

**292349 v2.0**

Aptean Ltd  
Copyright © 2011-2026.

# Contents

<b>1 292349.....</b>	<b>1</b>
1.1 Client Requirement.....	2
1.2 Solution.....	2
1.3 Scope.....	3
<b>2 Set-up.....</b>	<b>4</b>
2.1 Pre-requisites.....	4
2.2 Menu Structure.....	4
2.3 Data.....	4
2.4 Implementation Advice.....	4
<b>3 Functional Description.....</b>	<b>5</b>
3.1 EDI Maintenance.....	5
3.2 PLUTO to C-TMS Processing.....	6
3.3 Empty Stop Maintenance.....	9
3.4 C-TMS to Microlise Processing.....	9
3.5 C-TMS to PLUTO Processing.....	10
3.6 Interface Errors.....	11
<b>4 AUTHORISED BY.....</b>	<b>12</b>

1 292349

**A**aptean

DHL C-TMS

## Pluto C-TMS Integration

### FUNCTIONAL SPECIFICATION - 10.7

4/11/2011 - 2.0

Reference: FS 292349 PM-8LZMAN



## 1.1 Client Requirement

PLUTO - C-TMS Integration.

Receive and upload XML files representing vehicle movement by driver shift planned in PLUTO. Send 1 trip in each file.

Handshake confirmation of data successfully uploaded into C-TMS back to PLUTO.

C-TMS creation of actual arrivals and actual departures message to PLUTO based on same Trip and Order schema.

Store PLUTO stop number in C-TMS.

## 1.2 Solution

A new EDI process will be created in the EDI Maintenance screen. This process will upload and process orders and trips sent in from the PLUTO system. The current ?TripOrder? XML format will be used to transmit the data from PLUTO to C-TMS and from C-TMS to PLUTO

A unique identifier will be passed in for each order as the customer reference. This identifier will be unique and will define the order number, product number, product split and leg number. For each trip in a schedule day, PLUTO will provide a route code which defines the vehicle movement in the plan for the shift, for example ?BEL DD3-1? means ?Bellshill Double Deck 3 for shift 1. PLUTO plans based on vehicle movements for notional vehicles available from the fleet each day.

It is expected that all required information to create both orders and trips will be provided in the message. The existing validation process will take place and any messages that do not contain the necessary information will be rejected. If the orders or trips are invalid they will be rolled back and a new file will need to be sent to create the orders and trips. Once the orders and trip have been uploaded into C-TMS and the trip set to ?PLANNED? status a ?TripOrder? message will be sent back to PLUTO to confirm file has completed. This file will use the current ?TripOrder? XML format and will contain the C-TMS trip ID created along with the C-TMS OMS references created. A file will only be sent back to PLUTO if all orders and trips have been created successfully. If a file is not sent back from C-TMS to PLUTO, this will indicate the trip or orders have not been successfully loaded into C-TMS.

When creating the orders in C-TMS, each new order will be given a unique OMS reference; the identifier passed into C-TMS from PLUTO will be stored as the ?External Reference? (?Customer Reference?). Each trip created will be given a unique trip ID starting with ?PLU-? and the status will be set to ?PLANNED?. Changes will be processed against trips that have status ?EN-ROUTE? but only for legs not yet executed.

When creating trip stops in C-TMS from the interface message, the unique stop ID from PLUTO for the schedule and trip will be saved and will be sent back to PLUTO for the message handshake confirmation and for the actual arrival and actual departure times update messages.

A new system parameter will be introduced to prevent the deletion of empty trip stops during processing.

The C-TMS interface errors forms will be available to allow the status of uploaded messages to be reviewed. This will include ?SUCCESS? and ?FAILURE?. Any failures can be found and reviewed and error messages will indicate the failure reason.

If a file is received that contains a trip and orders that already exist in C-TMS, this file will be treated as an update. The PLUTO route code, PLUTO stop number and order external reference will be used to uniquely identify the trips and orders that already exist on C-TMS. The existing trips and orders will be deleted and replaced with the new data. The trips and orders will be removed completely to prevent issues with unique references. Updates to trips and orders will only be permitted until the actual arrival or actual departure is entered against a stop. This will be from a Microlise update normally but manual intervention in debrief will be possible and will provide the same input. Once this has happened, the trip stops and orders on those stops will be locked from any changes via the PLUTO interface.

PLUTO on occasion will send DELETE messages if a trip is completely removed from the plan prior to execution.

When an actual arrival or actual departure date and time is entered against a trip stop, a message will be triggered to PLUTO. The message will be in the ?TripOrder? XML format, and will contain the information for the trip that has been updated. The same leg lock strategy will be maintained in PLUTO based on the actual arrival and actual departure time available from the depot.



A record will be written to the interface table ?INT\_XML\_CONTROL? for each trip and a database job will run at a set interval to process the outstanding messages. This outbound message will be controlled in the EDI maintenance screen.

## 1.3 Scope

This change will be applied to system version 10.7.0.



## 2 Set-up

### 2.1 Pre-requisites

None

### 2.2 Menu Structure

Unchanged

### 2.3 Data

Unchanged

### 2.4 Implementation Advice

The EDI flows will be setup as described in section 3.1.



## 3 Functional Description

### 3.1 EDI Maintenance

A new EDI process will be setup in the ?EDI Maintenance? screen to process the inbound XML file from PLUTO via DHL Link into C-TMS using the ?TripOrder? XML format.

For example:

The screenshot shows the 'EDI Maintenance' screen with the following configuration:

- Process Name:** PLUTO\_TRIP\_ORDERS\_XML\_IN
- Filename Format:** PLUTO\_TMS\_TRIP\*
- Customer:** (dropdown)
- Cost Centre Code:** (dropdown)
- Location:** (dropdown)
- Direction:** Inbound
- Flow Type:** ORD\_XML
- Frequency Type:** Regular Interval
- Interval Length:** 5 Minutes
- Params:** (button)
- Output:** (button)
- Send DEL message:** (checkbox)
- Send ARR Message:** (checkbox)
- Status:** Stopped
- Last Run Date:** (dropdown)
- Next Run Date:** (dropdown)
- Start:** (button)
- Stop:** (button)
- Delivery Folder:** /webint/ndtst/interface/TRIPXML/in
- Archive Folder:** /webint/ndtst/interface/TRIPXML/in/archive
- Failures Folder:** /webint/ndtst/interface/TRIPXML/in/failures
- Acknowledgement Folder:** (dropdown)
- Send ACK?** (checkbox)

Buttons at the bottom: New, Delete.

The ?Params? screen will be used to set a new trigger called ?Send Handshake Message? to a value of ?Y? for a type of ?P? to send the ?handshake? message to PLUTO upon successful upload of the inbound file:

The screenshot shows the 'Process Trigger Types' screen with the following table:

Trigger	Value	Type

Buttons at the bottom: New, Delete, Close, Save.

The ?FTP Details? screen will be used to store the FTP details for the ?handshake? message:



FTP Details

Destination IP: [Text Box]

FTP Port: [Text Box]

FTP Username: [Text Box]

FTP Password: [Text Box]

FTP Directory: [Text Box]

**Close**

A new EDI process will be setup in the ?EDI Maintenance? screen to process the outbound XML file from C-TMS to PLUTO via DHL Link using the ?TripOrder? XML format.

A ?stop departure? message will be required to be sent to PLUTO.

For example:

EDI Maintenance

Process Name: PLUTO\_TRIP\_ACTUALS\_XML\_OUT

Filename Format: TMS\_PLUTO\_TRIP\_ACTUALS

Customer: [Text Box]

Cost Centre Code: [Text Box]

Location: [Text Box]

Flow Type: TRIP\_XML

Direction: Outbound

Frequency Type: Regular Interval

Interval Length: 5 Minutes

Params

Output

Send DEL message:

Send ARR Message:

Status: Stopped

Last Run Date: [Text Box]

Next Run Date: [Text Box]

Start

Stop

Delivery Folder: /webint/indstst/interface/TRIPXML/out

Archive Folder: /webint/indstst/interface/TRIPXML/out/archive

Failures Folder: /webint/indstst/interface/TRIPXML/out/failures

Acknowledgement Folder: [Text Box]

Send ACK?

**New**

**Delete**

The ?Params? screen will be used to set a new parameter called ?PRINT\_ORD\_SECTION? to a value of ?N? for a type of ?P? to exclude the order sections from the ?TripOrder? XML file.

## 3.2 PLUTO to C-TMS Processing

### 3.2.1 Trip and Order Validation

A unique identifier will be passed in for each order as the customer reference. This identifier will be unique and will define the purchase order number, purchase order number line, purchase order line number split and leg number. For each trip, PLUTO will provide a route code which defines the vehicle movement in the plan for the shift, for example ?BEL DD3-1? means ?Bellshill Double Deck 3 for shift 1?. PLUTO plans based on vehicle movements for notional vehicles available from the fleet.



It is expected that all required information to create both orders and trips will be provided in the message. The existing validation process will be performed and any messages that do not contain the necessary information will be rejected. If the orders or trips are invalid they will be rolled back and a new file will need to be sent to create the orders and trips.

Specific validation on the trip and order file will include the following checks for the upload:

File Level:

- The ?<EVENT\_SOURCE\_TYPE>? tag is ?PLUT?.
- The ?<EVENT\_DATE>? tag is a valid date.
- The filename has not been processed already (i.e. the filename does not exist on the ?INT\_XML\_TRIP\_STOP? database table)

Trip Level:

- The ?<ROUTE\_CODE>? tag is populated.
- The trip status is ?PLANNED?, ?ACCEPTED? or ?EN-ROUTE? if the trip ID is provided for update/replacement.

Trip Stop Level:

- The ?<STOP\_SEQ>? tag is populated with the stop number.
- At least one leg exists that may be updated (i.e. the actual arrival or departure time has not been set).

Order Level:

- The ?<SO\_REF>? tag is populated (i.e. external reference).
- The order status is ?UNSCHEDULED?, ?SCHEDULED?, ?SCHED\_COLL? or ?SCHED\_DEL?.

The external reference will store the PLUTO order reference in the following format:

- Order (i.e. Purchase Order Number)
- Product Line (i.e. Purchase Order Line Number)
- Product Split (i.e. Purchase Order Line Number Split)
- Leg (i.e. Trip Leg Number)

For example: ?99999999.0001.01.01?

The ?INT\_XML\_IN? package will be changed so that the PLUTO trip and order upload process is identified using the ?<EVENT\_SOURCE\_TYPE>? tag (i.e. ?PLUT?) and a new procedure called ?P\_PROCESS\_RESCHEDULED\_TRIP\_ORDER? is run from the existing ?PROCESS\_ORD\_XML\_IN? procedure called by the EDI polling process, this new procedure will be performed after the successful validation of the information provided in the file at the file level. The ?PROCESS\_ORD\_XML\_IN? procedure will be changed to check the event source type and if it is ?PLUT? then it will pass control to a new procedure called ?P\_PROCESS\_RESCHEDULED\_TRIP\_ORDER? to extract the data from the file.

Modifications of trip stops will only be permitted until the actual arrival or actual departure date and time is entered against a loading trip stop of a leg. This will be normally from an update message from Microlise, but manual intervention during trip debrief will be possible. Once this has happened, the trip stops and orders on those stops will be locked from any changes via the PLUTO interface.

Microlise will send update messages for actions ?STA? (i.e. start-up) and ?DEP? (i.e. departure) and it will be these messages that will ?lock? the legs of the trip to prevent further modification for planning because they relate to the trip stops where loading activities will occur.

If the actual arrival or actual departure date and time has been entered against a trip stop then the leg of the trip will be ?locked? and the trip stop cannot be updated or replaced by the upload process.

N.B. A leg will be considered to be the loading and unloading activities at the location stops and it is expected that there will be distinct stops for each activity (i.e. there will be pairs of trip stops that constitute a leg). It is also expected that the stop number will be consecutive for these stops.

For example, if stop 1 (which contains a loading activity) has actual times set then stop 1 and stop 2 will be ?locked? because the leg of the trip is between stops 1 and 2.



However, if stop 2 (which contains an unloading activity) has actual times set then stop 3 will not be ?locked? regardless of the activity because it is part of a separate leg.

### 3.2.2 Trip and Order Processing

When creating the transport orders in C-TMS, each new order will be given a unique OMS reference; the identifier passed into C-TMS from PLUTO in the ?<SO\_REF>? tag will be stored as the ?External Reference? (i.e. ?Customer Reference?).

Each trip created will be given a unique trip ID starting with ?PLU-?, and the status will be set to ?PLANNED?. Changes will be processed against trips that have a status of ?PLANNED?, ?ACCEPTED? or ?EN-ROUTE?, but only for legs not yet executed when ?EN-ROUTE? (i.e. without actual information recorded against the trip stop).

If a file is received that contains a trip (i.e. for the route code) and orders that already exist in C-TMS, this file will be treated as a replacement: therefore, the original trip and its orders will be deleted and new ones created in their place based on the trip and order information provided in the file.

The PLUTO route code, PLUTO stop number and order external reference will be used to identify the trips and orders that already exist on C-TMS. The existing trips and orders will be deleted and replaced with the new data. The trips and orders will be removed completely to prevent issues with unique references and it will include the audit trail of the trips and orders.

N.B. It will be the route code that will determine whether the trip and its orders will be replaced.

Deletion:

If the whole trip will be replaced then the new ?TRM.F\_DELETE\_RESCHEDULED\_TRIP? function will be run for the trip ID found for the route code provided. This function will delete any transport orders on the trip stops, haulage activities and trip stops.

The status of the trip will be set first to ?DELETED? to ensure that Microlise is informed of the deletion of the trip via the ?TRG\_SCH\_TRIP\_XML\_INT? database trigger, then the trip itself and its audit records will be deleted.

The new trip ID created will be assigned the same details, for example, carrier, driver, crew and vehicle; it will then be updated to the same status as the deleted trip.

If part of the trip (i.e. some stops) will be replaced then the new ?TRM.F\_DELETE\_RESCHEDULED\_STOP? function will be run in reverse sequence for the stop numbers not on ?locked? legs for the trip ID found for the route code provided.

The trip ID, carrier, driver, crew, vehicle and its status will be retained.

N.B. It is expected the same trip stops will be provided by PLUTO for any ?locked? legs.

Because the PLUTO purchase orders will be specific to the legs of the trip, the transport orders will be deleted with the trip stops; this is because there cannot be a direct link between the external reference and the OMS reference if the PLUTO purchase order is moved to a different trip and/or stop.

New functions will be created in the ?TRM? package for each stage of the deletion process:

- F\_DELETE\_RESCHEDULED\_ORDER\_ITEMS
- F\_DELETE\_RESCHEDULED\_ORDER\_AUDIT
- F\_DELETE\_RESCHEDULED\_ORDER
- F\_DELETE\_RESCHEDULED\_ACTIVITY
- F\_DELETE\_RESCHEDULED\_STOP
- F\_DELETE\_RESCHEDULED\_TRIP\_AUDIT
- F\_DELETE\_RESCHEDULED\_TRIP

A record will be written to the ?administration log file? for each order record deleted to provide future reference.

Note that a delete action could be requested in the TripOrder XML message using tag <EVENT\_ACTION> which will have the same delete outcome as described above.

Creation:



If the whole trip will be replaced then the new ?TRM.F\_CREATE\_RESCHEDULED\_TRIP? function will be run to create the transport orders, haulage activities, trip stops and trip.

If part of the trip (i.e. some stops) will be replaced then the new ?TRM.F\_CREATE\_RESCHEDULED\_STOP? function will be run in sequence for the stop numbers not on ?locked? legs for the trip ID found for the route code provided; this will mean that the new replacement trip stops, haulage activities and transport orders are created for the existing trip.

N.B. The orders will only need to be created for the trip stops with loading activities for the external reference because it will be assumed that the transport order will be uploaded in the correct sequence for the trip stops.

New functions will be created in the ?TRM? package for each stage of the creation process:

- F\_CREATE\_RESCHEDULED\_ORDER\_ITEMS
- F\_CREATE\_RESCHEDULED\_ORDER
- F\_CREATE\_RESCHEDULED\_ACTIVITY
- F\_CREATE\_RESCHEDULED\_STOP
- F\_CREATE\_RESCHEDULED\_TRIP

The trip created will have a ?PLU? prefix which will be done by passing ?PLU? as the source when the ?TRM.INSERT\_TRIP? function is called.

The trip and order audit records will be created by the existing database triggers.

Once the trip and orders have been created they will be validated using the existing functions to ensure that they do not contain any errors.

N.B. If the deletion or creation processes have been unsuccessful then the data processing will be rolled back to a ?savepoint? and the interface status updated to ?FAILURE?.

### 3.3 Empty Stop Maintenance

A new system parameter called ?TRM\_RETAIN\_EMPTY\_STOPS? will be created at the cost-centre level to control the deletion or retention of trip stops without any activities.

Normally such stops would be deleted but they will be retained for PLUTO trips, therefore, the system parameter will be set to ?Y? for the Gamma cost centre.

If the system parameter is set to ?N? or has a NULL value then the empty stops will be deleted as normal.

The ?TRM.DELETE\_EMPTY\_STOPS? function will be changed to check the system parameter prior to proceeding: this means that the system parameter will be referenced wherever trip stops may be deleted in C-TMS.

### 3.4 C-TMS to Microlise Processing

Microlise will need to be advised of any changes to the trips that have been made by the upload of the files.

The existing database triggers will be able to ensure that a message is sent to Microlise with the latest trip data if the tractor or trailer assigned to the trip is enabled.

The ?TRG\_SCH\_TRIP\_XML\_INT? trigger operates when the trip is updated with a new carrier, driver or tractor, or when its status is set to ?ACCEPTED?, ?EN-ROUTE? or ?DELETED?: if the trip is deleted then the trigger will operate.

The ?TRG\_SHA\_XML? trigger operates when the haulage activity (i.e. a transport order at a stop) is inserted or deleted when the status of the trip is ?ACCEPTED? or ?EN-ROUTE?.

N.B. Messages will only be generated when the trip status is ?ACCEPTED? if the system parameter ?MIC\_SEND\_ACCEPTED? is ?Y?.

These triggers should generate messages for Microlise when the trip is deleted and replaced by PLUTO or when some of the stops are deleted and replaced.



## 3.5 C-TMS to PLUTO Processing

### 3.5.1 Handshake Confirmation Message

When creating trip stops in C-TMS from the interface message, the unique stop ID from PLUTO for the schedule and trip will be saved and will be sent back to PLUTO for the ?handshake? confirmation message.

If, and once, the trip and orders have been uploaded successfully into C-TMS a ?TripOrder? message will be sent back to PLUTO to confirm that the upload of the file has been completed. This file will use the current ?TripOrder? XML format and will contain the trip ID created along with the OMS references created. If a file is not sent back from C-TMS to PLUTO, this will indicate the trip and orders have not been successfully loaded.

The ?handshake? message is the confirmation that the trip has been created successfully in C-TMS.

The standard ?TripOrder? file will be generated immediately, should the ?handshake? message be required, when the inbound file has been confirmed as being processed successfully.

### 3.5.2 Actual Times Message

When an actual arrival or actual departure date and time is entered against a trip stop, a message will be triggered to PLUTO. The message will be in the ?TripOrder? XML format, and will contain the information for the trip that has been updated. The same leg lock strategy will be maintained in PLUTO based on the actual arrival or actual departure time available from the depot.

The ?TRG\_STS\_XML\_OUT? database trigger will be changed to write a record to the ?INT\_XML\_CONTROL? interface control database table for the ?PLUTO\_TRIP\_ACTUALS\_XML\_OUT? EDI database job to generate the message.

The new trigger point will be when the actual arrival or actual departure date and time has been changed for a PLUTO trip stop (i.e. prefixed with ?PLU?).

A check will be performed to ensure that there is not an interface control record waiting to be processed before a new record is written.

The following values will be set:

Column	Value
EXTERNAL_SYSTEM	?PLUTO_TRIP_ACTUALS_XML_OUT?
EVENT_TYPE	?TRP?
TRIP_ID	SCH_TRIP.TRIP_ID

The database job will call the ?INT\_XML\_OUT2.GEN\_TRIP\_XML? procedure with the process name (i.e. ?PLUTO\_TRIP\_ACTUALS\_XML\_OUT?) to process the interface control record.

The parameter ?PRINT\_ORD\_SECTION? will control whether the order sections are included in the message and it is expected that they will not be required.

### 3.5.3 Route Code Tag

The ?<ROUTE\_CODE>? tag in the outbound ?TripOrder? XML file will need to be included for PLUTO trips.

A new system parameter called ?DISPLAY\_TRIP\_ORDER\_ROUTE\_CODE? will be created at the cost-centre level, if the system parameter is set to ?Y? then the tag will be displayed in the XML file.

The ?INT\_XML\_OUT2.GEN\_TRIP\_DET? procedure will be changed so that the tag is populated with the route code of the trip (i.e. ?SCH\_TRIP.ROUTE\_CODE?) when the new system parameter is set for the cost-centre of the trip.

Therefore, the route code in C-TMS will contain the route code received from PLUTO (e.g. ?BEL DD3-1? where 1 is the shift number).



## 3.6 Interface Errors

The C-TMS interface errors forms will be available to allow the status of uploaded messages to be reviewed. This will include ?SUCCESS? and ?FAILURE?. Any failures can be found and reviewed and error messages will indicate the failure reason.

The ?XML Trip? tab page can be used to enquire on the trips and stops and the ?XML Orders? tab page can be used to enquire on the transport orders.

N.B. It will not be possible to reprocess the failed files in the ?Interface Errors? screen and new files will need to be provided if a failure has occurred.

### Table Updates Required

No table changes are required for this development.

### Modules to be changed

Module Name	Module Type	Notes
INT_XML_IN.sql	Package	New inbound trip/order processing for PLUTO files.
INT_XML_OUT2.sql	Package	New outbound trip/order processing for PLUTO files.
TRM.sql	Package	New trip/order processing for PLUTO.
TRG_STS_XML_OUT.sql	Trigger	New trigger point for the actual times message.

### References

Ref No	Document Title & ID	Version	Date
1	EST-292349 PL-8LZMAN PLUTO-C-TMS Integration v1.0.doc	1.0	04/10/11

### Glossary

Term or Acronym	Meaning
C-TMS	Calidus TMS
PLUTO	DHL in-house planning tool

### Document History

Version	Date	Status	Reason	Initials
0.1	19/10/11	Draft	Initial version	PDR
0.2	19/10/11	Draft	Reviewed	MJC
0.3	21/10/11	Draft	Updated after review	PDR
1.0	24/10/11	Issue	Reviewed and Issued	MJC
2.0	4/11/11	Issue	Clarification of Route Code including shift number & deletion of C-TMS trip as an interfaced event action from PLUTO	DJM



## 4 AUTHORISED BY

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

