

# **FleXipod Interface**

Aptean Ltd  
Copyright © 2011-2025.

# Contents

- 1 Flexipod Interface.....1
  - 1.1 Configuration.....1
  - 1.2 Sending Trips to FlexiPOD.....3
  - 1.3 Actions Taken on Update.....4

# 1 Flexipod Interface

**Warning:** This is an incomplete guide.

This guide shows how the Aptean Proof of Delivery system (APOD or formerly Flexipod) can be configured to be used from CTMS.

This requires configuration as to when messages are sent, how they are formatted (the Flexipod attributes), at what level are messages sent (orders, stops, etc) and what information is expected to be received.

## 1.1 Configuration

### 1.1.1 System Parameters

Parameter	Description	Config By
FLEXIPOD	Set to Y to indicate the FlexiPod interface is active	SYSTEM
FLEXIPOD_ACID	Flexipod acid	SYSTEM
FLEXIPOD_BREAK_DROPS	Flexipod include breaks for the drops in the Add API call (Y/N)	SYSTEM
FLEXIPOD_CARRIERS	Set to Y indicates the Carrier can control if a trip is sent to FlexiPod	SYSTEM
FLEXIPOD_CLIENTNUM	Flexipod ClientNum sent in interface	SYSTEM
FLEXIPOD_DEPOTS	Set to Y indicates the Depot can control if a trip is sent to FlexiPod	SYSTEM
FLEXIPOD_DIRECT	Set to Y to indicate the FlexiPod interface is direct and not using Route Execution	SYSTEM
FLEXIPOD_DROP	Determines the Drop level STOP-OMS or ITEM	SYSTEM
FLEXIPOD_EXPIRY	Sets the expiry date for the manifest as a number of days after the end of the trip.	SYSTEM
FLEXIPOD_EXTERNALSYSTEMUPDATE	Flexipod process the External System Update messages in the API call (Y/N)	SYSTEM
FLEXIPOD_FORCEITEMSCAN	Force the items to be scanned separately: 0 to use the ScanLevel or 1 to force the scan by item.	SYSTEM
FLEXIPOD_LISTPENDINGMANIFESTS	Flexipod process the List Pending Manifest Update messages in the API call (Y/N)	SYSTEM
FLEXIPOD_MERGE_DROPS	Flexipod merge the orders on the drop in the Add API call (Y/N)	SYSTEM
FLEXIPOD_PASSWORD	Flexipod Webservice password	SYSTEM
FLEXIPOD_PROXY	Flexipod proxy server	SYSTEM
FLEXIPOD_READDROP	Flexipod process the Read Drop messages in the API call (Y/N)	SYSTEM
FLEXIPOD_SCAN_LEVEL	1,2 or 3 to indicate line or item level scanning	SYSTEM
FLEXIPOD_SCAN_TYPE	Contents and items	SYSTEM
FLEXIPOD_SIGNATORY	Set to Y to store the SignedByName as the actual signatory as information for the order at its collection of delivery location	SYSTEM
FLEXIPOD_SPLIT	Are orders to be split?	SYSTEM
FLEXIPOD_SYSTEM	Flexipod SYSTEM ID sent in interface	SYSTEM
FLEXIPOD_TPCLIENTNUM	Flexipod TPClientNum sent in interface	SYSTEM
FLEXIPOD_USERCODE	Flexipod webservice usercode	SYSTEM
FLEXIPOD_USERNAME	Flexipod webservice username	SYSTEM
FLEXIPOD_WEBSERVICE	Flexipod webservice endpoint URL	SYSTEM
FLEXIPOD_WEBSERVICE_TASK	Flexipod webservice endpoint URL for Tasks	SYSTEM
FLEXIPOD_WEBSERVICE_UI	Flexipod webservice endpoint URL	SYSTEM
FLEXIPOD_WEBSERVICE_UI_EXEC	Flexipod webservice endpoint URL for Route Execution	SYSTEM
FLEXIPOD_X_API_KEY	Flexipod x-api-key for External System Update messages	SYSTEM



### 1.1.2 Carrier

The FLEXIPOD flag must be enabled when configured by carrier (system parameter FLEXIPOD\_CARRIERS) for carriers that require the use of FlexiPOD.

### 1.1.3 Depot Locations

The FLEXIPOD flag must be enabled when configured by depot (system parameter FLEXIPOD\_DEPOTS) for depots that require the use of FlexiPOD.

### 1.1.4 EDI Processes

#### FLEXIPOD\_OUT

This process sends the triggered manifests to APOD.

- Flow Type: PROCESS
- Package Name: PROCESS, DP\_FLEXIPOD.PROCESS\_FLEXIPOD\_OUT
- Process: i\_process\_name FLEXIPOD\_OUT
- Interval: 5 minutes

#### FLEXIPOD\_SENDLISTPENDING

This process checks to see if there are any pending updates from the manifests and imports the details.

- Flow Type: PROCESS
- Package Name: PROCESS DP\_FLEXIPOD.send\_listpendingmanifestupdates
- Process: PROCESS FLEXIPOD\_SENDLISTPENDING
- Interval: 1 minute

### 1.1.5 Reason Codes and Decodes


A decode of name "FLX\_RESULT\_CODES" type "REFERENCE" must be configured to map all of the reason codes from APOD into CTMS.

Several ITEM\_NON\_CON reason codes must be created to map to the APOD reason codes.:

Examples:

APOD	CTMS	Description	At Fault	Action	Type	CBR
1	SD	Successful Delivery				
10	SD	Successful Delivery				
2	FV	Failed Visit			FAILURE	Y
3	SD	Successful Delivery				
4	UD	Unsuccessful Delivery			FAILURE	
5	SC	Successful Collection				
6	SJ	Skipped Job			FAILURE	Y
7	NR	No Longer Required			FAILURE	
8	PR	Payment Refused				

Plus any reason codes created in FlexiPOD must be created within TMS for the audit history to accurately represent any reasons selected by the driver.

 **Note:** For each implementation, the reason codes and numbers will differ. For example, 1 and 10 may not be "Successfully Delivered", so the decode mapping MUST be in place for the APOD interface.



## 1.1.6 Attribute Configuration

Attributes against Manifests, Drops and Items can be configured by your Aptean implementation team. Up to **Warning:** 10 attributes may be defined at each level except Drop, which may contain 20 attributes.

In general, the following is allowed at each level:

- Manifest
  - ◆ Information derived directly from the trip, such as route code, trailer type, driver, etc.
- Drop
  - ◆ Information derived from the transport order header, such as the main references, instructions
  - ◆ Information derived from the customer of the transport order, such as parameters against that customer.
  - ◆ Any sub-reference held against that transport order.
  - ◆ Any sub-reference held against the transport order's destination location.
- Item
  - ◆ Information derived from the order items.
  - ◆ Order Reference information
  - ◆ Any references from the Drop above
  - ◆ **Note:** Including Quarantine status through the order reference STATUS)

An example is below:

Project Name	Level	Number	Source	Target
Stapletons	Drop	1	so.booking_ref	
Stapletons	Drop	2	so.del_point_ref	
Stapletons	Drop	3	dp_flexipod.get_order_val('ORDER','PRICE',	
Stapletons	Drop	4		
Stapletons	Drop	5		
Stapletons	Drop	6	dp_flexipod.get_ref('LOCATION','PERMIT_NUMBER',	
Stapletons	Drop	7	oc.free_text1	
Stapletons	Drop	8	dp_flexipod.get_ref('LOCATION','UNITARY_AUTHORITY',	
Stapletons	Drop	9	dp_flexipod.get_ref('LOCATION','LICENCE_NUMBER',	
Stapletons	Drop	10	so.total_weight	
Stapletons	Drop	11	so.total_pieces	
Stapletons	Drop	12	oc.pay_on_delivery	
Stapletons	Drop	13	oc.collect_casings	
Stapletons	Drop	14		
Stapletons	Drop	15		
Stapletons	ItemAttributes	1	soi.du_type	
Stapletons	ItemAttributes	2	soi.weight	
Stapletons	ItemAttributes	3	dp_flexipod.get_order_val('ITEM','UNIT_WEIGHT',	
Stapletons	ItemAttributes	4	dp_flexipod.get_order_val('ITEM','UNIT_PRICE',	
Stapletons	ItemAttributes	5	dp_flexipod.get_order_val('ITEM','STATUS',	
Stapletons	Manifest	1	st.route_code	
Stapletons	Manifest	2	st.trailer_type	

## 1.2 Sending Trips to FlexiPOD

**Warning:** This has changed

This happens automatically when:

- the trip is set to status "ACCEPTED" or "EN-ROUTE"
- when any changes happen to the trip i.e. orders are added, removed or moved on a trip that is at those statuses.
- When the driver, carrier or tractor has been added or changed on a trip that is at those statuses.



## 1.3 Actions Taken on Update

Gets updates from Flexipod drops and processes them

Marks message as collected

Stores order references

- TrackingID
- TrackingURL
- JobReportName

Stores signatory and filename if present

- As ord\_information
- As References
  - ◆ SignedByName
  - ◆ SignedByFilename
  - ◆ SiteImageFilename

If item details provided:

- Updates stop Actual Arrive and Actual Depart
- If scan\_type = 'I'
  - ◆ Creates Order Item Reasons from decoded reason code for actions against the drops and items (such as successful delivery, etc).
  - ◆ Creates Order Item Reasons from decoded reason code if there are non-conformities
  - ◆ Updates item delivered quantities
  - ◆ Updates order line quantities
  - ◆ Updates POC/POD flag against the order
- If scan\_type = 'P'
  - ◆ Same for each item in the pallet.

If event\_name = 'Pre-Departure'

- Updates SU stop actual arrive

If event\_name = 'Depot Depart'

- Updates SU stop actual arrive and actual depart
- Sets the trip status
- Updates items as despatched
- Updates lines as despatched
- Validates the order

If event\_name = 'Depot Return'

- Updates CL stop actual arrive

If event\_name = 'End of Trip'

- Updates CL stop actual arrive and actual depart
- Sets the trip status

For all order updates:

- Change order status to DELIVERED if anything delivered
- Change order status to FAILED if nothing delivered



**Note:** All changes to order are audited.

