tag.  | Yes       | Yes   | N/A           |
|                | type      | The type of the value. Default value is <i>String</i> . For floating point numbers it must be set to <i>double</i> where the decimal character will be replaced by “.”   | Yes       | No    | String        |
|                | mandatory | Whether the element is mandatory or not. If the element is mandatory, the parser tries to locate the element in the input. If it is not found an error is reported.  | No        | No    | False         |
|                | start     | Used when reading in-house format.<br>Tells the parser where to begin reading characters in the current segment.<br>Indicates the <b>absolute</b> start position counting from the beginning of the outermost segment (currently this means from the beginning of the line). | Not used  | No    | 1             |
|                | length    | The number of characters to read   | Not used  | No    | 1             |

## 8 Examples

### 8.1 Example INPUTfile FIXED position.

```

EDIMGR:9999123456789:77777000000:ORDERS:
H 021220-1      03-04-207380990000000999123456789
Bordinglunden 22      Marmelade
R 001 123001      Juice
R 002 123003      Biscuit
R 003 T6001      Water
R 004 123005      Milk
R 005 123002

          9999 Postadress
          9999123456789
          LTR03-04-307.50
          LTR03-04-3015.00
          PCE03-04-3010.00
          LTR03-04-3014.00
          LTR03-04-309.00

          2.00
          20.00
          30.00
          10.00
          1.00

          TestCompagny
          9999123456789
          LTR03-04-307.50
          LTR03-04-3015.00
          PCE03-04-3010.00
          LTR03-04-3014.00
          LTR03-04-309.00

          ECU
          ECU
          ECU
          ECU
          ECU
  
```

## 8.2 Example INPUTfile EDI Invoice

UNA:+.? '  
UNB+UNOC:3+7300070000003:14+7350000910919:14+030205:1110+11+++++1'  
UNH+0200062019+INVOIC:D:96A:UN:EAN008'  
BGM+380+111100099+9'  
DTM+137:20030205:102'  
DTM+35:20030205:102'  
RFF+ON:XX'  
NAD+SU+7300070000003::9'  
RFF+VA:SE556074308901'  
NAD+BY+7350000910919::9'  
PAT+3++66:1'  
DTM+13:20030305:102'  
ALC+C++6++FC'  
MOA+8:50'  
TAX+7+VAT+++:::25+S'  
MOA+124:12.5'  
LIN+1++7300070000010:EN'  
QTY+47:7.17:KGM'  
QTY+46:1'  
MOA+203:668.24'  
PRI+AAA:93.2:CT::1:KGM'  
RFF+ON:xx:x'  
LIN+2++7300070000058:EN'  
QTY+47:10'  
MOA+203:735'  
PRI+AAA:73.5:CT'  
RFF+ON:xx:x'  
LIN+3++7300070000065:EN'  
QTY+47:10'  
MOA+203:712.8'  
PRI+AAA:71.28:CT'  
RFF+ON:xx:x'  
UNS+S'CNT+1:28.17'CNT+2:3'  
MOA+9:2340.14'  
MOA+131:50'  
MOA+79:2116.04'  
MOA+176:224.1'  
MOA+125:2116.04'  
TAX+7+VAT+++:::12+M'  
MOA+125:2116.04'  
MOA+124:211.6'  
TAX+7+VAT+++:::25+S'  
MOA+125:50'  
MOA+124:12.5'  
UNT+45+0200062019'  
UNZ+1+11'