

# Implementation Instruction Data Exchange T&T Crop Protection Products

---

<b>Topic</b>	: Implementation instruction
<b>Author</b>	: Annika Goudswaard (ADAMA), Conny Graumans (AgroConnect)
<b>Date</b>	: 19nov 2019
<b>Status</b>	: v19p03

---

Document history:

- 14may18, concerning the xsd's: added to TradePartyType: CityNamePostBox, CountryCodePostBox.
  - 14may18: added to enumeration textFunctionTypeCode: 107= AcknowledgementRemark, 108= ConfirmationRemark.
  - 14may18: added instruction for Acknowledgement-message and for Confirmation-message.
  - 29mei18: added "MessagePurposeCode" in the header, to indicate whether it concerns a test message or a production message.
  - 21aug18: added reading guideline for the different documents of the implementation instructions.
  - 2sep19: added overview code lists.
  - 12nov19: added instruction for the use of SSCC on the LogisticUnit.
  - 19nov19: modified the instruction for the acknowledgment of the sales and transport order; it is no longer mandatory to return the complete order message in the acknowledgement message, returning of a reference to the order-ID will do.
- 

## 1. Introduction

This document contains assumptions, remarks, explanations that are relevant for the implementation of digital exchange of Orders and DespatchAdvices in the crop protection products supply chain.

## 2. Reading guideline

The implementation instruction consists of several documents:

- 1) Implementation Instruction Data Exchange T&T Crop Protection Products  
That is this document.
- 2) Sequence diagrams T&T CPP.zip  
Contains a set of sequence diagrams, describing the data and product exchange between the different actors in the supply chain for different uses cases.
- 3) Class diagrams T&T CPP  
Data model describing the class diagrams for the Order and for the Despatch Advice (Delivery) messages and contains the definitions of all relevant classes and data elements. The class diagrams are used as a basis for specifying the standard xml (or Json) messages.
- 4) Information analysis T&T CPP.eap  
Enterprise Architect file containing the sequence diagrams and class model.
- 5) Xsd's and examples T&T CPP.zip  
Contains the xml schema definitions (xsd's) and xml examples of orders and despatch advice messages.
- 6) Excel overview of the structure of the Order message.

## 3. Assumptions and requirements

### 3.1. CRISTAL input

Important input for specifying the standard Order and Despatch Advice messages are the CropLife CRISTAL-documents (Communicating Reliable Information Systems To Agriculture and Logistics):

- CRISTAL-COMMON-PRACTICES-FOR-BAR-CODING-AND-LABELLING-OF-AGRO-PRODUCTS-1....pdf
- CRISTAL-ON-BOARDING-HANDBOOK-160117.pdf

- For these documents see: <https://croplife.org> .

### **3.2. logistic units, trade units, consumer unit.**

In the supply chain different types of units are recognized: logistic units, trade units, consumer unit.

- A consumer unit is the smallest package that can be sold to the end consumer. This, for instance, can be a single bottle or a sack of product. A ConsumerUnit is identified by a GTIN.
- A trade unit, for instance, is a box with several of the same consumer units. A trade unit is identified by either a SSCC or a GTIN. The use of SSCC at the box level (being a trade unit from the manufacturers point of view) is possible if a single box is to be shipped. In this case the SSCC on the box has to be combined with a GTIN.
- A logistic unit is an item of any composition established for transport and/or storage which needs to be managed through the supply chain. Logistic units take many shapes; it can be a single box containing a limited number of products, a pallet of multiple products, or an intermodal container containing multiple pallets. A logistic unit can be a pallet that contains several trade units and/or consumer units. A LogisticUnit must be identified by a SSCC.

### **3.3. Minimum dataset pallet for track & trace**

In order to describe a homogeneous pallet with one kind of items or batches or a heterogeneous pallet with different kind of items/batches, data to be transmitted are:

1. SSCC pallet
2. Detailed description of pallet load. For each item/batch:
  - a. Item EAN 13 (GTIN item)
  - b. Batch number
  - c. Production date
  - d. Total quantity
3. Box number
  - a. SSCC or box serial number (optional)

Remark:

- If the BL logistic part is described, details about boxes (serial number or SSCC) in the considered pallet (identified by a SSCC) must be given.
- SSCC, in the message, is optional for boxes of homogeneous pallets.
- Each box must get a mandatory serial number (can be equivalent to SSCC). The management of these numbers is not mandatory for the message receiver. Tracking is carried by the couple batch/date.
- The management of these numbers is not mandatory for the message receiver.
- Tracking is carried by the couple batch/date.

### **3.4. When to allocate a SSCC**

SSCC stands for Serial Shipping Container Code. This code is the unique identifier (serial number) of a specific transport (shipment container; box or a pallet). This unique identifier is needed to connect information about the content of that specific transport. For example information about how many boxes of a specific product are on a pallet or about how many cans of a specific product are in the box. This information can be used to prepare the goods entry. When the composition of a pallet or box changes a new SSCC needs to be created and applied.

There is always an SSCC linked to the DespatchAdvice as it is the unique number for the logistic unit (literally any means of packing) on which the product is sent. Most common is any size of pallet with product on it, but the item could also be sent as it is. If it is just a single can (consumer unit) or box (trade unit) it will still have an SSCC. The SSCC label will mention X number of consumer or trade units. The pallet does not have a GTIN code itself. Meaning if a full pallet of 30 boxes is sent, the SSCC label will mention 30 X trade unit GTIN.

### **3.5. Unique traceability details**

The 2d matrix code on the package contains:

- the batch number
- the GTIN
- the production date
- the serial number (optional)

#### Batch number

The batch number is allocated by the manufacturer and refers to the large production batch quantity, this batch number is transferred to the sales units of this specific batch number quantity.

This means that multiple sales units can have the same batch number.

A sales unit can be a trade unit (outer package, often a box) or an individual consumer unit (inner package, often a can).

#### GTIN

Each sales unit has a GS1 GTIN 14 digit number (GTIN), which is the product identifier for packed products (SKU's; Stock Keeping Units).

This means that multiple sales units of the same type of product and package have the same GTIN.

A similar product receives a new GTIN when it is sold in a different pack size or with a new trade name, this makes it a new sales unit.

In case a GTIN 13 (old name EAN13) code flat barcode for sales purposes already exists a "0" is added in front of the number to turn it into a 14 digit number, this will prevent double GTIN numbers linked to the same unit.

#### Production date

The production date is based on the packaging date of the sales unit.

If two different products are packed in one trade unit the production date placed on the box is the oldest date.

#### Serial number

Serial numbers on consumer units are optional.

Serial numbers are not required for a recall of material, as in case of a recall an entire batch needs to be taken back and not a specific consumer or trade unit of that batch.

When making use of serialisation the serial number will be added as a fourth unique value next to the GTIN, production date and batch number to each sales unit.

#### Relabelling

The combination of the GTIN, production date, batch number and optional serial number gives a unique 2D matrix code.

In case of re-labelling of the same sales unit without changing any of the details mentioned above, the traceability data can be kept.

In case of change of trade name of the same product a new GTIN is applicable, hence the 2D matrix contains new details and this change needs to be recorded in the ERP and applied on the consumer, trade and logistic units.

Without serialisation: one new 2D matrix needs to be created and applied for the whole batch volume.

With serialisation: for each consumer and trade unit a unique 2D matrix needs to be created and applied with keeping record of the previous serial number (the new data needs to be placed on the exact same package, it is not possible to pick another consumer/trade unit from the same batch).

### **3.6. Homogeneous and heterogeneous pallets**

In case that a pallet contains homogeneous boxes (each box contains exactly the same product) than it might not be necessary to unpack the pallet and to scan the SSCC of each box. The precondition for this case is that the full content of the pallet (all SSCC's of the boxes on the pallet) is communicated by a DespatchAdvice to the receiving party, in advance of the arrival of the homogeneous pallet.

### **3.7. Serialisation**

Serialisation means that each consumer unit, within the domain of the manufacturer, gets its own unique tag.

A GTIN is used to uniquely identify a type of a consumer unit. In case each individual package unit needs a unique identification a GTIN is not usable, so a serial number is required (serialisation).

## **4. Explaining details of the Order and Despatch Advice messages**

### **4.1. Difference between the commercial and logistic Despatch Advice message**

In the supply chain two versions of the Despatch Advice message are used: a commercial and a logistics version:

- the commercial Despatch Advice version from manufacturer/producer to distributor:
- the logistic Despatch Advice version goes from manufacturer/producer/distributor to logistics service provider.
- both versions contain batch information, but only the logistics version contains detailed information about the pallet load: what trade-items / consumer-items are on the pallet, complete with SSCC's and GTINs.

### **4.2. Measurement type**

For calculations and display of process, amounts and quantities a minimal number of decimals is recommended:

- Unit prices: 3 decimal places
- Amounts: 2 decimal places
- Quantities: 3 decimal places

### **4.3. Standard classifications and code lists**

Code lists:

- <http://www.gs1.se/en/our-standards/Technical-documentation/code-lists/>
- [https://www.unece.org/cefact/codesfortrade/codes\\_index.html](https://www.unece.org/cefact/codesfortrade/codes_index.html)

## **5. First pilot implementation: the ADAMA use case**

In the ADAMA use case, each pallet gets its unique SSCC. Individual boxes, for the time being, do not get a SSCC. So when a DespatchAdvice message is sent from ADAMA to the logistic service provider or to the customer, and it concerns a shipment of pallets, the SSCCs are communicated to the receiving party, using the DespatchAdvice message. So, every time products are moved from one location to another, each shipment goes with a DespatchAdvice message to inform the receiving party about the shipment that is on the way. This also goes for shipments between distribution centres of the same company, although you might want to tackle that within the ERP if both distribution centres make use of the same ERP.

**Remark:** not uniquely tagging individual boxes might have consequences for full tracking and tracing (it might break the chain of linked SSCCs and GTINs).

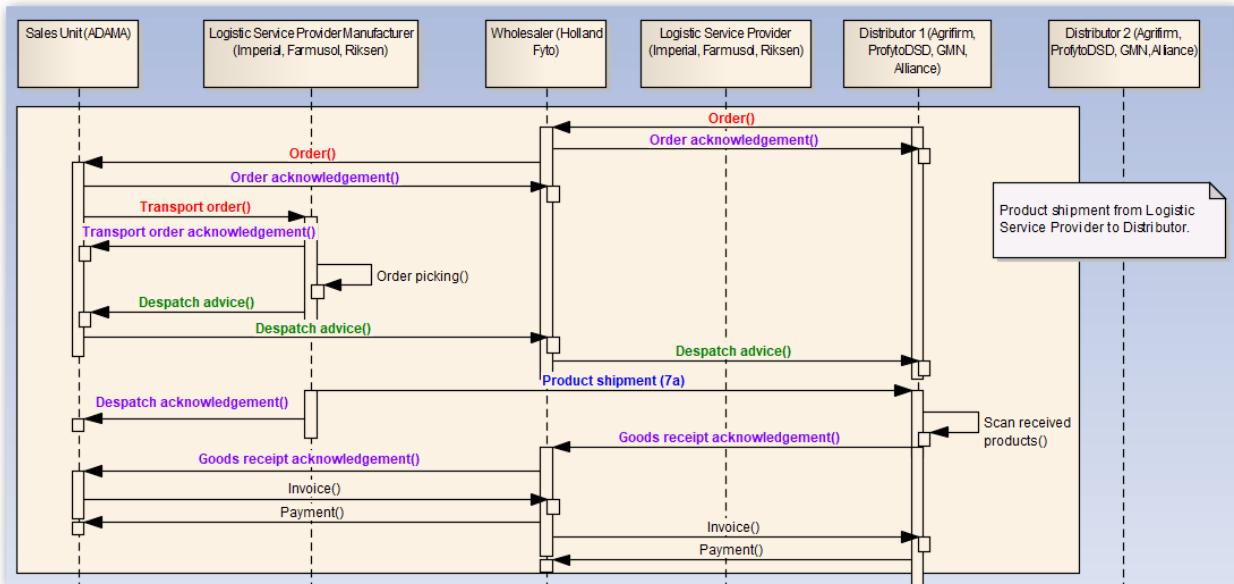
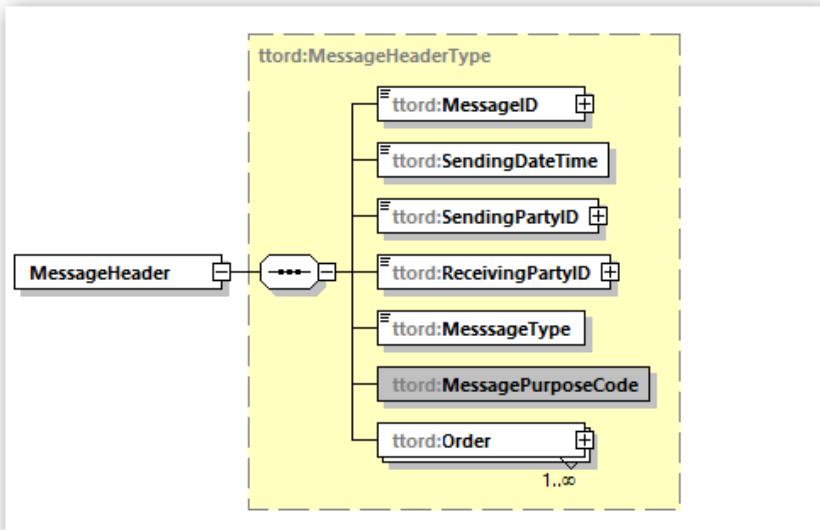


Figure 1. Sequence diagram product and data flows ADAMA use case.

### 5.1. For test or production

To indicate whether a message is a test message or a production message , MesagePurposeCode is used.



```

<xsd:simpleType name="messagePurposeCodeCode">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="47"/>
    <xsd:enumeration value="53"/>
    <!--47= for production, definitive-->
    <!--53= test-->
  </xsd:restriction>
</xsd:simpleType>
  
```

```

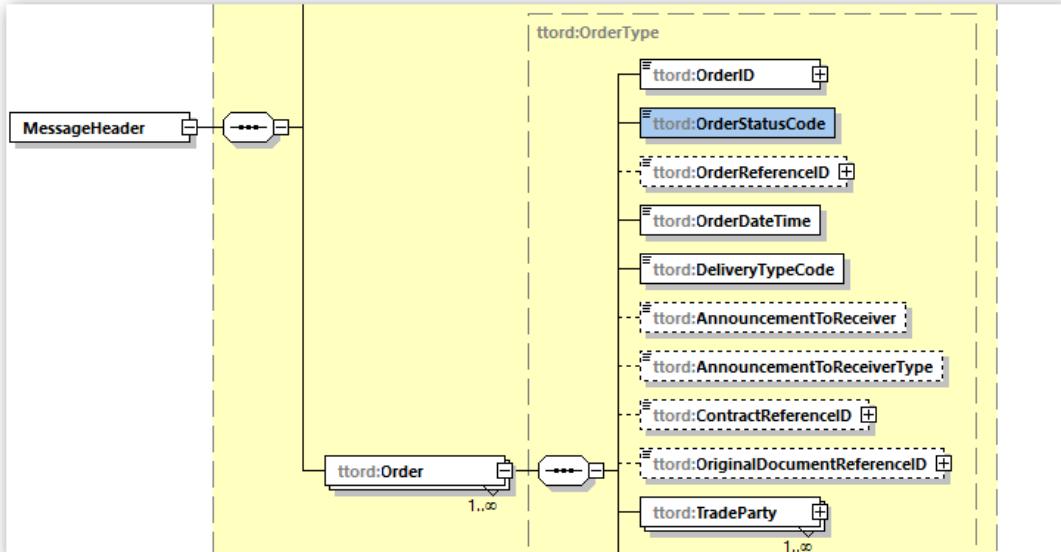
<?xml version="1.0" encoding="UTF-8"?>
<tord:MessageHeader xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:
http://www.agroconnect.nl/Portals/10/XSDs/TandT_CPP/v2018p01/TandT_CPP_Order_v201
http://www.agroconnect.nl/Portals/10/XSDs/TandT_CPP/v2018p01/TandT_CPP_Order_v201
http://www.agroconnect.nl/Portals/10/XSDs/TandT_CPP/v2018p01/TandT_CPP_Order_v201
    <tord:MessageID schemeID="REF">I00000330366</tord:MessageID>
    <tord:SendingDateTime>2018-05-14T12:43:13Z</tord:SendingDateTime>
    <tord:SendingPartyID schemeID="GLN">8719324290005</tord:SendingPartyID>
    <tord:ReceivingPartyID schemeID="GLN">8718077000008</tord:ReceivingPartyID>
    <tord:MessageType>101</tord:MessageType>
    <tord:MessagePurposeCode>53</tord:MessagePurposeCode>
<tord:Order>
    <tord:OrderID schemeID="REF">0000330366</tord:OrderID>
    <tord:OrderStatusCode>101</tord:OrderStatusCode>

```

## 5.2. Order update

When the order is updated, a new, revised order message will be send. In the new order it is that it replaces a previous order (include a reference to the previous order, and indicate by a status attribute that it is a replacement for a previous order).

To indicate that an order is a replacement of an earlier send order two special data-element are used:



The OrderStatusCode indicates whether it concerns the original document or a revised version.

In case it concerns a revised version OrderStatusCode = 102), OrderReferenceID needs to be filled with the identification (order number) of the previous order that needs to be replaced by this revised order.

The same goes for the delivery message (Despatch advice).

```

<xsd:simpleType name="documentStatusCode">
    <xsd:restriction base="xsd:string">
        <xsd:enumeration value="101"/>
        <xsd:enumeration value="102"/>
        <!--101= original-->
        <!--102= revised version-->
    </xsd:restriction>
</xsd:simpleType>

```

OrderStatusCode and OrderReferenceID might not be needed if it is agreed that when an order is send in with the same OrderID, the previously received order with this same OrderID must be replaced by this newly received version.

All items should be ordered using the GTIN to identify the ordered item. Only products with a GTIN can be ordered. An ordered product can be a consumer unit or a trade unit, indicated by ProductUnitTypeCode (102= TradeUnit, 103= ConsumerUnit).

The unit of ordered quantity should always be NBR (number) to indicate the amount (the number of items) that is ordered, and not e.g. LTR, KGM.

```
<ttord:OrderedQuantity unitCode="NBR">30</ttord:OrderedQuantity><!--betreft 30 dozen-->
<ttord:ProductUnit>
    <ttord:ProductUnitID_GTIN schemeID="GTIN">23457890</ttord:ProductUnitID_GTIN>
    <ttord:ProductUnitTypeCode>103</ttord:ProductUnitTypeCode><!--betreft ConsumerUnit-->
    <ttord:ProductUnitName>Doos met 6 flessen middel X</ttord:ProductUnitName>
</ttord:ProductUnit>
```

ADAMA prefers to have the desired delivery time in the order as a time slot with a begin and end date-time:

```
<ttord:Timing>
    <ttord:EventTypeCode>102</ttord:EventTypeCode><!-- Deliver-->
    <ttord:EventBeginDateTime>2018-03-15T09:30:47Z</ttord:EventBeginDateTime>
    <ttord:EventEndDateTime>2018-03-15T10:30:47Z</ttord:EventEndDateTime>
</ttord:Timing>
```

ADAMA prefers to receive the comments in de Order messages only at Order-level and not at OrderLine level.

Complete Business Solutions takes care of implementing the data exchange interface on the het ADAMA SAP Standard Business One environment:

- sales-orders will be pushed from the Proagrica platform to SBO.
- REST-xml is used for the exchange protocol.
- on SBO a new webservice adaptor will be developed; therefore it is not necessary to implement the Proagrica Interlock Adaptor.

## 6. Explaining the xsd's

The xml Order and DespatchAdvice messages are specified by a set xml schema definition files (xsd's).

```
TandT_CPP_DespatchAdvice_v2018p01.xsd
TandT_CPP_Order_v2018p01.xsd
TandT_DataTypes_v2018p01.xsd
UnqualifiedDataType_21p0.xsd
```

The Order and DespatchAdvice schema's import the Datatype schema's.

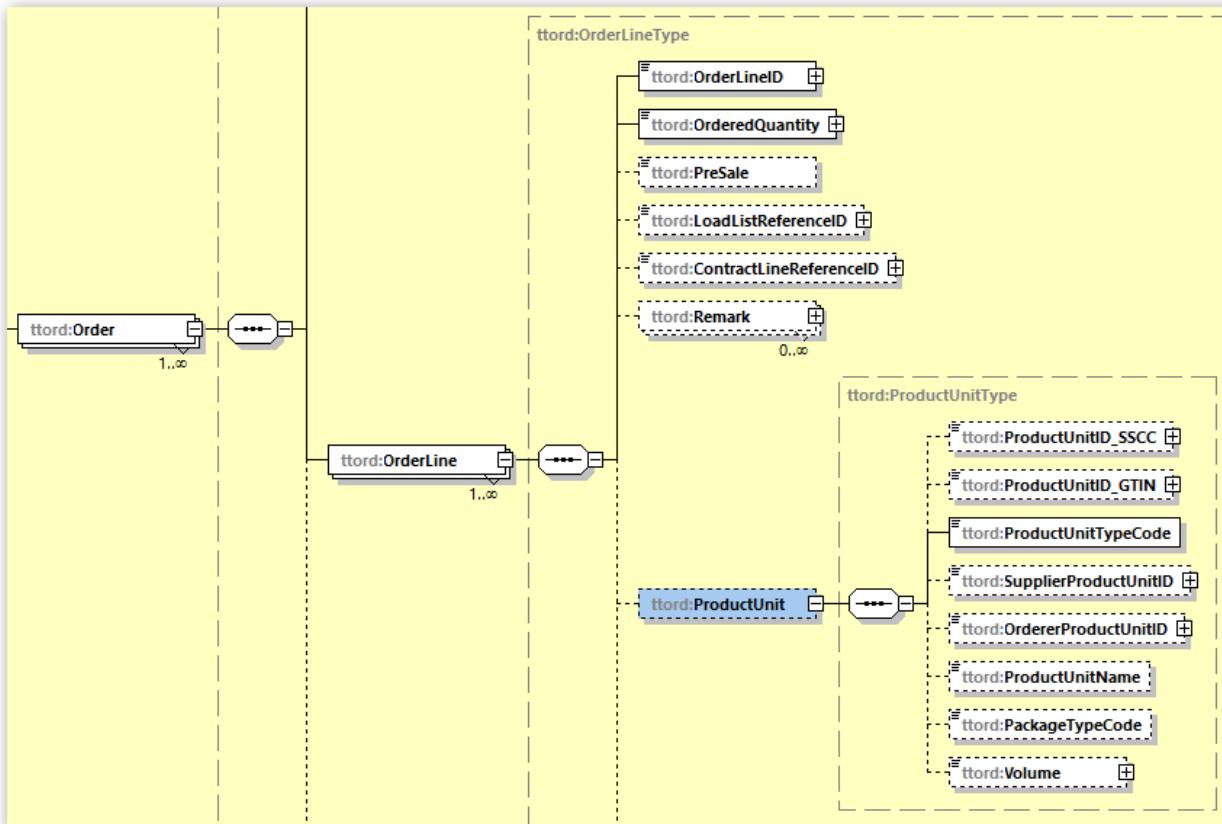
The Order and DespatchAdvice schema's are compliant to the Enterprise Architect class models for the Order and DespatchAdvice.

### 6.1. Instruction Order message

The Order message goes from wholesaler to manufacturer or from distributor to wholesaler.

Most essential in the Order message is the specification of the ordered products:

- Each Order has one to multiple OrderLine.
- Each OrderLine refers to one ProductUnit.
- Each ProductUnit is specified by a GTIN.
- OrderedQuantity specifies the quantity of the ordered ProductUnits.



Illustrated in XML:

```

<ttord:OrderLine>
  <ttord:OrderLineID schemeID="REF">OrderLineNr123456</ttord:OrderLineID>
  <ttord:OrderedQuantity unitCode="NBR">30</ttord:OrderedQuantity><!--betreft 30 dozen-->
  <ttord:ProductUnit>
    <ttord:ProductUnitID_GTIN schemeID="GTIN">23457890</ttord:ProductUnitID_GTIN>
    <ttord:ProductUnitTypeCode>103</ttord:ProductUnitTypeCode><!--betreft ConsumerUnit-->
    <ttord:ProductName>Doos met 6 flessen middel X</ttord:ProductName>
  </ttord:ProductUnit>
  <ttord:Timing>
    <ttord:EventTypeCode>102</ttord:EventTypeCode><!-- Deliver-->
    <ttord:EventDateTime>2018-02-15T09:30:47Z</ttord:EventDateTime>
  </ttord:Timing>
</ttord:OrderLine>
  
```

To specify the OrderedQuantity it is mandatory to use the attribute unitCode. unitCode must be filled with a value out of code list CL020. In the illustration NBR is used to indicate that a number of 30 units of the product type with GTIN 23457890 is ordered.

## 6.2. Instruction Transport Order message

The Transport Order message is sent by the manufacturer to the logistics provider or by the wholesaler to logistics provider.

The transport order is similar to a basic order message, with the difference that the transport order contains more detailed information about:

- the required means of transportation
- the load location and date/time of loading
- the unload location and date-time of unloading

```

<ttord:Timing>
  <ttord:EventTypeCode>101</ttord:EventTypeCode><!-- = Load-->
  <ttord:EventDateTime>2018-02-15T09:30:47Z</ttord:EventDateTime>
</ttord:Timing>
<ttord:Timing>
  <ttord:EventTypeCode>102</ttord:EventTypeCode><!-- = Deliver-->
  <ttord:EventDateTime>2018-02-17T09:30:47Z</ttord:EventDateTime>
</ttord:Timing>

```

### 6.3. Instruction DespatchAdvice message

The DespatchAdvice message has the same basic structure as an Order message, with the difference that the DespatchAdvice message contains more detailed information about the products and packaging of the to be delivered goods, such as:

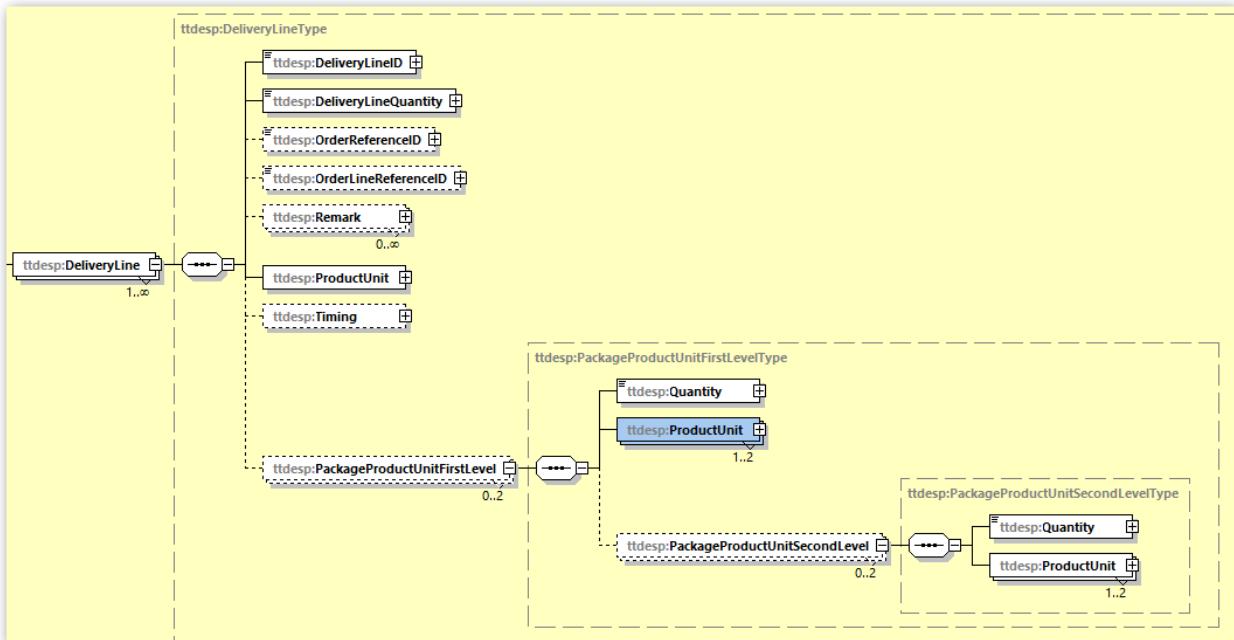
- delivery timing and instructions
- the SSCC's of pallets and boxes
- what items (GTINS) are stored on which pallet
- batch numbers and serial numbers of the individual items at the level of consumer units
- information about the weight and volume

```

<ttdesp:Delivery>
  <ttdesp:DeliveryID schemeID="REF">DEL234567</ttdesp:DeliveryID>
  <ttdesp:DeliveryStatusCode>101</ttdesp:DeliveryStatusCode><!--101= original-->
  <ttdesp:DeliveryDateTime>2018-10-10T09:30:47Z</ttdesp:DeliveryDateTime>
  <ttdesp:GrossWeight unitCode="KGM">2000</ttdesp:GrossWeight>
  <ttdesp:NetWeight unitCode="KGM">1800</ttdesp:NetWeight>
  <ttdesp:Timing>
    <ttdesp:EventTypeCode>102</ttdesp:EventTypeCode><!-- = Deliver-->
    <ttdesp:EventDateTime>2018-01-02T09:30:47Z</ttdesp:EventDateTime>
  </ttdesp:Timing>
  <ttdesp:Remark>
    <ttdesp:TextFunctionTypeCode>103</ttdesp:TextFunctionTypeCode><!-- = DeliverInstruction-->
    <ttdesp:TextLines>2e silo rechts</ttdesp:TextLines>
  </ttdesp:Remark>
  <ttdesp:MeansOfTransportation>
    <ttdesp:TransportMeansID schemeID="NUMMERPLAAT">82-ABC-89</ttdesp:TransportMeansID>
    <ttdesp:TransportMeansTypeCode>36</ttdesp:TransportMeansTypeCode><!-- = Truck, dry bulk-->
  </ttdesp:MeansOfTransportation>

```

The nesting of consumer units in boxes and boxes on pallets is specified using the construction with PackageProductUnit – First and Second Level:



In this example a LogisticUnit (a pallet) is delivered. The pallet is identified by a SSCC (4896512765432).

```

<ttdesp:DeliveryLine>
  <ttdesp:DeliveryLineID schemeID="REF">Nr123456</ttdesp:DeliveryLineID>
  <ttdesp:DeliveryLineQuantity unitCode="NBR">1</ttdesp:DeliveryLineQuantity>
  <ttdesp:OrderReferenceID schemeID="REF">Ordernr12345</ttdesp:OrderReferenceID>
  <ttdesp:OrderLineReferenceID schemeID="REF">Orderlinenr12345667</ttdesp:OrderLineReferenceID>
  <ttdesp:ProductUnit>
    <ttdesp:ProductUnitID_SSCC schemeID="SSCC">4896512765432</ttdesp:ProductUnitID_SSCC>
    <ttdesp:ProductUnitTypeCode>101</ttdesp:ProductUnitTypeCode><!-- = LogisticUnit-->
    <ttdesp:ProductUnitName>Pallet GBM voor Jansen</ttdesp:ProductUnitName>
    <ttdesp:PackageTypeCode>PA</ttdesp:PackageTypeCode><!-- = Pallet-->
    <ttdesp:ProductionDate>2001-12-17T09:30:47Z</ttdesp:ProductionDate>
    <ttdesp:ExpiryDate>2001-12-17T09:30:47Z</ttdesp:ExpiryDate>
  </ttdesp:ProductUnit>

```

The pallet contains two boxes. Each box on this pallet is specified as PackageProductUnitFirstLevel. In this example, each box has its own GTIN (40 12345 00003 07).

```

<ttdesp:PackageProductUnitFirstLevel>
  <ttdesp:ProductUnit>
    <ttdesp:ProductUnitID_GTIN schemeID="GTIN">40 12345 00003 7</ttdesp:ProductUnitID_GTIN>
    <ttdesp:ProductUnitTypeCode>102</ttdesp:ProductUnitTypeCode><!-- = TradeUnit-->
    <ttdesp:BatchNumber>34781289</ttdesp:BatchNumber>
    <ttdesp:ProductUnitName>Doos met 8 flessen Middel X</ttdesp:ProductUnitName>
    <ttdesp:PackageTypeCode>BX</ttdesp:PackageTypeCode><!-- = Box-->
    <ttdesp:ProductionDate>2017-12-17T09:30:47Z</ttdesp:ProductionDate>
    <ttdesp:ExpiryDate>2020-12-17T09:30:47Z</ttdesp:ExpiryDate>
  </ttdesp:ProductUnit>
  <ttdesp:PackageProductUnitSecondLevel>
    <ttdesp:ProductUnit>
      <ttdesp:ProductUnitID_GTIN schemeID="GTIN">40 12345 12345 7</ttdesp:ProductUnitID_GTIN>
      <ttdesp:ProductUnitTypeCode>103</ttdesp:ProductUnitTypeCode><!-- = ConsumerUnit-->
      <ttdesp:BatchNumber>34781289</ttdesp:BatchNumber>
      <ttdesp:ProductUnitName>Fles Middel X</ttdesp:ProductUnitName>
      <ttdesp:PackageTypeCode>BO</ttdesp:PackageTypeCode><!-- = Bottle-->
      <ttdesp:SerialNumber>PROD345678</ttdesp:SerialNumber>
      <ttdesp:ProductionDate>2017-12-31T09:30:47Z</ttdesp:ProductionDate>
      <ttdesp:ExpiryDate>2020-12-17T09:30:47Z</ttdesp:ExpiryDate>
    </ttdesp:ProductUnit>
  </ttdesp:PackageProductUnitSecondLevel>

```

To specify what items are in each box, PackageProductUnitSecondLevel is used. In this example each bottle in the trade unit (the box) has a GTIN, a batch number and a serial number.

## schemedID

In the xsd all identifying data-elements, like OrderID, GlobalISD, OrderLineID, are of the datatype IDType:

```
<xsd:complexType name="OrderLineType">
  <xsd:sequence>
    <xsd:element name="OrderLineID" type="udt:IDType" minOccurs="1" maxOccurs="1"/>
    <xsd:element name="OrderedQuantity" type="udt:MeasureType" minOccurs="1" maxOccurs="1"/>
    <xsd:element name="PreSale" type="udt:TextType" minOccurs="0" maxOccurs="1"/>
```

IDType has a special attribute attached, called schemaID, to indicate what type of schema is used to fill in the value of the identifier.

```
<xsd:simpleType name="schemeIDCode">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="REF"/>
    <xsd:enumeration value="GLN"/>
    <xsd:enumeration value="GTIN"/>
    <xsd:enumeration value="SSCC"/>
    <xsd:enumeration value="SERIAL"/>
    <!--REF= random reference-->
    <!--GLN= Global Location Number-->
    <!--GTIN= Global Trade Item Number-->
    <!--SSCC= Serial Shipping Container Code-->
    <!--SERIAL= Serial Number-->
  </xsd:restriction>
```

In the xml-message this is represented as follows:

```
<ttord:Order>
  <ttord:OrderID schemeID="REF">ORDER_NR_234567</ttord:OrderID>
  <ttord:OrderStatusCode>101</ttord:OrderStatusCode><!--101= original-->
  <ttord:OrderReferenceID schemeID="REF">ORDER_AGRIFIRM_346789</ttord:OrderReferenceID>
  <ttord:OrderDateTime>2018-01-23T09:30:47Z</ttord:OrderDateTime>
  <ttord:DeliveryTypeCode>101</ttord:DeliveryTypeCode><!-- Normal-->
  <ttord:TradeParty>
    <ttord:GlobalID schemeID="GLN">GLN_HollandFyto</ttord:GlobalID>
    <ttord:TradePartyRoleCode>OB</ttord:TradePartyRoleCode><!-- OrderedBy-->
    <ttord:TradePartyName>HollandFyto</ttord:TradePartyName>
    <ttord:StreetNameAndNumber>Ecu 2</ttord:StreetNameAndNumber>
```

## 6.4. Instruction Acknowledgement message

The Acknowledgement message has its own schema definition  
(TandT\_CPP\_Acknowledgement\_v2018p01.xsd).

Acknowledgement messages concern the technical processing of the received order and are not related to the content of the order.

The Order Acknowledgement message is a relatively simple message:

- concerning the message header:
  - o the Acknowledgement message has its own unique Message ID;
  - o SendingDateTime is the timestamp of sending the Acknowledgement message
  - o SendingPartyID and ReceivingPartyID differ from the original Order message;
  - o MessageType for the Acknowledgement message is '104'.
- concerning the referenced order:
  - o the OrderID refers to the original Sales- or TransportOrder.
  - o for OrderStatusCode only the values 103 (= accepted) or 104 (= not accepted) are allowed.
  - o optional: remarks concerning the acknowledgement can be added in the message using the Remark node as part of the Order node.
  - o for a Remark that concerns the acknowledgement code, 107 is used for the TextFunctionTypeCode. The remark itself is placed as free text in the TextLine field.
  - o a Remark can be used to indicate why an order was rejected. Examples: XML is not well-formed, XML is not valid, GTIN SendingPartyID not known, GTIN ReceivingPartyID not known, GTIN ordered product is not known, MessageType not known.

Illustration in xml:

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Example TandT_CPP_TransportOrder_Acknowledgement_12nov19, AgroConnect, Conny Graumans-->
<ttord:MessageHeader xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:ttord="http://www.agroconnect.nl/Portals/10/XSDs/TandT_CPP/v2018p01/TandT_CPP_Acknowledgement_v2018p01" xsi:schemaLocation="http://www.agroconnect.nl/Portals/10/XSDs/TandT_CPP/v2018p01/TandT_CPP_Acknowledgement_v2018p01 http://www.agroconnect.nl/Portals/10/XSDs/TandT_CPP/v2018p01/TandT_CPP_Acknowledgement_v2018p01.xsd">
    <ttord:MessageID schemeID="REF">ACK3809371</ttord:MessageID>
    <ttord:SendingDateTime>2019-11-12T15:25:47Z</ttord:SendingDateTime>
    <ttord:SendingPartyID schemeID="GLN">8719333007229</ttord:SendingPartyID><!--GLN Farmusol-->
    <ttord:ReceivingPartyID schemeID="GLN">8718077000008</ttord:ReceivingPartyID><!--GLN ADAMA-->
    <ttord:MessageType>104</ttord:MessageType><!--104= Acknowledgement-->
    <ttord:Order>
        <ttord:OrderID schemeID="REF">3809371</ttord:OrderID><!--reference to the TransportOrder message that was received-->
        <ttord:OrderStatusCode>104</ttord:OrderStatusCode><!--104= mesage was not accepted-->
        <ttord:OrderDateTime>2019-11-12T13:00:47Z</ttord:OrderDateTime><!--DateTime of the orriginal TransportOrder-->
        <ttord:TradeParty>
            <ttord:GlobalID schemeID="GLN">8719333007229</ttord:GlobalID><!--Farmusol-->
            <ttord:TradePartyRoleCode>WPK</ttord:TradePartyRoleCode><!--= WarehouseKeeper that accepted or declined the TransportOrder-->
            <ttord:TradePartyName>Farmusol B.V.</ttord:TradePartyName>
        </ttord:TradeParty>
        <ttord:Remark>
            <ttord:TextFunctionTypeCode>107</ttord:TextFunctionTypeCode><!--107= AcknowledgementRemark-->
            <ttord:TextLines>String</ttord:TextLines>
        </ttord:Remark>
    </ttord:Order>
</ttord:MessageHeader>
```

TextFunctionTypeCode is used to indicate the context of the Remark:

```
<xsd:simpleType name="textFunctionTypeCode">
    <xsd:restriction base="xsd:string">
        <xsd:enumeration value="101"/>
        <xsd:enumeration value="102"/>
        <xsd:enumeration value="103"/>
        <xsd:enumeration value="104"/>
        <xsd:enumeration value="105"/>
        <xsd:enumeration value="106"/>
        <xsd:enumeration value="107"/>
        <xsd:enumeration value="108"/>
        <xsd:enumeration value="999"/>
        <!--101= ShipToCommunication-->
        <!--102= WarehouseInstruction-->
        <!--103= DeliverInstruction-->
        <!--104= DespatchAdviceHeader/CMR-->
        <!--105= InvoiceInstruction-->
        <!--106= PaymentTerm-->
        <!--107= AcknowledgementRemark-->
        <!--108= ConfirmationRemark-->
        <!--999= Free text-->
    </xsd:restriction>
</xsd:simpleType>
```

Text lines concerning the Acknowledgement can be added as follows:

```
<ttord:Remark>
    <ttord:TextFunctionTypeCode>107</ttord:TextFunctionTypeCode><!--AcknowledgementRemark-->
    <ttord:TextLines>GTIN ReceivingPartyID not known</ttord:TextLines>
</ttord:Remark>
<ttord:Remark>
    <ttord:TextFunctionTypeCode>107</ttord:TextFunctionTypeCode><!--AcknowledgementRemark-->
    <ttord:TextLines>GTIN ordered product is not known</ttord:TextLines>
</ttord:Remark>
<ttord:TextLines>
```

## 6.5. Instruction Confirmation message

The Conformation message has the same structure as the Order message and is based on the same schema definition ([TandT\\_CPP\\_Order\\_v2018p01.xsd](#)).

The Confirmation is used to report back to the party that send the order, about how the Order was processed in the back office.

If the original Order can be fully executed as requested by the ordering party, than the Conformation message contains an exact copy of the original order.

In case the original order cannot be exactly executed, the Conformation message contains the modified (sales) order. The order can differ from the original purchase order because:

- products might be out of stock

- it cannot be delivered on the requested date/time
- etc.

In the header of the Confirmation message, MessageType is used to indicate that it concerns a Confirmation.

```

<ttord:MessageID schemeID="REF">MES89301</ttord:MessageID>
<ttord:SendingDateTime>2018-01-10T09:30:47Z</ttord:SendingDateTime>
<ttord:SendingPartyID schemeID="GLN">871807700008</ttord:SendingPartyID><!--GLN ADAMA-->
<ttord:ReceivingPartyID schemeID="GLN">8719324290005</ttord:ReceivingPartyID><!--GLN HollandFyto-->
<ttord:MessageType>105</ttord:MessageType><!-- Confirmation-->
<ttord:Order>
    <ttord:OrderID schemeID="REF">1808219</ttord:OrderID>
    <ttord:OrderStatusCode>105</ttord:OrderStatusCode><!--105= change-->
    <ttord:OrderDateTime>2018-02-26T09:30:47Z</ttord:OrderDateTime>
    <ttord:DeliveryTypeCode>101</ttord:DeliveryTypeCode><!-- Normal-->
    <ttord:DeliveryReferenceID schemeID="REF">3808392</ttord:DeliveryReferenceID>
    <ttord:CurrencyID>EUR</ttord:CurrencyID>
    <ttord:TradeParty>
        <ttord:GlobalID schemeID="GLN">8719324290005</ttord:GlobalID>
        <ttord:TradePartyRoleCode>OB</ttord:TradePartyRoleCode><!-- OrderedBy-->
        <ttord:TradePartyName>HollandFyto</ttord:TradePartyName>
    </ttord:TradeParty>
    <ttord:TradeParty>
        <ttord:GlobalID schemeID="GLN">8719324290227</ttord:GlobalID>
        <ttord:TradePartyRoleCode>IV</ttord:TradePartyRoleCode><!-- ToBeInvoiced-->
        <ttord:TradePartyName>HollandFyto</ttord:TradePartyName>
    </ttord:TradeParty>

```

Mandatory element in the Confirmation message are:

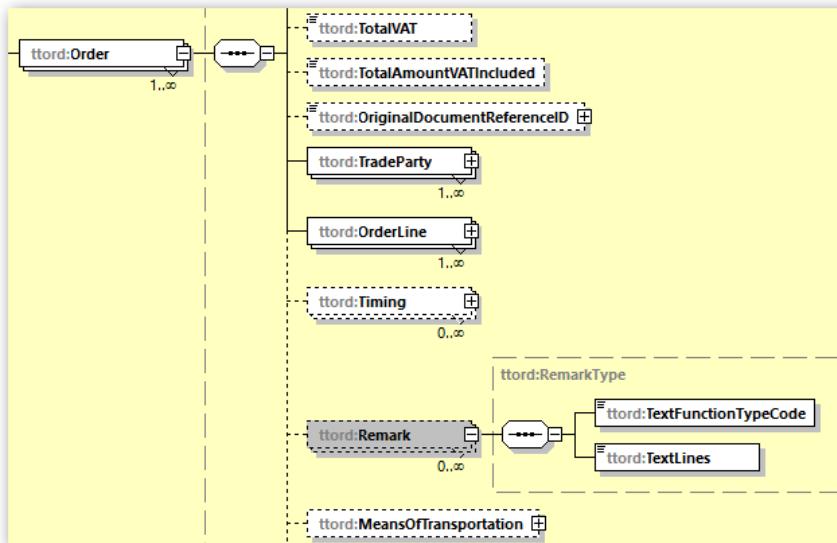
- MessageID: contains the confirmation number.
- OrderID: refers to the OrderID (order number) of the original order that was received.
- TradeParties: for buyer and supplier
- OrderStatusCode: e.g. 105 for 'change':

```

<xsd:simpleType name="documentStatusCode">
    <xsd:restriction base="xsd:string">
        <xsd:enumeration value="101"/>
        <xsd:enumeration value="102"/>
        <xsd:enumeration value="103"/>
        <xsd:enumeration value="104"/>
        <xsd:enumeration value="105"/>
        <!--101= original-->
        <!--102= revised version-->
        <!--103= accepted-->
        <!--104= not accepted-->
        <!--105= change-->
    </xsd:restriction>
</xsd:simpleType>

```

Remarks concerning the confirmation can be added in the message using the Remark node as part of the Order node:



TextFunctionTypeCode can be used to indicate the context of the Remark:

```

<xsd:simpleType name="textFunctionTypeCode">
    <xsd:restriction base="xsd:string">
        <xsd:enumeration value="101"/>
        <xsd:enumeration value="102"/>
        <xsd:enumeration value="103"/>
        <xsd:enumeration value="104"/>
        <xsd:enumeration value="105"/>
        <xsd:enumeration value="106"/>
        <xsd:enumeration value="107"/>
        <xsd:enumeration value="108"/>
        <xsd:enumeration value="999"/>
        <!--101= ShipToCommunication-->
        <!--102= WarehouseInstruction-->
        <!--103= DeliverInstruction-->
        <!--104= DespatchAdviceHeader/CMR-->
        <!--105= InvoiceInstruction-->
        <!--106= PaymentTerm-->
        <!--107= AcknowledgementRemark-->
        <!--108= ConfirmationRemark-->
        <!--999= Free text-->
    </xsd:restriction>
</xsd:simpleType>

```

For a Remark that concerns a confirmation, code 108 is used. The remark itself is places as free text in the TextLines field.

## Appendix A: Explanation to Bayer experts, September 2018

Explanation:

- As the first AgroCloser pilot implementation, the data exchange between distributor HollandFyto and producer Adama is now being implemented.
- The information flows for this pilot are shown in "Sequence diagram number 7". First the PurchaseOrder from Holland Fyto is implemented to Adama, then the TransportOrder from Adama to logistical service provider Imperial / Riksen, then the DespatchAdvice from Imperial / Riksen via Adama to Holland Fyto.
- For the first Bayer implementation it may be that one of the other sequence diagrams should serve as a basis (to be determined in more detail, to be tightened up).
  - Two standard xml messages are therefore defined for the standard data exchange: the Order message and the DespatchAdvice message. The message specifications are included as xsds (xml-schema definitions) in the corresponding zip-file (180603.Xsd's and examples T & T CPP.zip). The xsds can be opened with a special browser (for example XML-Spy) and show the hierarchical structure of the xml message with the tags of the data elements. The final xml messages can be validated against these xsds.

When opening the Order or DespatchAdvice xsd, the xsd's with data types (TandT\_DataTypes) and with unqualified data types (UnqualifiedDataTypes) are automatically read.

Furthermore, in the xsd-zip some example xml-messages are attached for the use case Adama.

Furthermore:

- The xsd's are derived from two class diagrams (that's what it actually starts with) that shows all definitions of the classes and the data elements used (see "180215.Class diagram T & T CPP, v2018p08.pdf"). Class diagrams and sequence diagrams have been elaborated with the Enterprise Architect (EA) analysis tool; the EA file is included.
- The class diagram report also shows which standard code lists to use.
- The code lists to be used are collected in the separate zip-file "180220.Code lists T & T\_CPP.zip".
- The choices made during the practical implementation of the xml messages are collected in "180702.Implementation instruction T & T CPP.pdf"; this is a living document that is constantly being supplemented.
- The standard messages are exchanged via the Proagrifica mailbox system.
- Every party involved in the supply chain is identified through a GS1-GLN. Products are identified with a GS1-GTIN and packaging / pallets with a GS1-SSCC.

Conny Graumans suggests that Bayer first delves into the material and that we then do a special telco session in which he can explain the specs in more detail and in which we can explore the approach to the Bayer use case.

The aim is for the parties involved to make maximum use of the standard AgroCloser Order and DespatchAdvice messages for the data exchange. But if Bayer already supports these data streams in a different message format, then it can be decided to convert the own format to the AgroCloser format on the Proagrifica platform; that would mean that Bayer would have to make minimal adjustments to the existing interface to its back office ERP.

## Appendix B: Overview code list to be used in standard messages

The following code lists are to be used in de standard T&T CPP messages.

code list	to be used in data elements	source(s)	comments
countryCode	CountryCode	► ISO 3166	ISO code of the country of the physical address.
currencyCode	CurrencyID	► ISO_ISO3AlphaCurrencyCode_2012-08-31.xsd	Code for the currency that is used to express a financial amount.
documentStatusCode	InvoiceStatusCode	► AgroConnect: CL602	Indication of the status of the exchanged document. 'Original' and 'Revised version' are used in the Order and DespatchAdvice message. 'Accepted', 'Not accepted' and 'Denied' are used in the Acknowledgement and Confirmation message.
eventTypeCode	EventTypeCode	► AgroConnect: CL606	Indication of the type of event for this specific timestamp or for this specific timeslot.
measurementUnitCode	GrossWeight NetWeight	► UNECE_MeasurementUnitCommonCodeVolume_4.xsd ► UNECE_MeasurementUnitCommonCodeWeight_4.xsd rec20-08052015 ► UnitsOfMeasurementRev9e_2014.xls	Must use the UNECE / ISO standard codes for types of measurements. Code NBR (Number) is added to the standard list (could be replaced by C62 = 1 unit).
messageTypeCode	MessageType	► AgroConnect: CL601	Specification of the type of message; can be an Order or a DespatchAdvice message.
messagePurposeCode	MessagePurposeCode	► UNECE_MessageFunctionCode_D16B.xsd	Indication of the purpose of this message. Codes are picked from
packageTypeCode	PackageTypeCode	► UNECE_PackageTypeCode_2006.xlsx ► <a href="http://www.unece.org/uncefact/codelist/standard/UNECE_PackageTypeCode_2006.xsd">http://www.unece.org/uncefact/codelist/standard/UNECE_PackageTypeCode_2006.xsd</a>	Must use the UN/Cefact coding.
productUnitTypeCode	ProductUnitTypeCode	► AgroConnect: CL605	Indicates whether the ProductUnits is a LogisticUnit, a TradeUnit or a ConsumerUnit.
schemeIDType	SchemalID	► AgroConnect: CL604	Identifying data-elements, like OrderID, GlobalISD, OrderLineID, are of the datatype, called IDType. IDType has a special attribute attached, called schemaID, to indicate what type of schema is used to fill in the value of the identifier.
textFunctionTypeCode	TextFunctionTypeCode	► AgroConnect: CL603	Indicates the type of event that is commented such as a warehouse instruction or a delivery instruction.
tradePartyRoleCode	TradePartyRoleCode	► UNECE 3035: UNECE_PartyRoleCode_D16B.xsd ► gs1:gdd:cl:PartyRoleCode	Indicates the role of the party in this trade event. For instance the seller, buyer, transporter or the party to be invoiced.
transportMeansTypeCode	TransportMeansTypeCode	► UNECE_TransportMeansTypeCode_2007.xsd	Specification of the type of carrier that is used (e.g. a lorry, a vessel, etc.).