

Table of content

Table of content.....	1
XML description.....	2
DTD and instructions to interface between Heating and Sanitation Database Server and Manufacturer System	2
Document History	2
Character Representation	2
DTD for XML-data	3
Instructions	4
Escape Sequences	5
Escape Sequences inside the XML document.....	6
Signatures	6
Data Validation	6
The meaning of the XML elements	7
Technical validation.....	9
Interface between Database Server and Manufacturer System.....	10
Document History	10
Instructions	10
HTTP Header	10
Inquiry.....	11
Response to inquiry.....	13
Message codes	14
Exceptions.....	14
Other receipts	14
Order.....	15
Response to order	16
Message Codes	17
Exceptions.....	17
Other receipts	17
Examples of valid Requests and Responses.....	19
Document History	19
Instructions	19
Inquiry.....	20
Manufacturer response to request	22
Manufacturer response to order	27

Paragraph 13

XML description

Consists presently of 3 sections:

- DTD and instructions to interface between Heating and Sanitation Database Server and Manufacturer System.
- Interface between Heating and Sanitation Database and Manufacturer System.
- Examples of Request and Response.

DTD and instructions to interface between Heating and Sanitation Database Server and Manufacturer System

Version date: 20030613

Document History

This document is a "working" document – meaning that it will continuously appear in new revised versions. In the table below, it is stated what the changes will be in coming versions of the document.

Version date	New in the document
20000221	Start, draft – used for 'seeing' what XML and DTD are
20010519	First version of DTD definition
20010814	Added Item text, exchange, unit quantity to Item in the DTD definition. Normalised date, time and user elements.
20011217	The login element is not required.
20020102	In case an element has none data the notation is by a single tag <name/> and not by a start and end tag <name></name>.
20020130	Error in DTD description corrected
20020522	Support 1-n items. More thoroughly explanation how the XML is sent and received in Heating and Sanitation database. Escape sequence in XML #PCDATA types. Description messages sent to end-user.
20020610	Specify the difference between message.code and item.code. Specify transportation tag in different context. delivery.date always have to be the desired delivery date.
20021010	Specification in "The meaning of the XML elements ": Login is mandatory and item elements mutual coherence.
20030613	The request value for Item.code is always -1.

Character Representation

The communication takes place with an ISO 8859-1 character set. The DTD file will also be represented by ISO 8859-1 characters.

DTD for XML-data

A DTD is a central document in connection to XML documents of the type XML version 1.0. DTD stands for Document Type Definition, and it is a document that describes which types of elements that can be found in XML documents referring to this DTD, and what reciprocal structure these can be placed in.

An XML document refers to the appurtenant DTD by use of a special <DOCTYPE ..> tag, which can be seen in the formats specified below.

As a starting point, you might say that a DTD document is only interesting if you want to check the validity of an XML document – and as such, it has no effect on the contents of the XML document.

If third-party XML parsers (software that can change/move data in an XML document into something else) are used, some of these will require access to the DTD document, but if you are making the parsing of the XML document yourself, the DTD document can be ignored.

When there are no marks after an element, the element occurs once and only once. A ? after an element indicates that the element may be included and if so, it only occurs once. An * after an element indicates that the element may be included and if so, it occurs once or several times. A + after an element indicates that the element is included and that it occurs once or several times.

```
<!--This is the DTD for the XML data exchange between VVS Database Server and the manufactors
System -->
<!ELEMENT VVSorder (deploymentmode?, creator?, login, inquiry?, inquiryresponse?, order?,
orderresponse?)>
<!ELEMENT login (user, passwd)>
<!ELEMENT inquiry (objectid, transaktion, receiver, item+, sender, deliverto?, transportation?,
delivery, send, ordernumber)>
<!ELEMENT inquiryresponse (objectid, transaktion, message, item+, delivery?, comments?)>
<!ELEMENT order (objectid, transaktion, receiver, item+, sender, deliverto?, transportation?, delivery,
send, ordernumber, exchange?)>
<!ELEMENT orderresponse (objectid, transaktion, message, item+)>
<!ELEMENT receiver (location, comments?)>
<!ELEMENT sender (location, fetch, comments?, reference?)>
<!ELEMENT deliverto (location)>
<!ELEMENT delivery (handlingcharge?, transportprice?, date, transportation?)>
<!ELEMENT location (EANlocation, name, address, postalcode, postalcity, countrycode)>
<!ELEMENT send (date, time, user)>
<!ELEMENT item (itemid, quantity, unit, unitquantity, price, text?, exchange, code, vvsnumber?)>
<!ELEMENT message (code, text?)>
<!ELEMENT deploymentmode (#PCDATA)>
<!ELEMENT creator (#PCDATA)>
<!ELEMENT user (#PCDATA)>
<!ELEMENT passwd (#PCDATA)>
<!ELEMENT objectid (#PCDATA)>
<!ELEMENT transaktion (#PCDATA)>
<!ELEMENT comments (#PCDATA)>
<!ELEMENT fetch (#PCDATA)>
<!ELEMENT reference (#PCDATA)>
<!ELEMENT EANlocation (#PCDATA)>
```

```
<!ELEMENT name (#PCDATA)>
<!ELEMENT address (#PCDATA)>
<!ELEMENT postalcode (#PCDATA)>
<!ELEMENT postalcity (#PCDATA)>
<!ELEMENT countrycode (#PCDATA)>
<!ELEMENT itemid (#PCDATA)>
<!ELEMENT quantity (#PCDATA)>
<!ELEMENT unit (#PCDATA)>
<!ELEMENT price (#PCDATA)>
<!ELEMENT handlingcharge (#PCDATA)>
<!ELEMENT transportprice (#PCDATA)>
<!ELEMENT date (#PCDATA)>
<!ELEMENT transportation (#PCDATA)>
<!ELEMENT time (#PCDATA)>
<!ELEMENT ordernumber (#PCDATA)>
<!ELEMENT code (#PCDATA)>
<!ELEMENT text (#PCDATA)>
<!ELEMENT unitquantity (#PCDATA)>
<!ELEMENT exchange (#PCDATA)>
<!ELEMENT vvsnumber (#PCDATA)>
```

Instructions

In the following when HTTP is mentioned, the term also covers HTTPS. The standard HTTP protocol is used. Notice that the examples have one parameter per line for the sake of readability, but in the actual communication these have to be concatenated into one line separated by '&'.

The indentation shown in response-data is merely there to give a better overview and it will not be present in the actual data packages.

When using special characters, the traditional URL escape sequences have to be used.

The HTTP method POST is used for all request types, which contribute to a uniform cut across the individual requests. The POST method indicates that all data is sent in the body of the http request; the body is identified as the part of the request that follows the http-header lines and a single blank line! (See the http specification).

There are two options when sending XML strings:

1. "Pure" XML string

HTTP Header Content-Type: text/xml

The manufacture system reads the XML directly from the request.

REMEMBER that the Manufacture system should have a set-up so that it is able to receive a HTTP request with Content-Type text/xml

2. The XML string will be send like a value on a attribute, XML=.....

HTTP Header Content -Type: application/x-www-form-urlencoded. The manufacture system reads the XML by getting (getParameter) the value of the parameter named "XML".

Using this option it is furthermore possible to send the XML string like it is or like urlencoded (in accordance with the Content-Type in the Header).

The manufacture decides the option to use, this is done by a contact to LEC AS. The default option is number 1 ("pure" XML).

Escape Sequences

Escape sequence is the term used for the codes that are enclosed in an HTTP document to replace certain characters that are used as special characters with a specific meaning. These characters, and the string they replace, are listed below.

Aside from these special characters, all characters with ASCII over 127 have to be replaced with %XX where XX is the hexadecimal representation of the ASCII value. This is also true in the case of the Danish letters æ, ø and å.

Character	Escape code	Character	Escape code
SPACE	%20	<	%3C
>	%3E	#	%23
%	%25	{	%7B
}	%7D		%7C
\	%5C	^	%5E
~	%7E	[%5B
]	%5D	~	%60
;	%3B	/	%2F
?	%3F	:	%3A
@	%40	=	%3D
&	%26	\$	%24
+	%2B	æ	%E6
ø	%F8	å	%E5

Æ	%C6	∅	%D8
Å	%C5		

Escape Sequences inside the XML document.

The text attributes (#PCDATA) in the XML string can hold all the characters according to ISO-8859-1, however Escapes the following characters:

Character	Name	Escape Code
&	Amp	& or &
<	Lt	< or <
>	Gt	> or >

Signatures

In the specification of the data formats below, the following signatures are used:

Signature	Description	Valid values
B	Boolean	[Y,N]
±	Sign	[+,-]
9	Numeric	[0,1,2,3,4,5,6,7,8,9]
9,99	Numeric decimal	[0,1,2,3,4,5,6,7,8,9 and ',']
X	Alphanumeric	[0-9,a-å,A-Å]
CCYYMMDD	Date	A valid date, for instance: 20011001
HHMMSS	Time	A valid (24 hours) time, for instance: 183654

Data Validation

-
-
- The transaction attribute contains 4 digits zzzz where 0010 is equal to Inquiry transaction and 0020 is an Order transaction.
- The message attribute contains 4 digits xxxx where xxxx is code for the message type. xxxx<1000 is message (the transaction is completed) and xxxx>1000 is error (the transaction is not completed)
- Request transaktionnumber and response transaktionnumber shall be the same.
- Request Objectld and response Objectld shall be the same.

The meaning of the XML elements

Here is a list of elements from DTD and a short description of its contents. An element can be a 'detail' element or a 'master' element, which is built on one or several other elements. It may be required that a specific element is included. If a 'detail' element is part of a 'master' element, the requirement is only relevant if the 'master' element also has to be included.

ELEMENT	Meaning	Required
order	Consists of the elements deploymentmode?, creator?, login?, inquiry?, inquiryresponse?, order?, orderresponse?	Yes
login	Consists of the element login.user and login.passwd.	Yes
inquiry	Consists of the elements objectid, transaction, receiver, item+, sender, deliverto?, transportation?, delivery, send, ordernumber	No
inquiryresponse	Consists of the elements objectid, transaction, message, responsitem?, delivery?,comments?	No
order	Consists of the elements objectid, transaction, receiver, item+, sender, deliverto?, transportation?, delivery, send, ordernumber	No
orderresponse	Consists of the elements objectid, transaction, message	No
receiver	Consists of the elements location, comments?	Yes
sender	Consists of the elements location, fetch, comments?, reference?	Yes
deliverto	Consists of the elements EANlocation	No
delivery	Consists of the elements handlingcharge?, transportprice?, date, transportation?	No
send	Consists of the elements date, time, user	Yes
item	Consists of the elements itemid, quantity, unit, price In the current version, the inquiry will only deal with one product number, but future versions may have the option of handling several product numbers simultaneously.	Yes
location	Consists of the elements EANlocation, name, address, postalcode, postalcity, countrycode	Yes
message	Consists of the element code and text?	Yes
deploymentmode	Indicates whether a request occurs as production or test. As default, a request is always "production". To indicate a test, the tag has to be included with the value "test".	No
creator	Identification of the installation/application that has created the XML file. May be relevant in future scenarios when the XML file is sent/forwarded between several different installations/applications.	No

login.user	Request: User logon to Manufacturer system. Value exists if there is a value for the element to disposal. Respon: Mandatory, but a value do not have to exist.	Yes
login.passwd	The request: User password to Manufacturer system. Value exists if there is a value for the element to disposal. The response: Mandatory, but a value do not have to exist.	Yes
objectid	Unique unequivocal identification of the XML transaction request. The request objectid tag and its value are mandatory in the response to secure that a request and its response always can be matched in the database. Secures that request and response can always be 'matched' in the Heating and Sanitation Database.	Yes
transaktion	Number identifying transaction type Currently there are two types: Inquiry (0010) and Order (0020)	Yes
EANlocation	Location number	Yes
name	Location name	Yes
address	Location address	Yes
postalcode	Location area code	Yes
postalcity	Location city	Yes
countrycode	Location country code	Yes
comments	Comments	No
item.itemid	Product number. A value is mandatory.	Yes
item.quantity	Quantity – desired quantity in Inquiry transaction, possible quantity in inquiryresponse. If item.price have a value this element also have to have a value.	Yes
item.unit	Quantity unit. If item.price have a value this element also have to have a value.	Yes
item.price	In Inquiry it is blank. In inquiryresponse, it states the manufacturer's net price, which is then included in Order transaction	Yes
item.exchange	The item exchange. If item.price have a value this element also have to have a value.	Yes
fetch	Sender handles pick-up. Can be 'Y' for yes or 'N' for no.	Yes
reference	Sender reference. It is the 'Custumorreferencenumber'	No
delivery.transportation	Sender's information about transportation.	No
transportation	For element inquiry and order this tag is ignored. Delivery.transportation is used.	No
handlingcharge	Handling fee	No
transportprice	Transport costs	No
delivery.date	Desired delivery date in Inquiry transaction. Keep it as desired delivery date in Inquiryresponse and Order transaction.	Yes
code	Return code stating status with Manufacturer	Yes
item.code	Return code with the manufactures delivery date or -1. If there is 1-n item with a delivery date, they all have to have the	Yes

	same date. This date should be the date from the item with the longest delivery time.	
message.text	Text for clarifying/supplementing the return code	No
item.text	Item text used in the creation of the EDIFACT copy order	No
send.date	Dispatch date	Yes
send.time	Dispatch time	Yes
send.user	Dispatch user id	Yes
ordernumber	Dispatch has order number attached. Blank on Inquiry.	Yes
order.exchange	The order exchange	No
unitquantity	Item unit quantity. The quantity for the Item unit. Is a part of the definition of the unit. Forexsampel and order of 3 a '50' kilo, where the '50' is the unitquantity and 3 is the quantity.	Yes
item.vvsnumber	The VVS number (not in use)	No

Technical validation

In relation with the data transport across the Internet and the parsing of data, the Heating and Sanitation database will catch the following exceptions:

Exception	Receipt on the screen
javax.net.ssl.SSLException	Problem in HTTPS transport.
java.security.AccessControlException	A missing permission somewhere.
java.net.MalformedURLException	URL is not constructed in a proper way.
java.io.InterruptedIOException	No response from the manufacture.
java.io.IOException	Communication error.
java.lang.IllegalArgumentException	Error in input parameters.
org.xml.sax.SAXException	XML parser reports an error.

Interface between Database Server and Manufacturer System

Version date: 20020522

Document History

This document is a "working" document – meaning that it will continuously appear in new revised versions. In the table below, it is stated what the changes will be in coming versions of the document.

Version date	New in the document
20010221	Start draft for 'seeing' what XML and DTD are. Descriptions are incomplete
20010519	First version
20010814	Added unit quantity to Item element. Normalised date, time and user elements. Changed <sendt> to <send> according to the DTD.
20020102	In case an element has none data the notation is by a single tag <name/> and not by a start and end tag <name></name>. VVSorter element is part of the examples.
20020130	Correction in XML description
20020522	Supports 1-n order lines. Supports several sending options (see "HTTP Header").

Instructions

Instructions on how to read this document can be found in the document Instructions for the XML interface between Heating and Sanitation Database Server and Manufacturer System

HTTP Header

The chosen sending option influence how the HTTP Header looks like. It can look like the following.

ex.:

- 1) When XML string is send "pure"
 - Content-Length = 2305
 - Content-Type = text/xml
 - Cache-Control = no-cache
 - Pragma = no-cache
 - User-Agent = Java1.2.2
 - Host = localhost:8080
 - Accept = text/html, image/gif, image/jpeg, *; q=.2, */*; q=.2

Message codes

For a general description of message codes see the section about data validation. This list of codes is expected to be elaborated in line with the physical integration with the Manufacturer System.

Trans- action no.	Message	Meaning
0010		
	0000	OK – The transaction was successful
	1000	ERR – Error in connection with reading the transaction - tag name of the last read tag is placed in the text section
	1001	ERR – The transaction is received and is correct, but not executed in the Manufacturer System.
	1003	ERR – The transaction is received and is correct, but the Manufacturer System is not available
	1007	ERR - Login not accepted

Above-mentioned message codes results in the following receipt on the screen.

Trans- action no.	Message	Receipt on the screen
0010		
	1000	Fejl i vvsorder transaktion 0010: " + content of inquiryresponse.message.text
	1001	Forespørgsel er ikke udført hos Fabrikant
	1003	Fabrikantens system er ikke tilgængeligt
	1007	Login til Fabrikant system blev afvist

Exceptions

Field	Content	Meaning
item.code	-1	RING

Other receipts

Further more there is some more technical oriented receipt (Java exceptions). See them under "Technical validation".

Message Codes

For a general description of message codes read the section on data validation.

Trans- action no.	Mess age	Meaning
0020		
	0000	OK – The transaction was successful
	1000	ERR – Error in connection with reading the transaction – tag name of last read tag is placed in the text section
	1001	ERR – The transaction is received and is correct, but not executed in the Manufacturer System
	1003	ERR – The transaction is received and is correct, but the Manufacturer System is not available
	1007	ERR – Login not accepted

Above-mentioned message codes results in the following receipt on the screen.

Trans- action no.	Messa ge	Receipt on the screen
0010		
	1000	Fejl i vvsorder transaktion 0020: " + content of inquiryresponse.message.text
	1001	Ordre er ikke udført hos Fabrikant
	1003	Fabrikantens system er ikke tilgængeligt
	1007	Login til Fabrikant system blev afvist

Exceptions

Field	Content	Meaning
item.code	-1	RING

Other receipts

Further more there is some more technical oriented receipt (Java exceptions). See them under "Technical validation".

Special validation

The synchronisation between the request and the response is defined by the following:

1. Number of lines (order.item elementer) recieved in the database from an order response shall be the same as the number of lines requested.
2. The manufacture itemnumber (order.item.itemid element) received in the database for order line 1 shall be the same as the manufacture number sends in order line 1. Item number received in order line 2 shall be the same as the number sent in order line 2 .. And so on.

Condition	Receipt on screen

1	receipt lines > sent lines	Ordre gennemført med flere varelinier end ønsket
1	receipt lines < sent lines	Ordre gennemført med færre varelinier end ønsket
2	receipt itemid != sent itemid	"Ordre gennemført med fejl i ordrelinie "+linienummer

Under all circumstances there will be created an EDI copy order, if the database is marked to create a copy order to the wholesaler.

Examples of valid Requests and Responses

Version date: 20020522

Document History

This document is a “working” document – meaning that it will continuously appear in new revised versions. In the table below, it is stated what the changes will be in coming versions of the document.

Version date	New in the document
20010221	Start – draft for giving an example of what XML and DTD are. Descriptions are incomplete
20010519	First version
20010814	Added unit quantity to Item element. Normalised date, time og user tags. Changed <sendt> to <send> according to the DTD.
20020102	In case an element has none data the notation is by a single tag <name/> and not by a start and end tag <name></name>. VVSorter element is part of the examples.
20020130	Correction in XML description
20020522	Supports 1-n item elements
20030613	The request value for Item.code is always -1.

Instructions

Access to the Manufacturer System takes place through the HTTP/HTTPS protocol.

Below are examples of both requests and responses for each of the transaction types that are supported. Notice that presently they have not been tested physically, so the examples are hypothetical.

For sending the request, this example uses MS Internet Explorer 5.0, and for sending the response, MS Internet Information Server 4.0 is used. These choices influence the header-lines User-agent and Server, while the rest are not affected by which types of application are communicating via HTTP.

Inquiry

Request

```
Content-Length = 2305
Content-Type = text/xml
Cache-Control = no-cache
Pragma = no-cache
User-Agent = Java1.2.2
Host = localhost:8080
Accept = text/html, image/gif, image/jpeg, *; q=.2, */*; q=.2
```

Or

```
Content-Length = 2309
Content-Type = application/x-www-form-urlencoded
Cache-Control = no-cache
Pragma = no-cache
User-Agent = Java1.2.2
Host = localhost:8080
Accept = text/html, image/gif, image/jpeg, *; q=.2, */*; q=.2
```

```
<?xml version="1.0" encoding="iso-8859-1"?>
<!DOCTYPE VVSorder PUBLIC "VVSorder" "d:/vvs/resources/VVSxml.dtd">
<VVSorder>
  <creator>5790000010714CVT</creator>
  <deploymentmode>test</deploymentmode>
  <login>
    <user/>
    <passwd/>
  </login>
  <inquiry>
    <objectid>CEYN04781</objectid>
    <transaktion>0010</transaktion>
    <receiver>
      <location>
        <EANlocation>5711000000000</EANlocation>
        <name>Test Fabrikant</name>
        <address>Fabrikvej 3</address>
        <postalcode>8000</postalcode>
        <postalcity>Aarhus C</postalcity>
        <countrycode>Danmark</countrycode>
      </location>
      <comments/>
    </receiver>
    <item>
      <itemid>vareid1</itemid>
      <quantity>12,00</quantity>
      <unit>PCE</unit>
      <unitquantity/>
      <price/>
```

```
<text>varetekst1</text>
<exchange>DDK</exchange>
<code>-1</code>
<vvsnumber/>
</item>
<item>
  <itemid>vareid2</itemid>
  <quantity>8,00</quantity>
  <unit>PCE</unit>
  <unitquantity/>
  <price/>
  <text>varetekst2</text>
  <exchange>DDK</exchange>
  <code>-1</code>
  <vvsnumber/>
</item>
<item>
  <itemid>vareid3</itemid>
  <quantity>16,00</quantity>
  <unit>PCE</unit>
  <unitquantity/>
  <price/>
  <text>varetekst3</text>
  <exchange>DDK</exchange>
  <code>-1</code>
  <vvsnumber/>
</item>
<sender>
  <location>
    <EANlocation>5711000000001</EANlocation>
    <name>Test Grossist</name>
    <address>Grossistvej 5</address>
    <postalcode>8000</postalcode>
    <postalcity>Aarhus C</postalcity>
    <countrycode>Danmark</countrycode>
  </location>
  <fetch>N</fetch>
  <comments/>
  <reference/>
</sender>
<deliverto>
  <location>
    <EANlocation>5790000000002</EANlocation>
    <name>Afdeling Nord</name>
    <address>Grossistvej 7</address>
    <postalcode>8200</postalcode>
    <postalcity>Aarhus N</postalcity>
    <countrycode>Danmark</countrycode>
  </location>
</deliverto>
```

```

<transportation/>
<delivery>
  <handlingcharge/>
  <transportprice/>
  <date>20020425</date>
  <transportation>De Danske Fragtmænd</transportation>
</delivery>
<send>
  <date>20020424</date>
  <time>1352</time>
  <user>ROES02</user>
</send>
<ordernumber/>
</inquiry>
</VVSorder>

```

Manufacturer response to request

Response

```

<?xml version="1.0" encoding="UFT-8"?>
<!DOCTYPE VVSorder SYSTEM "d:/vvs/resources/VVSxml.dtd">
<VVSorder>
  <deploymentmode>test</deploymentmode>
  <creator>LEC AS Fabrikant simulering</creator>
  <login>
    <user>login_user</user>
    <passwd>login_passwd</passwd>
  </login>
  <inquiryresponse>
    <objectid>INQ1</objectid>
    <transaktion>0010</transaktion>
    <message>
      <code>0000</code>
      <text>inquerytekst</text>
    </message>
    <item>
      <itemid>99999999999999999999</itemid>
      <quantity>20</quantity>
      <unit>STK</unit>
      <unitquantity>50</unitquantity>
      <price>25,25</price>
      <text>Vare tekst</text>
      <exchange>DDK</exchange>
      <code>-1</code>
      <vvsnumber/>
    </item>
    <item>
      <itemid>99999999999999999998</itemid>

```


Order
Request

```
Content-Length = 2522
Content-Type = text/xml
Cache-Control = no-cache
Pragma = no-cache
User-Agent = Java1.2.2
Host = localhost:8080
Accept = text/html, image/gif, image/jpeg, *, q=.2, */*; q=.2
```

Or

```
Content-Length = 2526
Content-Type = application/x-www-form-urlencoded
Cache-Control = no-cache
Pragma = no-cache
User-Agent = Java1.2.2
Host = localhost:8080
Accept = text/html, image/gif, image/jpeg, *, q=.2, */*; q=.2
```

```
<?xml version="1.0" encoding="iso-8859-1"?>
<!DOCTYPE VVSorder PUBLIC "VVSorder" "d:/vvs/resources/VVSxml.dtd">
<VVSorder>
  <deploymentmode>test</deploymentmode>
  <creator>5790000010714CVT</creator>
  <login>
    <user/>
    <passwd/>
  </login>
  <order>
    <objectid>CEYN5T265</objectid>
    <transaktion>0020</transaktion>
    <receiver>
      <location>
        <EANlocation>5711000000000</EANlocation>
        <name>Fabrikant</name>
        <address>Fabrikantvej 1</address>
        <postalcode>8000</postalcode>
        <postalcity>Aarhus C</postalcity>
        <countrycode>Danmark</countrycode>
      </location>
      <comments>Bemærkninger til leverancen</comments>
    </receiver>
    <item>
      <itemid>vareid1</itemid>
      <quantity>12,00</quantity>
      <unit>PCE</unit>
      <unitquantity>50</unitquantity>
```

```
<price>25,25</price>
<text>varetekst1</text>
<exchange>DDK</exchange>
<code>-1</code>
<vvsnumber/>
</item>
<item>
  <itemid>vareid2</itemid>
  <quantity>8,00</quantity>
  <unit>PCE</unit>
  <unitquantity>30</unitquantity>
  <price>30,25</price>
  <text>varetekst2</text>
  <exchange>DDK</exchange>
  <code>-1</code>
  <vvsnumber/>
</item>
<item>
  <itemid>vareid3</itemid>
  <quantity>16,00</quantity>
  <unit>PCE</unit>
  <unitquantity>40</unitquantity>
  <price>40,25</price>
  <text>varetekst3</text>
  <exchange>DDK</exchange>
  <code>-1</code>
  <vvsnumber/>
</item>
<sender>
  <location>
    <EANlocation>5711000000001</EANlocation>
    <name>Test Grossist</name>
    <address>Grossist vej 1</address>
    <postalcode>8000</postalcode>
    <postalcity>Aarhus</postalcity>
    <countrycode>Danmark</countrycode>
  </location>
  <fetch>N</fetch>
  <comments/>
  <reference>LECREO</reference>
</sender>
<deliverto>
  <location>
    <EANlocation>5790000000002</EANlocation>
    <name>Afdeling Nord</name>
    <address>Grossistvej nord 1</address>
    <postalcode>8200</postalcode>
    <postalcity>Aarhus N</postalcity>
    <countrycode>Danmark</countrycode>
  </location>
```

```
</deliverto>
<transportation/>
<delivery>
  <handlingcharge>300,00</handlingcharge>
  <transportprice>250,00</transportprice>
  <date>20020423</date>
  <transportation>De Danske Fragtmænd</transportation>
</delivery>
<send>
  <date>20020424</date>
  <time>1357</time>
  <user>ROES02</user>
</send>
<ordernumber>ordre nr 1</ordernumber>
<exchange/>
</order>
</VVSorder>
```

Manufacturer response to order

Response

```
<?xml version="1.0" encoding="UFT-8"?>
<!DOCTYPE VVOrder SYSTEM "d:/vvs/resources/VV5xml.dtd">
<VVOrder>
  <deploymentmode>test</deploymentmode>
  <creator>LEC AS Fabrikant simulering</creator>
  <login>
    <user>login_user</user>
    <passwd>login_passwd</passwd>
  </login>
  <orderresponse>
    <objectid>ORD1</objectid>
    <transaktion>0020</transaktion>
    <message>
      <code>0000</code>
      <text>ordertekst</text>
    </message>
    <item>
      <itemid>99999999999999999999</itemid>
      <quantity>20</quantity>
      <unit>STK</unit>
      <unitquantity>50
      </unitquantity>
      <price>25,25</price>
      <text>Varetekst</text>
      <exchange>DDK</exchange>
      <code>20021231</code>
      <vvsnumber/>
    </item>
    <item>
      <itemid>99999999999999999998</itemid>
      <quantity>30</quantity>
      <unit>STK</unit>
      <unitquantity>30</unitquantity>
      <price>30,25</price>
      <text>Varetekst30</text>
      <exchange>DDK</exchange>
      <code>-1</code>
      <vvsnumber/>
    </item>
  </orderresponse>
</VVOrder>
```