

287933 v1.0

Aptean Ltd
Copyright © 2011-2026.

Contents

1 287933.....	1
2 Functional Overview.....	3
2.1 Client Requirement.....	3
2.2 Solution.....	3
2.3 Scope.....	5
3 Set-up.....	6
3.1 Pre-requisites.....	6
3.2 Menu Structure.....	6
3.3 Data.....	6
3.4 Implementation Advice.....	6
4 Functional Description.....	7
4.1 Outbound Interface (C-TMS- Zetes).....	7
4.2 Unadvised Receipts.....	7
5 AUTHORISED BY.....	9

1 287933

Aaptean

DHL C-TMS

Changes to Zetes Interfaces (outbound and inbound)

FUNCTIONAL SPECIFICATION - 10.6

12/807/2011 - 1.0

Reference: FS 287933 OB-8G3KSC



2 Ready for **What's Next, Now™**



2 Functional Overview

2.1 Client Requirement

Changes are required to the existing Zetes flows to bring in line with all the changes made to the outbound scanning of items through the cross dock.

Outbound Interface (C-TMS - Zetes)

The current interface will send all items with a 'To Deliver' qty of greater than zero; however for items without one of the 4 outbound scanning codes, the interface will set the 'To Deliver' qty to be zero. If we then subsequently load an item that we have already interfaced to Zetes (item left off for delivery on day one as no space on the vehicle, but then sent on day 2), the item will move onto a new OMS_REF as created by the outbound functionality, but will be rejected by the Zetes interface as we will have already sent it to them.

A change therefore needs to be made to the interface, so that it will only send items that have one of the four 'Scan Out' codes (these are SL, UL, OL and XL). This will not only ensure that the qty on the interface is inline with the outbound scanning, but will also make the qty interfaced to Zetes match the qty that is on the drivers printed trip sheet. This should then also solve the problem of us sending them the same item number more than once, as if the item isn't on the vehicle, it will not be on the xml.

Inbound Interface (Zetes - C-TMS)

Some investigation work and changes also need to be made to the inbound interface whilst completing this RIO.

The inbound interface back to CTMS needs to work at item level, rather than at OMS level. Currently, if we fail to scan an item out of the cross dock, but do load it, when the store receive the item as 'Unplanned' and interface the item back to us, the original OMS_REF is what the POD flag and 'Delivered' qty goes against (I guess this is because they are using the original label to scan the item in against, which will have the OMS_REF of the order created by the SAP interface). The inbound interface should only put a POD flag and 'Delivered' qty against the OMS_REF of the item which doesn't have a 'X' in front of it (this X is created when items are moved from their original OMS_REF's onto new ones as part of the inbound and outbound scanning functionality). The interface (if not currently) should put the 'Short Delivered' code against items at item level too. I believe that currently if 5 items are sent to a store but only 4 are scanned in, the interface will short the entire OMS_REF, rather than just the single item that isn't scanned in (this may not be the case, however if it is then this needs changing). The interface also needs to be able to receive and update a 'Delivered' qty and POD flag against an OMS_REF that is at a delivered status (this may already be the case, but if it isn't then it needs developing). It is also important that users can manually set the 'Delivered' qty and put a POD flag against orders which are accidentally not scanned in at stores (however I believe this functionality is already possible).

This inbound interface is linked to RIO OB - 8EZLAR, as the date and time that the interface changes the 'Delivered' qty against each item will now be displayed on the Supplier Portal to the customer and to suppliers.

2.2 Solution

2.2.1 Outbound Interface (CTMS - Zetes)

The procedure that creates the C-TMS manifest message to Zetes has to be entirely in-line with the driver documentation printed once the delivery vehicles are loaded and issued to the driver.

The current message logic does consider the load scan reason codes (SL, UL, OL and XL) and if any one of the items have been loaded for an order with reference to these reason codes, all of the items not prefixed by X (those that have been split away to another order for receipt and or load) are included in the XML output as order details of the message.

The logic will be modified to remove the count of items with loaded reason codes, which currently if greater than 1, sends all items other those prefixed by X.

The new logic will consider each individual item and include the item for the order if:



1. The item identifier is not prefixed by X
2. One of the 4 positive loading reason codes has been captured for the item
3. The DELIVER_TO qty of the item is set to 1

The emphasis of the change is to consider the DELIVER_TO qty being not 0.

To illustrate the points above, if an order is planned to load with 3 pallets and only 2 are loaded according to the reason codes and DELIVER_TO, the third item should not feature in the output. Once the third pallet is loaded on another vehicle, the item will then be included for the vehicle delivery under a split order.

This method means that the unique item identifier for each item should only ever be transmitted to Zetes for one vehicle delivery manifest.

Note also that from investigation of the current files, examples have been identified where an order header is included in the outbound file according to the plan load but no order details because none of the items have been scanned to the vehicle. This might be as a consequence of a number of operational issues but usually where the product has not been physically received into the X-Dock hub at Leicester.

The outbound message should only ever include the order header information where at least one order item has been loaded to the delivery vehicle according to the 3 conditions described above.

2.2.2 Inbound Interface (Zetes - C-TMS)

Unadvised Receipts - Where an item is scanned into store that has not been pre-advised from C-TMS, this being the scenario where an item has been physically loaded but not scanned at the Leicester X-Dock, Zetes has a function to accept unadvised receipts into store.

Note that the label of each item contains the destination store, so it is assumed that unadvised receipts can only be processed for items delivered to the correct store. Also, the label of each item contains the unique item identifier (delimiter tag 21) and the original order OMS_REF (delimiter tag 240).

The interface upload will be modified where necessary to use the item identifier to lookup the appropriate order to update the DELIVERED qty. (*The original order derived from the label, and assumed to be present in the Zetes message to C-TMS, might have been superseded in C-TMS after the label was printed due to receipt scanning and order split functionality.*) It is assumed that an appropriate reason code will be provided by Zetes, this value to be determined by DHL and advised to ODSL.

Where C-TMS determines that an unadvised receipt has been processed in Zetes from the response message, it is assumed the item might or might not be assigned to the specific delivery trip and stop Zetes processed. Where not, the order split functionality will be used to generate a new OMS_REF for the item and this order will be allocated to the delivery trip - this means processing the loading logic retrospectively to maintain the data integrity in C-TMS.

Updating DELIVERED qty - The update message from Zetes from in-store scanning back to C-TMS is in the same format as the message sent to Zetes to pre-advise the items expected. The important principle governing the DELIVERED qty update for the items received (or confirmed short) is to update the order item of the unique ITEM_IDENTIFIER regardless of the OMS_REF. This update will be for the unique ITEM_IDENTIFIER and not any historic audit ITEM_IDENTIFIER prefixed by X as a result of order splitting.

Updating Reason Codes - Reason codes will be (and are currently) processed for each unique item, not for the whole order.

Order Status - The update from Zetes needs to consider orders might be at DELIVERED status; the delivered qty at the unique item identifier in C-TMS should still be updated to 1 if advised as received in the Zetes message.

Positive reason code on Store receipt - It is recommended that a new positive receipt reason code is introduced from the Zetes message update. For each item received a reason code SD will be created in addition to updating the DELIVERED qty to 1.

Updating Constraints - Currently, there is validation that the trip actual times for the delivery stops are captured before updates to order lines are allowed. This validation will be removed because the actual times are received from Microlise and the order updates from Zetes and the timings and delivery of the respective flow of messages to C-TMS cannot be



guaranteed in that sequence.

POD Flag - The POD flag will be set on each order (managed at order level not order line or item line level) in C-TMS as soon as the first item is confirmed as successfully delivered from the Zetes message. Note that it is possible to set the DELIVERED qty and the POD flag on order items and orders manually through C-TMS debrief and order forms.

2.3 Scope

This change will be applied to C-TMS system version 10.6.0.



3 Set-up

3.1 Pre-requisites

None

3.2 Menu Structure

Not required

3.3 Data

None required

3.4 Implementation Advice



4 Functional Description

4.1 Outbound Interface (C-TMS- Zetes)

Currently, the procedure which builds the outbound item data to Zetes INT_XML_OUT2.PROCESS_XML_OUTBOUND_ZETES will send out all items on an order if one of the items has been successfully loaded onto the trip.

The functionality will be amended to only send items which have been successfully loaded based on the reason codes associated with the items.

The codes identified as successful loads are SL,UL,OL, XL.

Currently the reason codes are established as part of the item display, at this point it is too late to start to suppress items.

Before an order header is written, new functionality will establish if there are any successfully loaded items for the order.

Only where items have been successfully loaded should the order header be displayed in the outbound file.

Only items which have a successful load reason code should be displayed in the detail. The existing code will be amended so that only successfully loaded items are considered for display.

If an item does not have a successful load reason code, it will not be sent to Zetes. If an order does not contain any items which have been successfully loaded on the trip, the order header will not be sent to Zetes.

Inbound Interface (Zetes - C-TMS)

4.2 Unadvised Receipts

Currently when receiving the inbound Zetes file, the item_identifier is sent in with an oms_ref.

For unadvised receipts, the oms_ref is always the original oms_ref the item was planned on.

As part of the processing, new functionality will find the current oms_ref for the item. This is the order where the item is listed without an X.

Once the relevant order for the item is identified, a second process will check if the order was planned on the trip.

Where the order is not planned on the current trip, the item will be split from the original order and applied to a new order.

As part of the process, the collection of the new order will be planned onto the collection trip of the original order and the delivery leg of the new order will be planned on the current trip.

Delivered Qty

When updating the delivered quantity, the oms_ref is used to identify which record is updated. This is currently the oms_ref sent from Zetes, changes will be made to use the current oms_ref for the item.

Order Status

Regardless of the order status, the delivered qty will still be updated.

Updating Constraints and PODs.



Currently to update the POD for an order the system calls TRM.SET_ACTUALS, passing the oms_ref provided by Zetes.

There is validation within the procedure that will not allow the process to set the POD on an order until the delivery stop actual data items have been populated. This can cause issues as the actual data items are populated from MICROLISE.

The existing call to TRM.SET_ACTUALS will be replaced with an update statement on SCH_ORD which sets the POD to ?Y? for the current order of the item and not the order received from Zetes.

Table Updates Required

No table changes are required for this development

References

Ref No	Document Title & ID	Version	Date
1	EST- 287933 OB-8G3KSC Changes to Zetes Interfaces (outbound & inbound) v2.0	2.0	23/05/2011

Glossary

Term or Acronym	Meaning
C-TMS	Calidus TMS

Document History

Version	Date	Status	Reason	Initials
0.1	05/07/2011	Draft	Initial version	SEW
0.1	12/07/2011	Draft	Review of the initial draft	PJH
1.0	12/07/2011	Issue	Issued to client	PJH



5 AUTHORISED BY

<i>Matt Crisford</i>	Development Manager
<i>Peter Greer</i>	TMSCC MTS Product Manager

