dentsu

TRACKING

Dentsu International
DATA DICTIONARY v2.0.1

This document details the Data Dictionary for EU Secondary Repository and Router.

Summary of changes

Date	Version	Done by	Comment
17.01.2019	0.1	Dentsu Aegis Network	Internal Draft
05.02.2019	0.2	Dentsu Aegis Network	First Draft shared with stakeholders
21.02.2019	1.0	Dentsu Aegis Network	First release
08.03.2019	1.1	Dentsu Aegis Network	Minor changes
28.03.2019	1.2	Dentsu Aegis Network	Minor changes
16.09.2019	1.3	Dentsu Aegis Network	Addition of the GS1 EPCIS, EDI interface. Improvement of the Validation and address field split.
16.10.2019	1.4	Dentsu Aegis Network	Addition of the validation error codes and the EventTimeLong field.
17.10.2019	1.4.1	Dentsu Aegis Network	• Update of the Message Time Long. • Update the definition of the Arrival validation to remove the FID control. • Update the VAL_EVT_24H definition to a warning returning a Http status 299.
20.12.2019	1.4.2	Dentsu Aegis Network	Update EOID, FID and MID, Error codes
31.10.2020	1.4.3	Dentsu Aegis Network	Update of the country codes in preparation for Brexit. Consolidation of optional features into the main documents
15.04.2021	1.4.4 DRAFT	Dentsu Aegis Network	Clerical changes: • Moving the definition of the technical and the business validation from the list of specification to the Data Dictionary • IDA sample. • Augment the M_Producer, M_Model, M_Number to 200 characters.
21.05.2021	1.4.4 DRAFT2	Dentsu Aegis Network	Update of the sequence validation summary table.
15.06.2021	1.4.4	Dentsu Aegis Network	Update of the technical Regular expressions in the datatype table.
07.07.2022	1.4.5 DRAFT	Dentsu Aegis Network	Validation • Relaxation of the location validation on aggregation and disaggregation performed outside the EU • Event Time and Message time long technical validation Optional feature • Optional extension of the deactivation event 2.3 • Optional disaggregation acknowledgement confirmation
26.08.2022	1.4.5 DRAFT2	Dentsu Aegis Network	Add the Total upUI limitation. Update the time validation to 72H
08.09.2022	1.4.5	Dentsu Aegis Network	Publication
10.04.2023	1.4.6 DRAFT	Denstu International	Added Kosovo in the list of accepted country codes & creation of the new "XK" code for Kosovo.

21.08.2023	2.0.0 Draft	Dentsu International	Updated IR implementing regulation of 01.03.2023
21.09.2023	2.0.0	Dentsu International	Amended based on the feedback received on the Draft. Please see the "List of Changes" document for all the details
26.09.2023	2.0.0	Dentsu International	Removed the possibility to automatically register a mobile part under some circumstances.
31.10.2023	2.0.0	Dentsu International	Minor corrections on fields and data types to align the entire document.
20.02.2024	2.0.1 Draft	Dentsu International	• All changes are highlighted in yellow in the document. • Extended the LUQ to be accessible (with limitations) to ID Issuers and Primary Repository Providers. • Renamed some LUQ fields to be consistent with the rest of the document. • 3.6.1.2 Changed the time description from "Intended Time" to "Time" to align with the regulation. No technical impact. • 3.7.1.5 Corrected the example. • 3.11 Renamed to "Query API", whole section rewritten to improve readability and comply with the new requirements for ID Issuers and Primary Repository providers. • Minor lexical/format corrections. • Extended the LUQ response fields to ETL events on query type ID = 4
15.03.2024	2.0.1	Dentsu International	• Amendment to the list of country codes
24.04.2024	2.0.1	Dentsu International	• Clarification on Parameters for LUQ messages • Correction of endpoint for LUQ

Distribution

Date	Version	Submitted to
05.02.2019	0.2	Mail to stakeholder
21.02.2019	1.0	Published
08.03.2019	1.1	Published
28.03.2019	1.2	Published
16.09.2019	1.3	Published
16.10.2019	1.4	Published
17.10.2019	1.4.1	Published
20.12.2019	1.4.2	Published
31.10.2020	1.4.3	Published
15.04.2021	1.4.4 DRAFT	Mail to Primary Providers and ID Issuer
21.05.2021	1.4.4 DRAFT2	Mail to Primary Providers and ID Issuer
15.06.2021	1.4.4	Published
07.07.2022	1.4.5 DRAFT	Mail to Primary Providers and Competent Authorities
26.08.2022	1.4.5 DRAFT2	Mail to Primary Providers and Competent Authorities
08.09.2022	1.4.5	Published
17.04.2023	1.4.6 Draft	Mail to Primary Providers and ID Issuer
21.08.2023	2.0.0 Draft	Circulated to all stakeholders
21.09.2023	2.0.0	Circulated to all stakeholders
08.03.2024	2.0.1 Draft	Circulated to all stakeholders

15.03.2024	2.0.1	Circulated to all stakeholders
24.04.2024	2.0.1	

Confidentiality Statement

The information contained in these documents is confidential, privileged and only for the information of the intended recipient and may not be used, published or redistributed without the prior written consent of Dentsu International.

1 Contents

Table of Figures

1 Introduction

This document defines a data dictionary for Dentsu Tracking System. It will include information about data base entities and flows, authentication, operational and transactional methods, security edge case, router definition, error messages, registration process, message validation and an overall connection diagram.

Note: For the description of the Repositories system components, architecture, processes, data flows, list of interfaces and messages, see the List of Specifications document.

2 Data description

2.1 Data types

There are some types used along the document, which need to be defined.

Data Type	Description	Type	Example or regular expression
ARC	Administrative Reference Code (ARC) or any successive code adopted under the Excise Movement and Control System (EMCS)	Text(30)	'15GB0123456789ABCDEF0' Validation RegEx: ^[a-zA-Z0-9]*$
aUI	Aggregated level unique identifier coded with: either The invariant set of ISO646:1991 and composed of four blocks: (a) ID issuer's prefix in accordance with ISO15459-2:2015, (b) serialization element in the format established by the ID issuer, (c) tobacco facility identifier code following the Data Type: FID and (d) timestamp following the Data Type: Time(s) or The invariant set of ISO646:1991 forming a code structured in accordance with ISO15459-1:2014 or ISO15459-4:2014 (or their latest equivalent))	Text(100)	
Boolean	Boolean value	Boolean	• 0 (false/disabled) • 1 (true/enabled)
Component	A data type defined in the data dictionary		Aggregation
Country	Country name coded with ISO-3166-1:2013 alpha-2 (or its latest equivalent). For overseas and autonomous regions, the country code of the relevant Member State is applicable.	Text(2)	'DE'

	Please refer to section 2.6.1 for the list of countries including any applicable exception.		
Currency	Currency name coded with ISO 4217:2015 (or its latest equivalent)	Text(3)	'EUR'
Date	A UTC data in text corresponding to the following format: YYYY-MM-DD	Text(10)	E.g. '2017-03-31'
Decimal	Number values, decimal allowed	Decimal	E.g. '1' or '22.2' or '333.33'
Email	Maximum 80 characters	Text(80)	E.g. 'info@test.com' Validation RegEx: ^((([a-zA-Z]|\d|[!#\$%&'*\+\-\/=\?\^_`{|}~]|[\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-\uFFEF])+(\.([a-zA-Z]|\d|[!#\$%&'\*\+\-\/=\?\^_`{|}~]|[\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-\uFFEF])+)*)|((\x22)((((\x20|\x09)*(\x0d\x0a))?(\x20|\x09)+)?(([\x01-\x08\x0b\x0c\x0e-\x1f\x7f]|\x21|[\x23-\x5b]|[\x5d-\x7e]|[\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-\uFFEF])|(\\([\x01-\x09\x0b\x0c\x0d-\x7f]|[\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-\uFFEF])))*(((\x20|\x09)*(\x0d\x0a))?(\x20|\x09)+)?(\x22)))@((([a-z]|\d|[\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-\uFFEF])|(([a-z]|\d|[\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-\uFFEF])([a-z]|\d|-|\.|_|~|[\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-\uFFEF])*([a-z]|\d|[\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-\uFFEF])))\.)+(([a-z]|[\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-\uFFEF])|(([a-z]|[\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-\uFFEF])([a-z]|\d|-|\.|_|~|[\u00A0-\uD7FF\uF900-\uFDCF\uFDF0-\uFFEF])*([a-z]|[\u00A0-

			\uD7FF\uF900-\uFDCF\uFDF0-\uFFEF]))\.?$
EOID	Economic operator identifier code corresponding to the format established by ID issuer coded with the invariant set of ISO646:1991 EOID starts with the alphanumeric characters that constitute the ID issuer identification code, followed by alphanumeric sequence which is unique within the code pool of the ID issuer.	Text(50)	
EO_CODE	EO_CODE established by ID issuer coded with the invariant set of ISO8859-15:1999	Text(50)	
FID	Tobacco facility identifier code corresponding to the format established by ID issuer coded with the invariant set of ISO646:1991	Text(50)	
Integer	Rounded number values, no decimal numbers	Integer	E.g. '1' or '22' or '333'
IIID	ID Issuer code in line with the issuing agency codes of ISO/IEC 15459	Text(35)	E.g. 'FTR'
ITU	Individual transport unit code (e.g. SSCC) generated in accordance with ISO15459-1:2014 (or its latest equivalent)		'00791234560000000018'
List	Must be only one of the values present in the 'Values' column		
MID	Machine identifier code corresponding to the format established by ID Issuer coded with the invariant set of ISO646:1991	Text(50)	
MRN	Movement Reference Number (MRN) is a unique customs registration number. It contains 18 digits and is composed of the following elements: (a) last two	Text(18)	'19IT9876AB88901235' Validation RegEx:

	digits of the year of formal acceptance of export movement (YY), (b) country name coded with ISO3166-1:2013 alpha-2 (or its latest equivalent) of the Member State to which the declaration was sent, (c) unique identifier for entry/import per year and country, and (d) check digit.		^[0-9]{2}[A-Z]{2}[a-zA-Z0-9]+[0-9]{1}$
PN	Product number – numeric identifier used in the EUCEG system to identify product presentations (e.g. GTIN (Global Trade Identification Number) of the product)	Text(30)	'00012345600012'
SEED	Excise number composed of: (a) country name coded with ISO-3166-1:2013 alpha-2 (or its latest equivalent) (e.g. 'LU') and (b) eleven alphanumeric characters, if needed, padded to the left with zeroes (e.g. '00000987ABC').	Text(13)	'LU00000987ABC' Validation RegEx: ^[A-Z]{2}[a-zA-Z0-9]{11}$
Serial	Number corresponding with the invariant set of ISO646:1991 used for serialisation		
SSCC	SSCC-18 container code generated in line with ISO6346:1995 (or its latest equivalent)	Text(20)	00791234560000000018
Text (X)	Set of characters coded with ISO8859-15:1999 limited to X characters		E.g. 'Abcd' or '123455588845' or 'Abcde:12345'
Time(s)	UTC (Coordinated Universal Time) time in the following format: YYMMDDhh	Text(8)	'19071619'
Time(L)	UTC Time in the following format format : YYYY-MM-ddTHH:mm:ss.SSSZ		E.g. '2020-08-13T16:01:34.477Z'
TPID	Tobacco Product Identifier (TP-ID) – numeric identifier used in the EU-CEG system in the format: NNNNN-NN-NNNNN	Text(14)	'02565-16-00230' Validation RegEx: ^[0-9]{5}-[0-9]{2}-[0-9]{5}$

upUI(L)	Unit packet level unique identifier coded with the invariant set of ISO646:1991 and composed of three blocks: (i) ID Issuer's prefix in line with ISO154592:2015, (ii) middle block in the format established by ID Issuer and (iii) timestamp following the Data Type: Time(s)		
upUI(i)	Unit packet level unique identifier coded with the invariant set of ISO646:1991 and composed of two blocks: (i) ID issuer's prefix in accordance with ISO15459-2:2015 and (ii) middle block in the format established by the ID issuer (i.e. upUI(i) is upUI(L) without the timestamp, a code to be generated by ID issuers in accordance with Article 8(2) of this Regulation)		
upUI(s)	Unit packet level unique identifier coded with the invariant set of ISO646:1991 and composed of two blocks: (i) ID issuer's prefix in accordance with ISO15459-2:2015 and (ii) serialisation element in the format established by the ID issuer (i.e. UI made visible in the human readable format on the unit packets in accordance with Article 23 of this Regulation) Whenever possible ID issuers are requested not to use upper-case letter "O" (Oscar) and lower-case letter "l" (lima) as well as upper-case letter "I" (India) in order to avoid confusions with digits "0" (zero) and "1" (one), respectively.		

2.2 Priority types

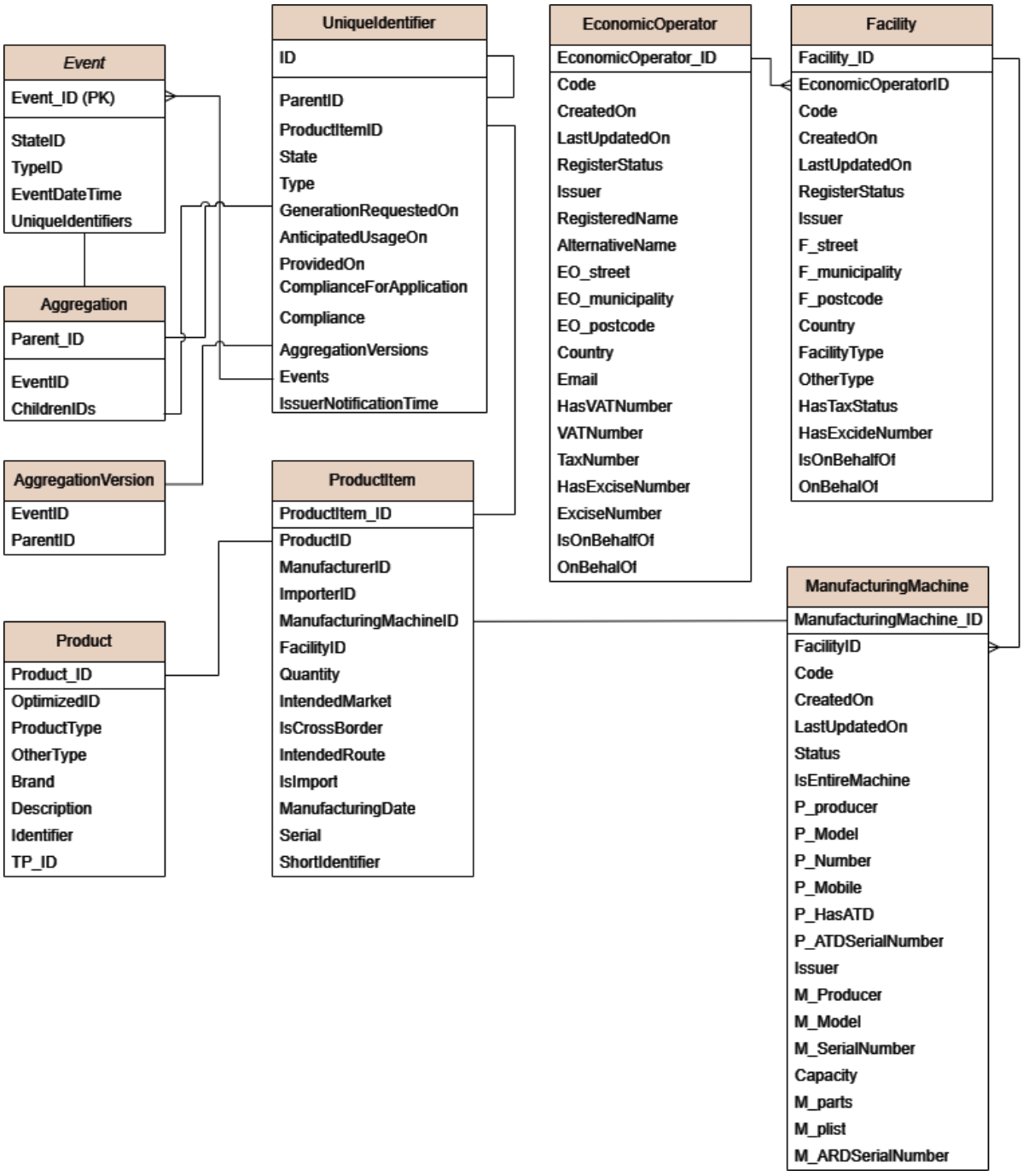
Type	Explanation
Mandatory (M)	The variable must be completed.
Optional (O)	The variable is for optional fields which could be filled depending on the record status or type.

2.3 Cardinality types

Type	Explanation
Simple (S)	Single value
Multiple (M)	Multiple values

2.4 Minimum Data model

The minimal data model describes the contents, format, and structure of a database and the relationship between its different elements.
Note: the Minimum Data Model may be extended.



2.4.1 Event

Field	Description	Data Type	Mandatory	Comments
Id *(PK)*	Internal identification number of this event	Integer	M	
StateID	The state of the event	EventState ID	M	EventState Type

TypeID	The type of the event	EventType ID	M	EventType Type
EventDateTime	Date and Time when the event occurs	Time(L)	M	
Content	Full content of the event.	Component	M	

2.4.2 Product

Field	Description	Data Type	Mandatory	Comments
Id (PK)	Internal identification number. This number is generated by the ID Issuer	Text(4)	M	
EO_ID (FK)	Economic operator identifier code of the submitting entity (either EU manufacturer or EU importer)	EOID	M	
F_ID	Facility identifier code	FID	M	
Process_Type	Indication if the production process involves machinery	Boolean	M	0 – No (only for fully hand made products) 1 – Yes
M_ID	Machine identifier code	MID	O	
P_Type	Type of tobacco product	Integer	M	See TobaccoProduct Type
P_OtherType	Description of other type of tobacco product	Text(200)	M, if P_Type = 12 (other tobacco product)	
P_CN	Combined Nomenclature (CN) code	Text(200)	O	
P_weight	Average gross weight of unit packet, including packaging, in grams with 0,1 gram accuracy	Decimal	M	
P_Brand	Brand of tobacco product	Text(200)	M	
TP_ID	The identification number of the product used in the EU-CEG system.	TPID	M, if Intended_Market is an EU country	
TP_PN	Tobacco product number used in the EU-CEG system	PN	M, if Intended_Market is an EU country	
Intended_Market	Intended country of retail sale.	Country	M	
Intended_Route1	Indication if the product is intended to be moved across country boarders with terrestrial transport.	Boolean	M	0 – No 1 – Yes
Intended_Route2	The first country of terrestrial transport after the product leaves the Member State of manufacturing or the	Country	M, if Intended_Route1 = 1	

	Member State of importation.			
Import	Indication if the product is imported into the EU	Boolean	M	0 – No 1 – Yes
Req_Quantity	Requested quantity of unit packet level UIs	Integer	M	
P_OtherID	Optional Product ID	Text(20)	O	

2.4.3 TobaccoProductItem

Field	Description	Data Type	Mandatory	Comments
Id (PK)	The identification code (i.e. unique identifier) of the product item as required by Article 15(2)	upUI(L)	M	
ProductId (FK)	The identification code of the product	Product Id	M	
ManufacturerId (FK)	Identifier of the manufacturer of this tobacco product	MID	M	
ImporterId (FK)	The identifier of the importer into the Union, if applicable	EOID		
ManufacturingMachine_Id (FK)	The identifier of the manufacturing machine	Manufacturing Machine MID	M	
FacilityId (FK)	The identifier of the manufacturing facility. This date is the one used for requesting the issuance of codes.	Facility FID	M	
IntendedMarket	Intended country of retail sale	Country	M	
IsCrossBorder	Indication if the product is intended to be moved across country boarders with terrestrial transport	Boolean	M	0 – No 1 – Yes
IntendedRoute	The first country of terrestrial transport after the product leaves the Member State of manufacturing or the Member State of importation	Country		M, if Intended_Route1 = 1
IsImport	Indication if the product is imported into the EU	Boolean	M	0 – No 1 – Yes
ManufacturingDate	Date of manufacturing. This date is the one used for requesting the issuance of codes	Time(s)	M	
Serial	Serial number provided by the ID Issuer	Serial	M	
ShortIdentifier	Short unique identifier	upUI(s)		

2.4.4 UniqueIdentifier

Field	Description	Data Type	Mandatory	Comments
ID (PK)	Unique identifier of the unit packets or aggregated packaging level	Text(50)	M	
State	The state of the unique identifier	UniqueIdentifierState ID	M	UniqueIdentifierState Type
Type	The type of the unique identifier	UniqueIdentifierType ID	M	UniqueIdentifierType Type
GenerationRequestedOn	Date and Time when the generation was requested	Time (L)	M	
AnticipatedUsageOn	Date and Time when the generator intends to use it	Time (L)	M	
IssuerNotificationTime	Date and Time when the generation was notified to the storage	Time(L)	M	
ParentId	The identifier of the parent element that contains this item	UniqueIdentifier ID	O	

2.5 Registered entities

2.5.1 EconomicOperator

Field	Description	Data Type	Mandatory	Comments
Id (PK)	Economic operator's registered ID	EOID	M	
CreatedOn	Timestamp when the registration has been accomplished	Time(L)	M	
LastUpdatedOn	Timestamp of the last change on the register	Time(L)		
RegisterStatus	Status of the registration	Integer	M	RegisterStatus Type
Issuer	Identification number of the ID Issuer solution that has processed the registration	IIID	M	
EO_Name1	Economic operator's registered name	Text(100)	M	
EO_Name2	Economic operator's	Text(100)	O	

	alternative or abridged name			
EO_street	Economic operator's street name and house number (or road number and kilometer)	Text(300)	M	
EO_municipality	Economic operator's municipality (city, town or village)	Text(100)	M	
EO_postcode	Economic operator's postal code	Text(50)	M	'n/a' is permitted value if no postal code has been assigned
EO_A_info	Additional information on economic operator's address (e.g. location in the shopping mall or industrial area)	Text(100)	O	
EO_CountryReg	Economic operator's country of registration	Country	M	See Country
EO_Email	Economic operator's email address; used to inform about registration process, incl. subsequent changes and other required correspondence	Text(80) (Regex protected)	M	
VAT_R	Indication of the VAT registration status	Boolean	M	0 – No VAT registration 1 – VAT number exists
VAT_N	Economic operator's VAT number	Text(20)	M, if VAT_R = 1	
TAX_N	Economic operator's tax registration number	Text(20)	M, if VAT_R = 0	
EO_ExciseNumber1	Indication if the economic operator has an excise number issued by the competent authority for the purpose of identification of persons/premises	Boolean	M	0 – No SEED number 1 – SEED number exists

EO_ExciseNumber2	Economic operator's excise number issued by the competent authority for the purpose of identification of persons/premises	SEED	M, if EO_ExciseNumber1 = 1	
OtherEOID_R	Indication if the economic operator has been allocated an identifier by another ID Issuer	Boolean	M	0 – No 1 – Yes
OtherEOID_N	Economic operator identifier codes allocated by other ID Issuers	Text(5000)	M, if OtherEOID_R = 1	List of EOIDs
Reg_3RD	Indication if the registration is made on behalf of a retail outlet operator not otherwise involved in the tobacco trade	Boolean	M	0 – No 1 – Yes
Reg_EOID	Identifier of the economic operator that acts on behalf of a retail outlet operator not otherwise involved in the tobacco trade	EOID	M, if Reg_3RD = 1	
EO_OtherID	Optional identifier	Text(50)	O	
EO_CODE	Economic operator's confirmation code provided in response to the registration of economic operator	EO_CODE	M	

2.5.2 Facility

Field	Description	Data Type	Mandatory	Comments
EO_ID(FK)	Economic operator identifier code	EOID	M	
F_ID (PK)	Facility code from the RFA code issuer call	FID	M	
CreatedOn	Timestamp when the registration has been accomplished	Time(L)	M	

LastUpdatedOn	Timestamp of the last change on the register	Time(L)		
RegisterStatus	Status of the registration	Integer	M	RegisterStatus Type
EO_CODE	Economic operator's confirmation code provided in response to the registration of economic operator	EO_CODE	M	
F_street	Facility's street name and house number (or road number and kilometer)	Text(300)	M	
F_municipality	Facility's municipality (city, town or village)	Text(100)	M	
F_postcode	Facility's postal code	Text(50)	M	'n/a' is permitted value if no postal code has been assigned
F_A_info	Additional information on facility's address (e.g. location in the shopping mall or industrial area)	Text(100)	O	
F_Country	Facility's country	Country	M	See Country
F_Type	Type of facility	Integer	M	See FacilityType
F_Type_Other	Description of other facility type	Text(5000)	M, if F_Type = 4	
F_Status	Indication if a part of the facility has a bonded warehouse status	Boolean	M	0 - No 1 - Yes
F_ExciseNumber1	Indication if the facility has an excise number issued by the competent authority for the purpose of identification of persons/premises	Boolean	M	0 - No SEED number 1 - SEED number exists
F_ExciseNumber2	Facility's excise number issued by the competent authority for the purpose of identification of persons/premises	SEED	M, if F_ExciseNumber1 = 1	

OtherFID_R	Indication if the facility has been allocated an identifier by another ID Issuer	Boolean	M	0 - No 1 - Yes (possible only for non-EU facilities)
PrevFID_B	Indication if the facility was acquired from another operator and had already a facility identifier code	Boolean	M	0 – No (first time registration) 1 – Yes
PrevFID_ID	Previous facility identifier used by the former operator of the facility	FID	M, if PrevFID_B = 1	
OtherFID_N	Facility identifier codes allocated by other ID Issuers	Text(5000)	M, if OtherFID _R = 1	List of FID
Reg_3RD	Indication if the registration is made on behalf of a retail outlet operator not otherwise involved in the tobacco trade	Boolean	M	0 – No 1 – Yes (possible only if F_Type = 3)
Reg_EOID	Identifier of the economic operator that acts on behalf of a retail outlet operator not otherwise involved in the tobacco trade	EOID	M, if Reg_3RD = 1	
latitude	GPS coordinates, latitude value	Decimal	O	Optional information described in the revised Annex II
longitude	GPS coordinates, longitude value	Decimal	O	Optional information described in the revised Annex II

2.5.3 ManufacturingMachine

Field	Description	Data Type	Mandatory	Comments
M_ID (PK)	Machine identifier received from the RMA request made to the code issuer.	MID	M	
F_ID (FK)	Facility identifier code	FID	M	
CreatedOn	Timestamp when the registration has been accomplished	Time(L)	M	
LastUpdatedOn	Timestamp of the last change on the register	Time(L)		
Status	Status of the registration	Integer	M	RegisterStatus Type

PrevMID_B	Indication if the object of this request was already registered, e.g. in relation to another facility identifier code	Boolean	M	0 – No (first time registration) 1 – Yes
PrevMID_ID	Previous machine identifier used for the object of this request	MID	M, if PrevMID_B = 1	
M_entirety	Indication if this request concerns the machine (v. a part of thereof)	Boolean	M	0 – No (machine part) 1 – Yes (machine)
P_Producer	Part's producer	Text(20)	M, if M_entirety = 0	
P_Model	Part's model	Text(20)	M, if M_entirety = 0	
P_Number	Part's serial number	Text(20)	M, if M_entirety = 0	
P_Mobile	Indication if this part is intended to be used with multiple machines (fixed v. mobile part)	Boolean	M, if M_entirety = 0	
P_ATD1	Indication if an anti-tampering device in the sense of Article 2(7) records the functioning of this part	Boolean	M, if M_entirety = 0	
P_ATD2	Anti-tampering device's serial number	Text(100)	M, if M_entirety = 0 and P_ATD1 = 1	
P_Description	Part's description explaining its technical function	Text(500)	O	
M_Producer	Machine producer	Text(20)	M, if M_entirety = 1	
M_Model	Machine model	Text(20)	M, if M_entirety = 1	
M_Number	Machine serial number	Text(20)	M, if M_entirety = 1	
M_Capacity	Maximum capacity over 24hour production cycle expressed in unit packets	Integer	M, if M_entirety = 1	
M_parts	Indication if the machine consists of multiple separately identifiable parts	Boolean	M, if M_entirety = 1	0 – No 1 – Yes
M_plist	List of the identifiable parts	MID	M (limited to 1000 MID), if M_entirety = 1 and M_parts = 1	List of MID (parts)
M_ATD	Serial number of the anti-tampering device in the sense of Article 2(7)	Text(100)	M, if M_entirety = 1 and M_parts = 0	

2.6 Master Data Types

2.6.1 Country Codes

Code	Value
AD	Andorra
AE	United Arab Emirates
AF	Afghanistan
AG	Antigua and Barbuda
AI	Anguilla
AL	Albania
AM	Armenia
AO	Angola
AQ	Antarctica
AR	Argentina
AS	American Samoa
AT	Austria
AU	Australia
AW	Aruba
AX	Åland Islands
AZ	Azerbaijan
BA	Bosnia and Herzegovina
BB	Barbados
BD	Bangladesh
BE	Belgium
BF	Burkina Faso
BG	Bulgaria
BH	Bahrain
BI	Burundi
BJ	Benin
BL	Saint Barthélemy
BM	Bermuda
BN	Brunei Darussalam
BO	Bolivia (Plurinational State of)
BQ	Bonaire, Sint Eustatius and Saba
BR	Brazil
BS	Bahamas
BT	Bhutan
BV	Bouvet Island
BW	Botswana
BY	Belarus
BZ	Belize
CA	Canada
CC	Cocos (Keeling) Islands
CD	Congo, Democratic Republic of the
CF	Central African Republic
CG	Congo
CH	Switzerland
CI	Côte d'Ivoire
CK	Cook Islands
CL	Chile
CM	Cameroon
CN	China
CO	Colombia
CR	Costa Rica
CU	Cuba
CV	Cabo Verde
CW	Curaçao
CX	Christmas Island
CY	Cyprus
CZ	Czechia
DE	Germany
DJ	Djibouti
DK	Denmark
DM	Dominica
DO	Dominican Republic
DZ	Algeria
EC	Ecuador
EE	Estonia
EG	Egypt
EH	Western Sahara
ER	Eritrea
ES	Spain
ET	Ethiopia
FI	Finland
FJ	Fiji
FK	Falkland Islands (Malvinas)
FM	Micronesia (Federated States of)
FO	Faroe Islands
FR	France
GA	Gabon
GB	United Kingdom of Great Britain
GD	Grenada
GE	Georgia
GF	French Guiana
GG	Guernsey
GH	Ghana
GI	Gibraltar
GL	Greenland
GM	Gambia
GN	Guinea
GP	Guadeloupe
GQ	Equatorial Guinea
GR	Greece
GS	South Georgia and the South Sandwich Islands
GT	Guatemala
GU	Guam
GW	Guinea-Bissau
GY	Guyana
HK	Hong Kong
HM	Heard Island and McDonald Islands
HN	Honduras
HR	Croatia
HT	Haiti
HU	Hungary
ID	Indonesia
IE	Ireland
IL	Israel
IM	Isle of Man
IN	India
IO	British Indian Ocean Territory
IQ	Iraq
IR	Iran (Islamic Republic of)
IS	Iceland
IT	Italy
JE	Jersey
JM	Jamaica
JO	Jordan
JP	Japan
KE	Kenya
KG	Kyrgyzstan
KH	Cambodia
KI	Kiribati
KM	Comoros
KN	Saint Kitts and Nevis
KP	Korea (Democratic

	People's Republic of)
KR	Korea, Republic of
KW	Kuwait
KY	Cayman Islands
KZ	Kazakhstan
LA	Lao People's Democratic Republic
LB	Lebanon
LC	Saint Lucia
LI	Liechtenstein
LK	Sri Lanka
LR	Liberia
LS	Lesotho
LT	Lithuania
LU	Luxembourg
LV	Latvia
LY	Libya
MA	Morocco
MC	Monaco
MD	Moldova, Republic of
ME	Montenegro
MF	Saint Martin (French part)
MG	Madagascar
MH	Marshall Islands
MK	Macedonia, the former Yugoslav Republic of
ML	Mali
MM	Myanmar
MN	Mongolia
MO	Macao
MP	Northern Mariana Islands
MQ	Martinique
MR	Mauritania
MS	Montserrat
MT	Malta
MU	Mauritius
MV	Maldives
MW	Malawi
MX	Mexico
MY	Malaysia
MZ	Mozambique
NA	Namibia
NC	New Caledonia
NE	Niger
NF	Norfolk Island
NG	Nigeria
NI	Nicaragua
NL	Netherlands
NO	Norway

NP	Nepal
NR	Nauru
NU	Niue
NZ	New Zealand
OM	Oman
PA	Panama
PE	Peru
PF	French Polynesia
PG	Papua New Guinea
PH	Philippines
PK	Pakistan
PL	Poland
PM	Saint Pierre and Miquelon
PN	Pitcairn
PR	Puerto Rico
PS	Palestine, State of
PT	Portugal
PW	Palau
PY	Paraguay
QA	Qatar
RE	Réunion
RO	Romania
RS	Serbia
RU	Russian Federation
RW	Rwanda
SA	Saudi Arabia
SB	Solomon Islands
SC	Seychelles
SD	Sudan
SE	Sweden
SG	Singapore
SH	Saint Helena, Ascension and Tristan da Cunha
SI	Slovenia
SJ	Svalbard and Jan Mayen
SK	Slovakia
SL	Sierra Leone
SM	San Marino
SN	Senegal
SO	Somalia
SR	Suriname
SS	South Sudan
ST	Sao Tome and Principe
SV	El Salvador
SX	Sint Maarten (Dutch part)
SY	Syrian Arab Republic
SZ	Eswatini

TC	Turks and Caicos Islands
TD	Chad
TF	French Southern Territories
TG	Togo
TH	Thailand
TJ	Tajikistan
TK	Tokelau
TL	Timor-Leste
TM	Turkmenistan
TN	Tunisia
TO	Tonga
TR	Turkey
TT	Trinidad and Tobago
TV	Tuvalu
TW	Taiwan, Province of China
TZ	Tanzania, United Republic of
UA	Ukraine
UG	Uganda
UM	United States Minor Outlying Islands
US	United States of America
UY	Uruguay
UZ	Uzbekistan
VA	Holy See
VC	Saint Vincent and the Grenadines
VE	Venezuela (Bolivarian Republic of)
VG	Virgin Islands (British)
VI	Virgin Islands (U.S.)
VN	Viet Nam
VU	Vanuatu
WF	Wallis and Futuna
WS	Samoa
YE	Yemen
XI	Northern Ireland
XK	Kosovo
XZ	International waters
YT	Mayotte
ZA	South Africa
ZM	Zambia
ZW	Zimbabwe

2.6.2 DeactivationReasonType

Value	Name
1	Product destroyed
2	Product stolen
3	UI destroyed
4	UI stolen
5	UI unused
6	Other

2.6.3 EventState

Value	Name	Description
1	Received	Initial state. The Data Acquisition component has just received the event and stores it.
2	Valid	The Data Processing component has verified that the format and contents are correct.
3	Invalid	The Data Processing component has found some issues regarding the format or the contents. Event is promoted to invalid for further analysis by the storage provider.
4	Routed	The Data Processing component has routed (or copied) successfully the event to the other Data Storage.
5	ConsolidationInProgress	The Data Processing attempts to consolidate the information included in the event, if possible.
6	Consolidated	If the consolidation has been done, it is then promoted to Consolidated.
7	Orphaned	If the consolidation has not been possible because some prior events were missing, it is promoted to Orphaned.
8	Cancelled	Final state if the System receives a recall message regarding this event.

2.6.4 EventType

Value	Name
REO	Registration of an Economic operator
CEO	Correction for an economic operator identifier code
DEO	De-registration of economic operator identifier code
RFA	Request for a facility identifier code
CFA	Correction of information concerning the facility identifier code
DFA	De-registration of facility identifier code
RMA	Request for a machine identifier code
CMA	Correction of information concerning the machine identifier code
DMA	De-registration of machine identifier code
ICV	Identifier code verification
ICM	Validation of the IRU Message successful transmission to the Primary repository
ULO	Flat file and registry File upload
PLO	Partial Flat file and registry transmission
ISU	Request for unit level UIs
IRU	Response for unit level UIs
ISA	Request for reporting the issuance of serial numbers at aggregated level
IRA	Response for reporting the issuance of serial numbers at aggregated level
IDA	Request for deactivation of UIs
IRR	Request for reactivation of UIs
EUA	Application of unit level UIs on unit packets

EPA	Application of aggregated level UIs on aggregated packaging
EDP	Dispatch Event
EDX	Dispatch 3.8
ERP	Reception event
ETL	Trans-loading event
EUD	Message to report an UI disaggregation
EVR	Report the delivery carried out with a vending van to retail outlet
EIV	Message to report an invoice
EPO	Purchase order
EPR	Payment record
RCL	Recall messages
LUP	Download Offline flat file
CTM	Connectivity Test Messages

2.6.5 FacilityType

Value	Name
1	Manufacturing site with warehouse
2	Standalone warehouse
3	First retail outlet
4	Other

2.6.6 InvoiceType

Value	Name
1	Original
2	Correction
3	Other

2.6.7 NotificationType

Value	Name	Description
1	Informative	The notification only includes descriptive information, but not related to any error or abnormal situation.
2	Warning	The notification includes information about some alert or warning to be considered.
3	Alarm	The notification includes information about some alarm triggered by the System.
4	InternalError	The notification includes information about some error that has occurred within the System.
5	Other	The notification includes information about some other situation, not listed above, that has occurred within the System.

2.6.8 PaymentType

Value	Name
1	Bank transfer
2	Bank card
3	Cash
4	Other

2.6.9 RecallReasonType

Value	Name
1	Reported event did not materialise
2	Message contained erroneous information
3	Other

2.6.10 RegisterStatus

Value	Name
1	Registered
2	De-registered

2.6.11 TobaccoProductType

Value	Name
1	Cigarette
2	Cigar
3	Cigarillo
4	Roll your own tobacco
5	Pipe tobacco
6	Waterpipe tobacco
7	Oral tobacco
8	Nasal tobacco
9	Chewing tobacco
11	Novel tobacco product
12	Other (product placed on the market before 19 May 2014, not covered by categories 1-9)

2.6.12 TransportMode

Value	Name
0	Other
1	Sea Transport
2	Rail transport
3	Road transport
4	Air transport
5	Postal consignment
6	Fixed transport installations
7	Inland waterway transport

2.6.13 UniqueIdentifierState

Value	Name	Description
1	Generated	The ID Issuer reports the issue of some codes and the Secondary repository creates a unique identifier record with the initial state (i.e. Generated).
2	Activated	The unique identifier, after being verified by the manufacturer, matches one unique identifier stored in the Secondary repository under the status "Generated". Additionally, the information contained in the date element of information matches the valid activation date for that unique identifier.

3	Deactivated	The manufacturer reports the deactivation of that unique identifier. Other cause of deactivation is when manufacturers tries to activate a unique identifier whose date element of information does not match the valid activation date for that unique identifier.
4	Expired	The Secondary repository promotes to "Expired" the codes that have been issued, but their activation has not been reported within a certain period of time (i.e. expiration time).
5	Delivered	The distributor or wholesaler reports that this tobacco product item has been successfully dispatched to the final retailer.

2.6.14 UniqueIdentifierType

Value	Name	Description
1	UnitPacket	Unique identifier at unit packet level
2	AggregatedPackaging	Unique identifier at aggregated packaging level

3 Messages

3.1 Message types to be exchanged

Described in the Regulation Annex II "Key messages to be sent by the economic operators" 5 categories of messages, related to:

- Identifier codes for economic operators, facilities and machines
- Unique identifiers for unit level UIs and aggregated level UIs
- Recording and transmission of information on product movements
- Transactional events
- Recalls

The following table summarizes the JSON formatted messages.

Message Type	Annex II Reference	Message description
REO	(1.1)	Registration of an Economic operator
CEO	(1.2)	Correction for an economic operator identifier code
DEO	(1.3)	De-registration of economic operator identifier code
RFA	(1.4)	Request for a facility identifier code
CFA	(1.5)	Correction of information concerning the facility identifier code
DFA	(1.6)	De-registration of facility identifier code
RMA	(1.7)	Request for a machine identifier code
CMA	(1.8)	Correction of information concerning the machine identifier code
DMA	(1.9)	De-registration of machine identifier code
ICV		Identifier code verification
ICM		Validation of the delivery of an IRU message

ULO		Flat file and registry File upload
ULOD		Flat file and registry File upload callback
PLO		Partial Flat file and registry transmission
ISU	(2.1)	Request for unit level UIs
IRU		Response for unit level UIs
ISA	(2.2)	Request for reporting the issuance of serial numbers at aggregated level
IRA		Request for reporting the issuance of serial numbers at aggregated level
IDA	(2.3)	Request for deactivation of UIs
IRR	(2.4)	Request for reactivation of UIs for products reported as stolen but recovered
EUA	(3.1)	Application of unit level UIs on unit packets
EPA	(3.2)	Application of aggregated level UIs on aggregated packaging
EDP	(3.3)	Dispatch Event
ERP	(3.4)	Reception event
ETL	(3.5)	Trans-loading event
EUD	(3.6)	Message to report an UI disaggregation
EVR	(3.7)	Report the delivery carried out with a vending van to retail outlet
EDX	(3.8)	Dispatch of tobacco products from a facility to laboratories, waste disposal centres, national authorities, international governmental organisations, embassies and military bases
EIV	(4.1)	Message to report an invoice
EPO	(4.2)	Purchase order
EPR	(4.3)	Payment record
RCL	(5.0)	Recall messages
LUP		Download Offline flat file
LUQ		Query Messages
LDI		Lookup Dispatch Interface
CTM		Connectivity Test Messages

3.1.1 Optional II2MN II2DW interfaces

The ID Issuer defines the communication between the EO and the ID issuer corresponding to interfaces II2MN and II2DW.
The proposed messages presented in this Data Dictionary are sample messages to illustrate the overall flow of data from the EO to the

Secondary repository. These messages should be considered as a Guideline with no obligation of implementation.
All messages part of the II2MN and II2DW interfaces are marked as optional in this document.

3.1.2 Message and endpoints

		Message support
Primary Repository End Point		
Primary Endpoint	The primary endpoint	IRU,IRA,IDA,IRR,EUA,EPA,EDP,EDX, ERP,ETL,EUD,EVR,EIV,EPO,EPR,RCL
Router Endpoints		
Router	The authentication endpoint	
Router	The resource endpoint	IRU,IRA,IDA,IRR,EPA,EDP,EDX,ERP,ETL,EUD,EVR, EIV,EPO,EPR,RCL,CTM
Router	The flat file upload	ULO, PLO
Router	The verification endpoint	ICV, LDI
Secondary Repository Endpoints		
Secondary Repository	The resource endpoint	IRU,IRA,IDA,IRR,EUA,EPA,EDP,EDX,ERP,ETL,EUD, EVR,EIV,EPO,EPR,RCL,LUQ
Secondary Repository	The Offline flat file download	LUP

3.2 Common schema elements

3.2.1 Basic information block concerning the request

Basic information block concerning the request - schema					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
Message_Type	The identifier of the type of message	Text(Limited to set of known message_types)	S	M	See above types of messages list
Code	The internal code of acknowledgment of the message. Used for recall too.	Text(50)	S	M	property is nullable
RejectionData	The failure data recorded in the primary should the validation fail.	FailureData (See below table)	S	O	This should only be filled if the primary validation fails.
Reception_Time	System reception Time added by the Router or the Primary Repository	Time(L)	S	M for Router and Primary providers (Does not apply to EO)	

Note 1: The reception Time is added by the entry point traceability system (Router or primary repository). The reception Time provided by the router to the primary repositories should be maintained and the

primary repositories should accept the field and forward it to the secondary repository.
Note 2: The Code should be set to null for the initial request.

RejectionData - schema					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
ResponseText	The response of the primary	Text(5000)	S	M	
Errors	List of the errors. Array containing Error_Code, Error_Descr, Error_InternalId, Error_Data (string)	Array of objects	S	M	

If the secondary repository receives a message with this "RejectionData" non null, it will not process the message and will instead record / audit the failure. This for later analysis, used to find possible illicit trade.

3.2.2 Basic information block concerning the response

Basic information block concerning the response - schema					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
Message_Type	The identifier of the type of message that the response refers to	Text	S	M	See above types of messages list
Error	Indicates the failure of the message reception	Boolean	S	M	0 – No 1- Yes
Errors	Array containing Error_Code, Error_Data (string), Error_Descr, Error_InternalID	Array of objects	S	M if Error = 1	
Code	Unique identifier of the message. Used for recall too.	Text(50)	S	M	
Checksum	The calculated checksum of the data received	Text(5000)	S	M	

3.2.3 Basic Error block description

Data Type	Description	Type	Example or regular expression
Error_Code	Error code describing the error.	Text(30)	
Error_Data	Text field containing error related data such as values of attributes, list of UIs For all lists, use the # character as separator.	Text(5000)	

Error_Descr	Description of the error code, that must contain the related controls, related RecallCode and fields when applicable.	Text(5000)	
Error_InternalID	**Optional** internal ID of the error. This internal ID can be used for maintenance or audit purpose.	Text(50)	

```
{
   …
      "Errors": [
         {
              "Error_Code": "UI_NOT_EXIST",
              "Error_Descr": "Text describing the error code",
              "ErrorData": "CF12D12AB887#CFEEAB2AB887#CFEED12AB887#AB1212AB6395"
         }
      ],
   …
 }
```

3.2.4 Response Information block

Basic information block concerning the response - schema					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
Information_Type	The identifier of the type of information	Text(5000)	S	M	
Data	Indicates the failure of the message reception	Text(5000)	S	M	0 – No 1- Yes
Data_List	Array of strings	Text(string limit = 5000)	M	O	

```
{
   …
      "Information": [
         {
              "Info_Type": "TotalupUI",
              "Data": "5000"
         }
      ],
   …
 }
```

3.2.4.1 Information_Type

Information_Type	Description
TotalupUI	Total number of upUI present in the event. This optional field is supported on the following requests • EPA – (3.2) Application of aggregated level UIs on aggregated packaging • EDP – (3.3) Dispatch of tobacco products from a facility

Information_Type	Description
	• EDX – (3.8) Dispatch of tobacco products from a facility to laboratories, waste disposal centres, national authorities, international governmental organisations, embassies and military bases. • ERP – (3.4) Arrival of tobacco products at a facility • ETL – (3.5) Trans-loading event • EVR – (3.7) Report the delivery carried out with a vending van to retail outlet Please note, however, that there is a certain limitation of the feature which is that if **any of the underlying tobacco packs in the hierarchy represented by the aUIs sent in the message to check was produced before 1st of July 2020, the checksum returns "-1".**
ChildUIList	List of children This optional field is supported on the following requests • EUD – (3.6)

3.2.5 Empty array and null values

3.2.5.1 Empty array

An array structure is represented as square brackets surrounding zero or more values (or elements).

```
{
   …
      "Errors": [],
   …
 }
```

3.2.5.2 Null value

A JSON null value MUST be a literal named null.

```
{
   …
      "Code": null,
   …
 }
```

3.2.6 Decimal points

According to the JSON Standard RFC7159 a JSON decimal separator value MUST be a period ".".

```
{
   …
      "DecimalValue": 35.21
   …
 }
```

3.2.7 Common Error codes

HTTP status	Error Code	Error Description
401	SECURITY_INVALID_OR_EXPIRED_TOKEN	Invalid or Expired security token **Please note that in this case the recall code or internal ID is not returned, as the message has not reached the processing service yet**.
400	INVALID_SIGNATURE	Invalid signature
400	REQUIRED_FIELD_FAILED_VALIDATION MAX_LENGTH_FAILED_VALIDATION MIN_LENGTH_FAILED_VALIDATION ENTRY_LENGTH_FAILED_VALIDATION	When one or multiple fields do not contain valid format
400	PAYLOAD_NOT_UNIQUE	When the message has already been processed successfully.
400	INVALID_REQUEST_FORMAT	This error is returned when at least one of the mandatory fields are missing.
400	INVALID_MESSAGE_TYPE	When the field "Message_Type" is out of the defined list.
400	INVALID_INPUT_FORMAT	When the body of the message doesn't contain a valid JSON.
500	SYSTEM_ERROR	Internal system error.

3.3 Message Validation

3.3.1 Overview

Validation is the process to accept or reject an incoming message.

Upon the reception of reported events on UI, a certain number of calculated information called metadata are computed and managed internally on the Secondary repository and Router. The UI metadata consists of

- The **UI state information** (indicating if the UI is in stock in a location, in transit and other)
- The **UI Location FID** (current location or last known location)
- The **UI Aggregation context** (if the UI is part of an aggregation)
- The **UI Event history**

This metadata is internally used by the Secondary repository and the Router to perform the validation in real time and allows the system to meet the response time regulatory SLA.

Once the event is validated and accepted by the router or the secondary repository, the metadata for UIs are updated.

3.3.1.1 Principle: Duplication of validation

The complete traceability system should be considered as one system and the validation is performed at the first component of the system.

- Messages processed by the Router and transmitted to the Primary repository, should not be validated at the Primary repository level.
- Messages processed by the Primary repository and forwarded to the Secondary repository, is validated in order to ensure internal data integrity between systems.

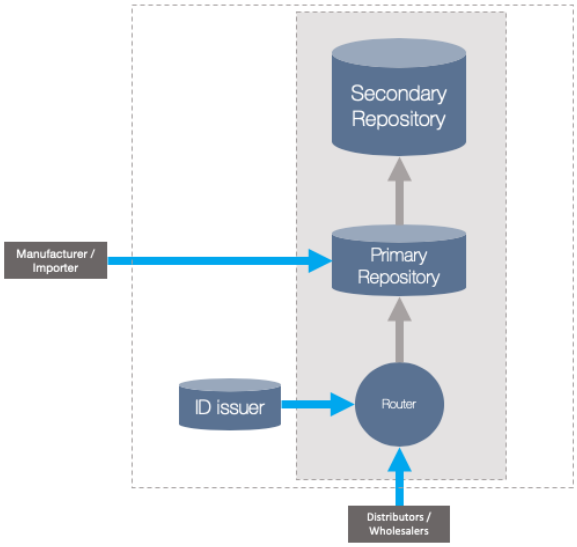


Figure 1 Validation Overview

3.3.1.2 Validation response

The Https status of the response provide information on the way the receiving system is handling the data.

- Http status **200 – 202**: the message is processed successfully
- Http status **299**: the message contains one or more errors (validation failure), the message is processed even with the errors. (some processing might be limited). The response contains the list of failed validations.
- Http status **400**: the message contains one or more validation errors. The message will NOT be processed. An audit of the message and its corresponding response will be stored on the EU Secondary. The response contains the list of failed validations.

Http status 299 is used in multiple circumstances. This warning status is used to informing the sending system of an unrecognized UI during the "Transition Period" as well as out of sequence event during the grace period. This status is also used in case a message is reported outside the 24hour (defined on the validation VAL_EVT_24H). The response message contains the list of validations failures that caused the warning.

3.3.1.3 Technical and business validation

The validation process is composed by a technical aspect and business validation. The technical validation ensures that the message follows the general format and message structure including field availability and predefined values. This validation is performed on all components.
The business validation is performed after the technical validation and focuses on the state of business data.

3.3.1.4 Message transmission overview

The following diagram describes the different message transmission and response options.

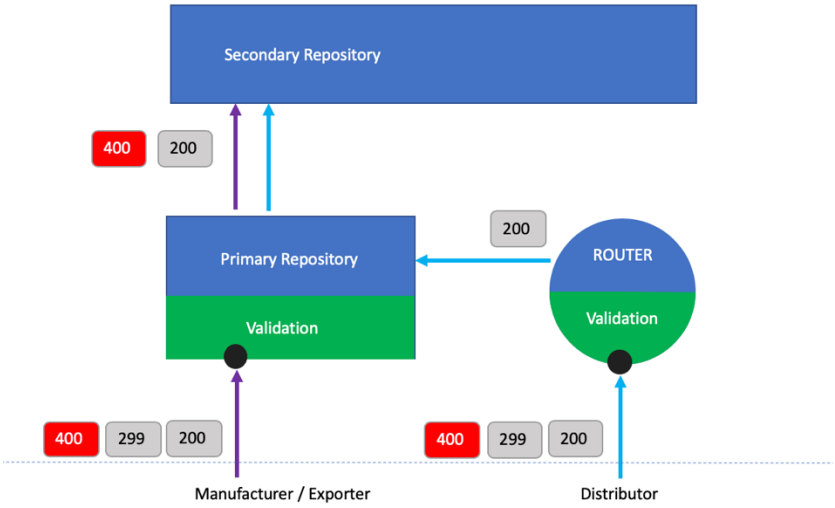


Figure 2 EU secondary validation system diagram

Note that the technical validation is not represented in the following diagram. Only the business validation. It is clear that messages that fail the technical validation (wrong format, not authorized) will be rejected by all systems with an http status 400.

3.3.1.4.1 Secondary to Primary Feedback

The Secondary repository performs the full validation (technical and business validation) all messages from the different primary repositories as described in the list of Specification.

The information provided by the primary repository is used to update the internal state of the different UI. The states are then used during the validation enforcement on the router.

The Secondary repository is running all the validation in order to confirm the coherence of the state changes.

In case the secondary validation process returns a negative response, the secondary repository will reject the message (http status 400)
The Meta data associated with the UI is not updated if the message is rejected (error 400).

If a message is rejected by the secondary repository, the primary repository can retransmit the same message (same payload and same RecallCode) as an attempt to get it accepted.

3.3.1.4.2 Router to Primary

The router performs the validation of the messages received. Depending on the type of messages, the message is then forwarded to the Primary repository. The primary repository will accept the messages and always (messages that are passing or failing the business validations) return a http 200 or 202 to the router.

3.3.1.4.3 Router and Primary to EO

The messages from the Economic operators are validated at the first point of entry. For the manufacturer and importer, the messages are transmitted to the primary repositories.
The messages from the distributors are validated by the router based on the information processed on the secondary repository.

Depending on the content of the message, the EO might receive a successful response (http 200 or 202) a warning (http 299) or a rejection data (http 400).

3.3.2 Timestamps

Control	Event Time	Message Time Long	Reception Time	Record Time
Source	Provided by EO system	Provided by EO system	Provided by Primary repository or Router	Established by the Secondary repository
Precision	hour	milliseconds	milliseconds	milliseconds
		Used for sequence validation	Used for sequence validation	Used for audit purposes
Note		Note: the Message time can be the Event Time on the millisecond precision		

3.3.2.1 Event Time & Message Time Long Technical Validation

To prevent Economic Operator to report inaccurate Event Time and Message Time Long information.
The following technical validation

- prevents dates to be reported prior to May 2019
- prevents dates in the future for more than 72h

Note the Event Time and Message Time Long are GMT timestamps.

Control	Description	Scope
VAL_TIME_2019	Prevent Event Time and message time long to be reported before May 2019	All messages
VAL_TIME_72	Prevent Event Time and Message Time long to be reported more than 72h in the future (compared with current time)	All messages

3.3.3 Type of validation

3.3.3.1 Security validation

The security validation is the first part of the technical validation ensuring that connecting systems are authenticated and authorized to transmit data to the traceability system.

Control	Description	Scope
VAL_SEC_HASH	Integrity check of the checksum	All messages
VAL_SEC_TOKEN	Oauth Security Token validation	All messages

3.3.3.2 Message Structure validation

The technical validation ensures that the messages are following the technical guidelines and allows the system to successfully access the message data accurately.

Control	Description	Scope
VAL_MSG_JSON	JSON structure validation	All JSON messages
VAL_MSG_TYPE	Message type validation	All messages
VAL_FIE_MAN	Mandatory Field validation (per message type)	All messages
VAL_FIE_FORMAT	Field format validation	All messages
VAL_FIE_REF	Existence of correctly reference enumerations. (As defined in data dictionary)	All messages

3.3.3.3 Message Transmission validation

The transmission controls are established to prevent technical duplication and processing of messages.

Control	Description	Scope
VAL_MSG_DUPLICATE	Message payload already processed successfully by the Router or Primary should be rejected.	All messages
VAL_MSG_CODE_DUPLICATE	Message identified by a Recallcode that has already been processed successfully should be rejected.	All messages

3.3.3.4 Unique Identifiers validation

The following validations are performed on each UI that is present in a message.

3.3.3.4.1 Message level validation

The UI present in a message should be present only once. If the UI is present multiple times, the message will be rejected as non-compliant.

Control	Description	Scope
VAL_UI_MULT_MSG	Multiple duplicate UI present in the messages. For EPA (message 3.2), the validation on the parent UI should also be performed in order to avoid first level cyclical reference.	*IRU - IDA – IRR - EUA – EPA – EDP – EDX - ERP- ETL- EUD- EVR – EIV – EPO - EPR*

3.3.3.4.2 Application Validation

Validation that he upUI(i) is only applied once to a upUI(L)

Control	Description	Scope
VAL_UI_EXIST_APP	upUI(i) has been received as part of an IRU message (calculated via upUI(L) minus timestamp). This validation fails if the upUI(i) is not found and has not been reported.	*EUA - IDA*
VAL_UI_HR_EXIST	upUI(s) has been received as part of an IRU message. This validation fails if the upUI(s) is not found and has not been reported	*EUA*
VAL_UI_DUPLICATE_APP	UI validity Check if the upUI(i) has already been applied to a upUI(L)	*EUA*
VAL_UI_FID_APP	Validation of the FID defined in the IRU message (2.1).	*EUA*

3.3.3.4.3 Existence

The following control ensures that the UI (upUI and aUI) comply with the regulation and could participate in product movement or transaction event reporting.

Control	Description	Scope
VAL_UI_EXIST_UPUI	UI existence upUI exists and has been successfully applied	*EPA (Children) – EDP – ERP- ETL- EVR - EIV – EPO – EPR – IDA – IRR - EDX*
VAL_UI_EXIST_AUI	aUI existence aUI has been aggregated (part of an EPA)	*IDA – EPA (Children) – EDP – ERP- ETL- EVR - EIV – EPO – EPR – EDX - IRR*
VAL_UI_EXIST_UPUI_SEQ	UI validity • upUI exists and has been successfully applied • upUI has not been part of any deactivation message.	*EPA (Children) – EDP – ERP- ETL- EVR - EDX*
VAL_UI_EXIST_AUI_SEQ	aUI validity • aUI has been aggregated (part of an EPA) • and has not been disaggregated (including implicit disaggregation) • nor deactivated.	*IDA – EPA (Children) – EDP – ERP- ETL- EVR - EDX*

3.3.3.4.4 upUI Expiration

As per the regulation, the upUI(s) that are issued by the ID Issuer and reported in the IRU messages have a limited application period.

Control	Description	Scope
VAL_UI_EXPIRY	Validation that the application or the aggregation date doesn't exceed the 6 months period after the code has been issued.	*EUA, EPA*

3.3.3.5 UI level Message sequence validation

3.3.3.5.1 Sequence Validation overview

The following tables describes the summary overview of allowed events related to a certain UI based on the last event received for that specific UI.
Note that the transactional events (4.x) are not described in the rows as there is no sequence validation implemented on these events

Legend:

- Yes (Green): the message could be accepted for that specific UI
- Yes (blue): The message is accepted, and a location validation control is applied.
- No: the message should be rejected

Control	Description	Scope
VAL_UI_ORD_SEQUENCE	The general sequence validation.	*IRU – IRA- IDA- IRR - EUA- EPA- EDP- ERP- ETL- EUD- EVR - EDX*

Message Received	PREVIOUS MESSAGE on the UI present in the received message								
	IRU 2.1	IRA 2.2	IDA 2.3	IRR 2.4	EUA 3.1	EUA 3.1 Import	EPA 3.2 parent UI	EPA 3.2 parent UI Import	EPA 3.2 Child
IRU 2.1	No	No	No	No	No	No	No	No	No
IRA 2.2	No	No	No	No	No	No	No	No	No
IDA 2.3	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes	Yes
IRR 2.4	No	No	Yes	No	No	No	No	No	No
EUA 3.1	Yes	No	No	No	No	No	No	No	No
EUA 3.1 Import	Yes	No	No	No	No	No	No	No	No
EPA 3.2 parent UI	No	Yes	No	Yes	No	No	No	No	Yes
EPA 3.2 child UI (upUI)	No	No	No	Yes	Yes	Yes	No	No	Yes
EPA 3.2 child UI (aUI)	No	No	No	Yes	No	No	Yes	Yes	Yes
EDP 3.3 Export (type 1)	No	No	No	Yes	Yes	No	Yes	No	Yes
EDP 3.3 (type 2)	No	No	No	Yes	Yes	No	Yes	No	Yes
EDP 3.3 VM (type 3)	No	No	No	Yes	Yes	No	Yes	No	Yes
EDP 3.3 VV (type 4)	No	No	No	Yes	Yes	No	Yes	No	Yes
ERP 3.4	No	No	No	No	No	Yes	No	Yes	No
ERP 3.4 (Return)	No	No	No	No	No	No	No	No	No
ETL 3.5	No	No	No	No	No	No	No	No	No
ETL 3.5 (Export)	No	No	No	No	No	No	No	No	No
EUD 3.6	No	No	No	Yes	No	No	Yes	Yes	Yes
EVR 3.7	No	No	No	No	No	No	No	No	No
EDX 3.8	No	No	No	Yes	Yes	No	Yes	No	Yes
EIV 4.1	No	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EPO 4.2	No	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EPR 4.3	No	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes

Message Received	PREVIOUS MESSAGE on the UI present in the received message											
	EDP 3.3 (type 1) Export	EDP 3.3 (type 2)	EDP 3.3 (type 3) VM	EDP 3.3 (type 4) VV	ERP 3.4	ERP 3.4 (Return)	ETL 3.5	ETL 3.5 Export	EUD 3.6	EUD 3.6 (aUI implicitly disaggregated) - reuse of aUI	EVR 3.7	EDX 3.8
IRU 2.1	No	No	No	No	No	No	No	No	No	No	No	No
IRA 2.2	No	No	No	No	No	No	No	No	No	No	No	No
IDA 2.3	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
IRR 2.4	No	No	No	No	No	No	No	No	No	No	No	No
EUA 3.1	No	No	No	No	No	No	No	No	No	No	No	No
EUA 3.1 Import	No	No	No	No	No	No	No	No	No	No	No	No
EPA 3.2 parent UI	No	No	No	No	No	No	No	No	Yes	No	No	No
EPA 3.2 child UI (upUI)	No	No	No	No	Yes	Yes	No	No	No	No	No	No
EPA 3.2 child UI (aUI)	No	No	No	No	Yes	Yes	No	No	No	No	No	No
EDP 3.3 Export (type 1)	No	No	No	No	Yes	Yes	No	No	No	No	No	No
EDP 3.3 (type 2)	No	No	No	No	Yes	Yes	No	No	No	No	No	No
EDP 3.3 VM (type 3)	No	No	No	No	Yes	Yes	No	No	No	No	No	No
EDP 3.3 VV (type 4)	No	No	No	No	Yes	Yes	No	No	No	No	No	No
ERP 3.4	No	Yes	No	No	No	No	Yes	No	No	No	No	No
ERP 3.4 (Return)	Yes	Yes	Yes	Yes	No	No	Yes	Yes	No	No	Yes	Yes
ETL 3.5	No	Yes	No	No	No	No	Yes	No	No	No	No	No
ETL 3.5 (Export)	Yes	No	No	No	No	No	No	Yes	No	No	No	No
EUD 3.6	No	No	No	No	Yes	Yes	No	No	No	Yes	No	No
EVR 3.7	No	No	No	Yes	No	No	No	No	No	No	No	No
EDX 3.8	No	No	No	No	Yes	Yes	No	No	No	No	No	No
EIV 4.1	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EPO 4.2	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
EPR 4.3	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes

The type of the Dispatch event 3.3 EDP refers to the Destination_ID1 field.

- Type 1 – Non EU dest.
- Type 2 – EU destination other than VM – fixed quantity delivery
- Type 3 – EU VM(s)
- Type 4 – EU destination other than VM – delivery with VV

The type of return of the ERP (3.4) is based on the Product_Return field

- 0 – No
- 1– The arrival is a type return

3.3.3.5.2 Application and deactivation sequence validation

Control	Description	Scope
VAL_UI_ORD_REACTIVATION	upUI(s) that has been deactivated should not allow any application event (EUA).	*EUA*
VAL_UI_ORD_DEACTIVATED	UI – presence of UI in a message after being deactivated.	*EPA – EDP – EDX - ERP- ETL- EUD- EVR – IDA*
VAL_UI_ORD_REACTIVATION_NOT_ALLOWED	Reactivation of UIs not allowed (only permitted if in preceding message type 2-3, field Deact_reason1 = 2). So the validation triggers if the UIs are NOT deactivated or if they are deactivated with Deact_reason1 != 2	*IRR*

3.3.3.5.3 Aggregation and Disaggregation Principles

Principle 1: All aggregation events are full aggregation. Once the aUI is aggregated, a subsequent aggregation event on the same parent aUI shall be rejected.

Principle 2: All disaggregation events are full disaggregation. Once the aUI is disaggregated, no movement should be reported on that aUI (unless it is re-aggregated again as a parent aUI).

Principle 3: Implicit disaggregation. Disaggregation event reporting is mandatory only when the aUI is reused in a subsequent aggregation event (as a parent aUI). The implicit disaggregation is detected when at least one child UI is reported in an aggregation or product movement. The parent aUI of this child UI will be considered as disaggregated. In the case that the child UI is part of an aggregation hierarchy, all parent aUIs will be disaggregated.
As a consequence, once the implicit disaggregation is detected, no movement should be reported on that parent aUI (unless an explicit disaggregation event is declared for the implicitly disaggregated aUI, which would allow a re-aggregation of the aUI as parent with a new aggregation event).

Principle 4: All disaggregation must be performed at a location. No Disaggregation are allowed during the transport.
Note that implicit disaggregation might be triggered while in transit by a deactivation event IDA (2.3), a delivery carried out by vending van EVR (3.7) or an arrival ERP (3.4) of type Return.

3.3.3.5.4 Implicit disaggregation

Since disaggregation events are only mandatory when the parent aUI is intended to be reused, Implicit disaggregation event will happen.

3.3.3.5.4.1 Triggers

These events will be detected / triggered when a child UI is identified on one of the following messages: IDA (2.3), EPA (3.2), EDP (3.3), ERP (3.4) of type Return, EUD (3.6), EVR (3.7) and EDX (3.8).
Note that transactional events are not triggering any implicit disaggregation.

Example of implicit disaggregation triggered by EDP (3.3)

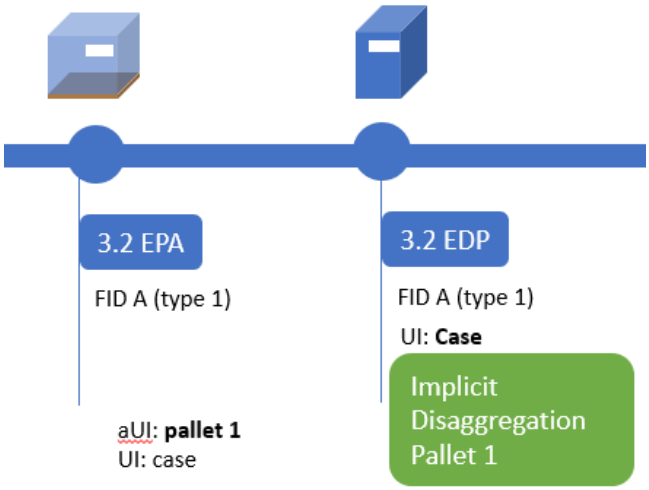


Figure 3 Implicit disaggregation triggered by EDP (3.3)

Example of implicit disaggregation triggered by EVR (3.7)

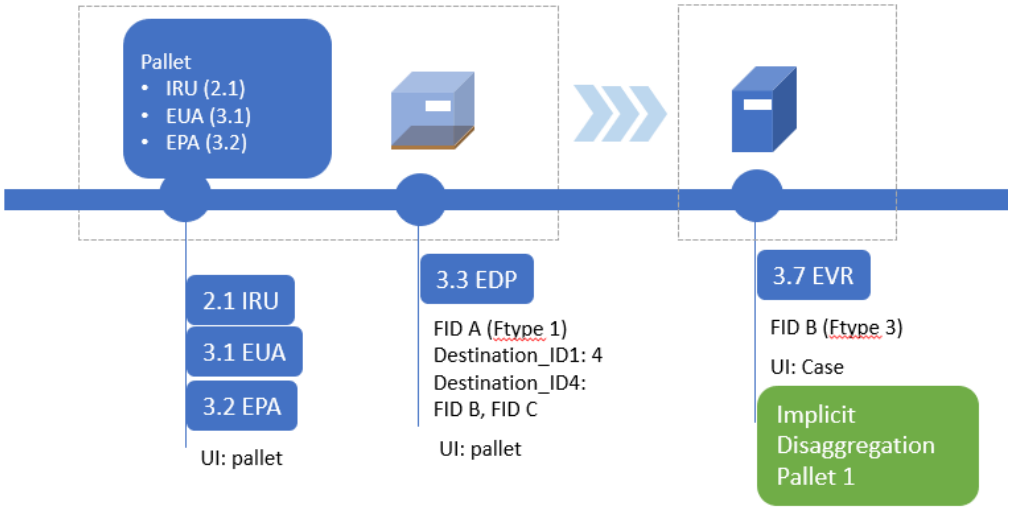


Figure 4 Implicit disaggregation triggered by EVR (3.7)

3.3.3.5.4.2 Disaggregated state

Once the UI is implicitly disaggregated, the UI should be considered disaggregated and should not be part of any subsequent product movement (3.x). The definition of implicit disaggregation is that the affected parent aUI is considered completely unlinked from all the children it had at the moment of the implicit disaggregation trigger ("Open"). If a reported UI has more than one linked parent at the moment of the reporting, all parents should be implicitly disaggregated, this includes grand-parents and recursively upwards linked UIs.

3.3.3.5.4.3 Recall

It is important to note that implicit disaggregation should be rolled back in case the event that triggered them is Recalled.

3.3.3.5.4.4 Explicit disaggregation after implicit disaggregation

The scope of the VAL_UI_ORD_IMPLDISAGG and VAL_UI_ORD_DISAGG have been removed from EUD (3.6). This means that it is possible to disaggregate an aUI after being implicitly disaggregated.

3.3.3.5.5 Aggregation and Disaggregation Validation

Control	Description	Scope
VAL_UI_ORD_AGG_MULT	Validation that a parent UI has not been part of any prior aggregation event (as parent) without being part of a disaggregation event. This control prevents the reuse of an aUI without prior disaggregation.	*EPA*
VAL_UI_ORD_DISAGG	Validation that an aUI that has been disaggregated cannot be part of any product movement prior of being aggregated as a parent.	*EDP - ERP – ETL – EVR – EDX – EPA (as children) - EUD*

VAL_UI_ORD_IMPLDISAGG	Validation that an aUI has been implicitly disaggregated cannot be part on any product movement prior of being part of an EUD message and then aggregated.	*EDP – ERP – ETL – EVR- EPA - EDX*

3.3.3.5.6 Location based Validation

3.3.3.5.6.1 UI Location update

The sequence validation also considers the implementation of location-based controls.
The FID of the UI is updated upon processing of the following events.

- ERP (3.4)
- EUA (3.1) update of the location of the UI. The location of the UI should match the location defined in the IRU message.
- EPA (3.2) update of the newly created parent id.
- EVR (3.7) In case of implicit disaggregation, only the UIs present in the event will have the location metadata updated.
- RCL (5.0) rolls back to the previous state
- IRR (2.4) Updates the location of the UIs to the F-ID where they are reactivated. This does not need to be the location where they were deactivated (IDA), can be any F-ID.

3.3.3.5.6.2 Location based controls

Control	Description	Scope
VAL_UI_ORD_AGG_FID	Validation that the aggregation and the disaggregation events must happen at the same facility (FID) where the products have been either created or arrived.	*EPA - EUD*

3.3.3.5.7 Dispatch and arrival Validation

Principle 5: The reporting on the Arrival should be done on the same UI that have been reported during the Dispatch/Transloading process. This is a consequence of principle 4. This means that an Arrival Event that contains child UI of UI reported during the Dispatch/Transloading Event will be rejected. The same UI must be reported.

Exception VAL_UI_ORD_ARRIVAL_RETURN: an arrival of type returns skips the principle 5 but still enforces the event sequence validation.

Control	Description	Scope
VAL_UI_ORD_ARRIVAL	Validation that a UI is part of a prior reported dispatch or	*ERP (Product_Return = 0)*

	transloading event (EDP 3.3, ETL 3.5). This validation concerns the sequence of events. *Exception:* Imported products	
VAL_UI_ORD_ARRIVAL_RETURN	Validation that a UI is part of a prior reported dispatch or transloading event (EDP 3.3, ETL 3.5, EVR 3.7, EDX 3.8) for the specified destination. This validation concerns the sequence of events. In this validation principle 5 doesn't apply.	*ERP (Product_Return = 1)*

3.3.3.5.7.1 Imported goods Exception

The IRU (response to message 2.1 defined in Annex II) contains the import flag information. This flag (which takes the values "true" for product that will be produced for "import into EU" and "false" for product that is produced inside EU) will be used to assess if the exception should be implemented.

a) Unit packs that contain the import flag = true, that are applied and eventually aggregated with EUA (message 3.1) and EPA (message 3.2) in a facility **outside** the EU must be reported in an **arrival message** with a facility whose country is inside the EU (the physical importation of the goods) before being part of any product movement inside the EU.
b) Unit packs that contain the import flag = false, that are applied and eventually aggregated with EUA (message 3.1) and EPA (message 3.2) in a facility whose country is **inside** the EU, must not report an arrival message.

3.3.3.5.7.2 Arrival of type return

The reporting of the arrival of type return can be performed on child UI. This operation is allowed and will trigger an implicit disaggregation.

3.3.3.5.7.3 Arrival of type return after delivery carried out via vending van (EVR)

The Router allows for Arrival Returns to be declared after product has been delivered via Vending Van (EVR – 3.7). This is to accommodate the

fact that Retail Outlets do not declare a Dispatch in case they wish to return some product to the Distributor that was acquired via Vending Van distribution. This exception only works if the F-ID declared in the EVR is of type "Retail Outlet".

3.3.3.5.7.4 Dispatch validation

Control	Description	Scope
VAL_UI_ORD_DISPATCH	Validation that a UI last location (FID) matches the source location (FID) of the dispatch event. The UI must have been: - Applied or aggregated on that specific location (FID) - Arrived on that location. - Reactivated on that location	*EDP - EDX*

3.3.3.6 Message Event Time Validation

The following messages validation compares the event time (Event Time) to the actual reception time of the event by the first point of entry.

Control	Description	Scope
VAL_EVT_24H	Validation that the Events are reported within 24 hours from the occurrence of the event. This validation is performed on the Event Time compared to the Record Time of the Primary repository or the Router. NOTE: this validation will be reduced to 3 hours after 20 May 2028 **This validation should not be blocking but rather generating a warning to the sender system**	*EUA – EPA – EVR – EIV – EPO – EPR*
VAL_EVT_TIME	"Within 24 hours prior to the occurrence of the event" rule for dispatch and trans-loading event messages is a rule and the system shall reject non-compliant messages. Control is based on the "actual date – Event Time" time difference. **This validation should not be blocking but rather generating a warning to the sender system**	*EDP – ETL - EDX*

3.3.3.7 Identification Code validation

Identification codes are used in a number of messages. The validation of the existence and fact that the identification code is active is part of the business validation as described in the table below.

Control	Description	Scope

VAL_ENT_EXIST_EOID	Check if the EOID exists in the EU wide registry	*All messages*
VAL_ENT_EXIST_FID	Check if FID, exists in the EU wide registry	*All messages*
VAL_ENT_EXIST_MID	Check if MID, exists in the EU wide registry	*ISU - IRU*
VAL_ENT_ACTIVE_EOID	Check if EOID is marked as active in the repository (Router only)	*IRU – IRA – IDA – IRR - EUA – EPA – EDP – ERP – ETL – EUD- EVR - EDX*
VAL_ENT_ACTIVE_FID	Check if FID is marked as active in the repository (Router only)	*ISU – ISA – IRU – IRA - IRR – EUA – EPA - EDX*
VAL_ENT_ACTIVE_MID	Check if MID is marked as active in the repository (ID Issuer / Router)	*ISU – IRU*
VAL_ENT_REL_EOID_FID	Check if EOID FID relation is defined in the EU wide registry	*IRU – IRA*
VAL_ENT_REL_ FID_MID	Check if FID MID relation is defined in the EU wide registry	*IRU – IRA*
VAL_ENT_MPART	Check that the codes are not ordered for a Machine ID that is a Machine Part	*ISU - IRU*

3.3.3.8 Recall Validation

3.3.3.8.1 General recall rules

The sequence validation on the product movement introduces additional controls on the recall process. In order to maintain the consistency of the history of the UI, only the recall of the last event for each UI will be authorized.
If a message to be recalled, contains a UI (any in the reported list) that has a subsequent event, the subsequent event must be recalled first.

For the sake of clarity, the following scenarios describe the process of recall. An EO report two product movements on a UI. (Event 1 and Event 2). If the EO wishes to recall Event 1, the EO has first to recall Event 2 and only after recall Event 1. Moreover, Event 2 must be the last event occurred on all UIs contained in Event 2 for Event 2 could be recalled.

3.3.3.8.2 Transaction events

Transaction events (4.x) are not subject to this rule as they are not impacted by the sequence validation control and therefore, transaction events (4.x) can be recalled at any time.

3.3.3.8.3 Recall Validation Controls

Control	**Description**	**Scope**
VAL_RECALL_EXIST	Check if RecallCode exists	*RCL*
VAL_RECALL_LAST	Check if for all UIs related to the event identified by RecallCode is the very last unrecalled event occurred on all	*RCL*

	such UI including related implicitly disaggregated parents.	
VAL_RECALL_EXPIRED	Check that message 2.3's reception time is less than 24h from when receiving the Recall message	*RCL*

3.3.4 Validation Scope

	IRU (2.1)	IRA (2.2)	IDA (2.3)	IRR (2.4)	EUA (3.1)	EPA (3.2)	EDP (3.3)	ERP (3.4)	ETL (3.5)	EUD (3.6)	EVR (3.7)	EDX (3.8)	EIV (4.1)	EPO (4.2)	EPR (4.3)	RCL (5)
Technical validation																
VAL_SEC_HASH	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
VAL_SEC_TOKEN	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
VAL_MSG_JSON	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
VAL_MSG_TYPE	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
VAL_FIE_MAN	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
VAL_FIE_FORMAT	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
VAL_FIE_REF	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
VAL_MSG_DUPLICATE	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
VAL_MSG_CODE_DUPLICATE	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
VAL_UI_MULT_MSG	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	
VAL_EOID_SELLER													X			
VAL_EOID_PAYMENT_RECIPIENT															X	
Business rule validation																
UI creation																
VAL_UI_EXIST_APP			X		X											
VAL_UI_HR_EXIST					X											
VAL_UI_DUPLICATE_APP					X											
VAL_UI_EXIST_UPUI				X		X	X	X	X		X	X	X	X	X	
VAL_UI_EXIST_AUI				X		X	X	X	X	X	X	X	X	X	X	
VAL_UI_EXIST_UPUI_SEQ			X			X	X	X	X		X	X				
VAL_UI_EXIST_AUI_SEQ			X			X	X	X	X	X	X	X				
VAL_UI_EXPIRY					X	X										
Entity Validation																
VAL_ENT_EXIST_EOID	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
VAL_ENT_EXIST_FID	X	X		X	X	X	X	X	X	X	X	X				
VAL_ENT_EXIST_MID	X															
VAL_ENT_ACTIVE_EOID	R	R	R	R	R	R	R	R	R	R	R	R				
VAL_ENT_ACTIVE_FID	R	R	R	R	R	R	R	R	R	R	R	R				
VAL_ENT_ACTIVE_MID	R															
VAL_ENT_REL_EOID_FID	R	R														
VAL_ENT_REL_ FID_MID	R	R														
VAL_ENT_MPART	R															
Sequence Validation																
VAL_UI_FID_APP					X											
VAL_UI_ORD_REACTIVATION					X											
VAL_UI_ORD_REACTIVATION _NOT_ALLLOWED				X												
VAL_UI_ORD_DEACTIVATED			X			X	X	X	X	X	X	X				
VAL_UI_ORD_AGG_MULT						X										
VAL_UI_ORD_DISAGG							X	X	X		X	X				
VAL_UI_ORD_IMPLDISAGG						X	X	X	X		X	X				
VAL_UI_ORD_AGG_FID						X				X						
VAL_UI_ORD_ARRIVAL								X								
VAL_UI_ORD_ARRIVAL_RETURN								X								

VAL_UI_ORD_DISPATCH							X					X				
VAL_UI_ORD_SEQUENCE			X	X	X	X	X	X	X	X	X	X				
Message Timing																
VAL_EVT_24H					X	X					X		X	X	X	
VAL_EVT_TIME							X		X			X				
VAL_TIME_2019	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	
VAL_TIME_72	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	
Recall																
VAL_RECALL_EXIST																X
VAL_RECALL_LAST																X
VAL_RECALL_EXPIRED																X

X – validation on Primary, Secondary and Router
R – Validation on Router

3.3.5 Validation Responsibility

	Primary Repository Error http status for EO	Router Error http status for EO	Secondary Repository Error http status for Primary
Technical validation			
VAL_SEC_HASH	400	400	400
VAL_SEC_TOKEN	401	401	401
VAL_MSG_JSON	400	400	400
VAL_MSG_XML	400	400	400
VAL_MSG_TYPE	400	400	400
VAL_FIE_MAN	400	400	400
VAL_FIE_FORMAT	400	400	400
VAL_FIE_REF	400	400	400
VAL_MSG_DUPLICATE	400	400	400
VAL_MSG_CODE_DUPLICATE	400		400
VAL_EOID_SELLER	400	400	400
VAL_EOID_PAYMENT_RECIPIENT	400	400	400
Business rule validation			
VAL_UI_MULT_MSG	400	400	400
VAL_UI_EXIST_APP	400		400
VAL_UI_HR_EXIST	400		400
VAL_UI_DUPLICATE_APP	400		400
VAL_UI_FID_APP	400		400
VAL_UI_EXIST_UPUI	400	400	400
VAL_UI_EXIST_AUI	400	400	400
VAL_UI_EXIST_UPUI_SEQ	400	400	400
VAL_UI_EXIST_AUI_SEQ	400	400	400
VAL_UI_EXPIRY	400		400
VAL_UI_ORD_REACTIVATION	400	400	400
VAL_UI_ORD_REACTIVATION _NOT_ALLOWED	400	400	400
VAL_UI_ORD_DEACTIVATED	400	400	400
VAL_UI_ORD_AGG_MULT	400	400	400
VAL_UI_ORD_DISAGG	400	400	400
VAL_UI_ORD_IMPLDISAGG	400	400	400
VAL_UI_ORD_AGG_FID	400	400	400
VAL_UI_ORD_ARRIVAL	400	400	400
VAL_UI_ORD_ARRIVAL_RETURN	400	400	400

VAL_UI_ORD_DISPATCH	400	400	400
VAL_UI_ORD_SEQUENCE	400	400	400
VAL_EVT_24H	299	299	
VAL_EVT_TIME	299	299	
VAL_TIME_2019	400	400	400
VAL_TIME_72	400	400	400
VAL_ENT_EXIST_EOID	400	400	400
VAL_ENT_EXIST_FID	400	400	400
VAL_ENT_EXIST_MID	400	400	400
VAL_ENT_ACTIVE_EOID	400	400	400
VAL_ENT_ACTIVE_FID	400	400	400
VAL_ENT_ACTIVE_MID	400	400	400
VAL_ENT_REL_EOID_FID		400	400
VAL_ENT_REL_ FID_MID		400	400
VAL_ENT_MPART		400	
VAL_RECALL_EXIST	400	400	400
VAL_RECALL_LAST	400	400	400
VAL_RECALL_EXPIRED	400	400	299

3.3.6 Validation and Error Code

	Error Code	**http Status**
Technical validation		
VAL_SEC_HASH	INVALID_SIGNATURE	400
VAL_SEC_TOKEN	INVALID_OR_EXPIRED_TOKEN	401
VAL_MSG_JSON	INVALID_INPUT_FORMAT REQUIRED_FIELD_FAILED_VALIDATION MAX_LENGTH_FAILED_VALIDATION MIN_LENGTH_FAILED_VALIDATION ENTRY_LENGTH_FAILED_VALIDATION EXCISE_NUMBER_NOT_VALID NON_COMPATIBLE_UIS NOT_THE_SAME_NUMBER_OF_ITEMS	400
VAL_MSG_XML	FAILED_VALIDATION	400
VAL_MSG_TYPE	FAILED_VALIDATION	400
VAL_FIE_MAN	FAILED_VALIDATION	400
VAL_FIE_FORMAT	INVALID_INPUT_FORMAT	400
VAL_FIE_REF	FAILED_VALIDATION	400
VAL_MSG_DUPLICATE	PAYLOAD_NOT_UNIQUE	400
VAL_MSG_CODE_DUPLICATE	FAILED_VALIDATION	400
VAL_UI_MULT_MSG	MULTIPLE_UID UI_NOT_VALID UIS_NOT_VALID	400
VAL_EOID_SELLER	FAILED_VALIDATION	400
VAL_EOID_PAYMENT_RECIPIENT	FAILED_VALIDATION	400
Business rule validation		
UI creation		
VAL_UI_EXIST_APP	UIS_APPLICATION_ERROR	400
VAL_UI_HR_EXIST	UIS_APPLICATION_ERROR	400
VAL_UI_DUPLICATE_APP	UIS_APPLICATION_ERROR	400
VAL_UI_EXIST_UPUI	UI_NOT_EXIST UI_NOT_VALID	400

VAL_UI_EXIST_AUI	UI_NOT_EXIST	400
VAL_UI_EXIST_UPUI_SEQ	UI_NOT_VALID	400
VAL_UI_EXIST_AUI_SEQ	UI_NOT_EXIST	400
VAL_UI_EXPIRY	UI_EXPIRED	400
Entity Validation		
VAL_ENT_EXIST_EOID	EOID_NOT_EXIST_OR_ACTIVE	400
VAL_ENT_EXIST_FID	FID_NOT_EXIST_OR_ACTIVE	400
VAL_ENT_EXIST_MID	MID_NOT_EXIST_OR_ACTIVE	400
VAL_ENT_ACTIVE_EOID	EOID_NOT_EXIST_OR_ACTIVE	400
VAL_ENT_ACTIVE_FID	FID_NOT_EXIST_OR_ACTIVE	400
VAL_ENT_ACTIVE_MID	MID_NOT_EXIST_OR_ACTIVE	400
VAL_ENT_REL_EOID_FID	FID_NOT_RELATED_TO_EOID	400
VAL_ENT_REL_ FID_MID	MID_NOT_RELATED_TO_FID	400
VAL_ENT_MPART	MID_MACHINE_PART	400
Sequence Validation		
VAL_UI_FID_APP	FID_MISMATCH	400
VAL_UI_ORD_REACTIVATION	UI_DEACTIVATED	400
VAL_UI_ORD_REACTIVATION _NOT_ALLOWED	FAILED_VALIDATION	400
VAL_UI_ORD_DEACTIVATED	UI_DEACTIVATED	400
VAL_UI_ORD_AGG_MULT	MULTIPLE_AGGREGATION	400
VAL_UI_ORD_DISAGG	UI_ALREADY_DISAGGREGATED	400
VAL_UI_ORD_IMPLDISAGG	UI_ALREADY_DISAGGREGATED	400
VAL_UI_ORD_AGG_FID	LOCATION_MISMATCH	400
VAL_UI_ORD_ARRIVAL	ARRIVAL_NOTALLOWED	400
VAL_UI_ORD_ARRIVAL_RETURN	ARRIVAL_NOTALLOWED	400
VAL_UI_ORD_DISPATCH	LOCATION_MISMATCH	400
VAL_UI_ORD_SEQUENCE	UI_SEQUENCE_ERROR	400
Message Timing		
VAL_EVT_24H	OPERATION_WITHIN_24_HOURS	299
VAL_EVT_TIME	SHIPMENT_WITHIN_24_HOURS	299
VAL_TIME_2019	OPERATION_PRIOR_MAY_2019	400
VAL_TIME_72	OPERATION_AFTER_72H	400
Recall		
VAL_RECALL_EXIST	CODE_NOT_EXIST CODE_NOT_UNIQUE	400
VAL_RECALL_LAST	RECALL_NOT_LAST_EVENT	400
VAL_RECALL_EXPIRED	RECALL_EXPIRED	400

3.3.7 Secondary repository special processing of technical historical data.

3.3.7.1 Context

In some exceptional cases, if the primary repositories fail to report some events, the secondary validation will prevent the primary to report these events a posteriori.

The following edge cases have been identified.

- MAINT_01: Repacking scenario
- MAINT_02: Historical Transloading scenario
- MAINT_03: Arrival before Deactivation

In order to allow the secondary to be a copy of the primary repositories, the secondary repository will perform a limited update of the meta data. Only the event list will be updated
The metadata used for the validation (state and location) will not be updated in these specific cases.

The secondary will add specific audit trail during the processing of these specific cases.
The secondary will add specific information to the event allowing the competent authority to be informed of this specific processing.

Note that any other cases not exactly matching these scenarios will be rejected. (http status 400)

The primary repository will receive a successful answer (http status 202) and will not have to resend the message.

3.3.7.2 MAINT_01: Repacking scenario

The initial packing process for a certain aUI corresponds to the reporting of an Aggregation (EPA 3.2) event.
The repacking process consists of an explicit Disaggregation (EUD 3.6) event followed by a new Aggregation (EPA 3.2) event.
In unlikely event of an issue in the reporting of the original aggregation and disaggregation event, the final aggregation event will be processed by the secondary repository.

MSG Sequence	MSG Type	UIs	Primary ACK	Secondary ACK	Comments
1	EPA1 (3.2)	aUI	202	TIMEOUT	Message is not transmitted to Secondary due to technical reasons
2	EUD (3.6)	Same aUI	202	TIMEOUT	Message is not transmitted to Secondary due to technical reasons
3	EPA 2 (3.2)	Same aUI	202	202	

Message supported:

- aggregation message (EPA),
- disaggregation message (EUD).

Controls

- Messages must have been rejected by the regular endpoint.
- Control on the Existence of the aUI
- Control on the Reception Time or Message Time Long

3.3.7.3 MAINT_02: Historical Transloading scenario

ETL message

MSG Sequence	MSG Type	UIs	Primary ACK	Secondary ACK	Comments
1	EDP	Any	202	202	
2	ETL	Same as in EDP	202	TIMEOUT	Message is not transmitted to Secondary due to technical reasons
3	ERP	Same as in EDP	202	202	

Message supported:

- Transloading message (ETL 3.5),

Controls.

- Control on the MessageTimeLong that ETL was sent after an EDP (3.3)
- Control on the MessageTimeLong that ETL was sent before an ERP (3.4)

3.3.7.4 MAINT_03: Arrival before Deactivation

MSG Sequence	MSG Type	UIs	Primary ACK	Secondary ACK	Comments
1	EDP (3.3)	Any	202	202	
2	ERP (3.4)		202	TIMEOUT	Message is not transmitted to Secondary due to technical reasons
3	IDA (2.3)		202	202	

Message supported:

- Arrival message (ERP 3.4)

3.3.7.5 Recall clarification

Once the events are accepted, including events accepted with special processing (meaning that events have been accepted out of sequence), the recall validation will apply on the full event sequence.
By definition the events accepted with special processing are NOT the latest events for some UI present in the event. For that reason, the Recall process must first be applied on the latest event (based on the Reception_Time).
In the case of MAINT_02 where the ETL (3.5) transloading event has been reported after the ERP (3.4) Arrival event, the Recall must first by reported on the ERP and then another recall on the ETL.

MSG Sequence	MSG Type	UIs
1	EDP	Any
2	ETL	Same as in EDP
3	ERP	Same as in EDP

3.4 Identifier codes for economic operators, facilities and machines messages

3.4.1 REO - (1.1) Registration of an Economic operator

3.4.1.1 Description

Submit the information for the first registration of the economic operator.

3.4.1.2 Description of the fields

registration of economic operator – request					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Req	Block of basic information elements	Component << Basic Information Request >>	S	M	Message_Type = REO
EO_Name1	Economic operator's registered name	Text(100)	S	M	
EO_Name2	Economic operator's alternative or abridged name	Text(100)	S	O	
EO_street	Economic operator’s street name and house number (or road number and kilometer)	Text(300)	S	M	
EO_municipality	Economic operator’s municipality (city, town or village)	Text(100)	S	M	
EO_postcode	Economic operator’s postal code	Text(50)	S	M	‘n/a’ is permitted value if no postal code has been assigned
EO_A_info	Additional information on economic operator’s address (e.g. location in the shopping mall or industrial area)	Text(100)	S	O	
EO_CountryReg	Economic operator's country of registration	Country	S	M	See Country

EO_Email	Economic operator's email address; used to inform about registration process, incl. subsequent changes and other required correspondence	Text(80) (Regex protected)	S	M	
VAT_R	Indication of the VAT registration status	Boolean	S	M	0 – No VAT registration 1 – VAT number exists
VAT_N	Economic operator's VAT number	Text(20)	S	M, if VAT_R = 1	
TAX_N	Economic operator's tax registration number	Text(20)	S	M, if VAT_R = 0	
EO_ExciseNumber1	Indication if the economic operator has an excise number issued by the competent authority for the purpose of identification of persons/premises	Boolean	S	M	0 – No SEED number 1 – SEED number exists
EO_ExciseNumber2	Economic operator's excise number issued by the competent authority for the purpose of identification of persons/premises	SEED	S	M, if EO_ExciseNumber1 = 1	
OtherEOID_R	Indication if the economic operator has been allocated an identifier by another ID Issuer	Boolean	S	M	0 – No 1 – Yes
OtherEOID_N	Economic operator identifier codes allocated by other ID Issuers	EOID	M	M, if OtherEOID_R = 1	
Reg_3RD	Indication if the registration is made on behalf of a retail outlet operator not otherwise	Boolean	S	M	0 – No 1 – Yes

	involved in the tobacco trade				
Reg_EOID	Identifier of the economic operator that acts on behalf of a retail outlet operator not otherwise involved in the tobacco trade	EOID	S	M, if Reg_3RD = 1	
EO_OtherID	Optional identifier	Text(50)	S	O	
Extensibility	Optional extensibility field	Text(5000)	S	O	

3.4.1.3 Response

Field	Description	Data Type	Cardinality	Priority	Values
BasicInfo_Resp	Block of basic information elements	Component << Basic Information Response >>	S	M	Message_Type = REO
EO_ID	Economic operator's registered ID	EOID	S	O	
EO_CODE	Economic operator's confirmation code provided in response to the registration of economic operator	EO_CODE	S	O	
Extensibility	Optional extensibility field	Text(5000)	S	O	

3.4.1.4 Request sample

```
{
  "EO_Name1": "Example Legal Entity",
  "EO_Name2": "",
  "EO_street": "59 Legal Street",
  "EO_municipality": "myCity",
  "EO_postcode": "123123",
  "EO_CountryReg": "DE" ,
  "EO_Email": "email@test.com",
  "VAT_R": 1,
  "VAT_N": "VATNumber 1",
  "TAX_N": "Tax",
  "EO_ExciseNumber1":1,
  "EO_ExciseNumber2": "LA111FD",
  "OtherEOID_R": false,
  "OtherEOID_N": [],
  "Reg_3RD": false,
  "Reg_EOID": "",
  "EO_OtherID ": "GLNSAMPLE",
  "Message_Type": "REO",
  "Code": "873345b2-882f-4064-91f0-90669b46c30a",
  "EO_OtherID": "XFG6GN5J5JG98VJKFHJKKJ"
}
```

3.4.1.5 Successful response sample

HTTP Status 202


```
{
   "Code": "873345b2-882f-4064-91f0-90669b46c30a",
    "EO_ID": "SAMPLEEOID1",
    "EO_CODE": "873345b2-882f-4064-91f0-90669b46c30a",
    "Message_Type": "REO",
    "Error": false,
    "Errors": null,
    "Checksum": "DFG65H"
}
```


3.4.1.6 Error response sample

Processing errors

HTTP status		
<< Common response code >>		
400	ALREADY_EXISTS	Indicated that the CRUD action in add a new entity failed, as the item already exist. This is when checking of the item id already exists.

3.4.2 CEO – (1.2) Correction for an economic operator identifier code

3.4.2.1 Description

Submit the information of an economic operator known to the repository in order to update 1 or more properties. This information in entirety will over write the previous data held regarding the master data of this economic operator. Links (for example dispatches) to / from this EO_ID will be maintained.

3.4.2.2 Description of the fields

Field	Description	Data Type	Cardinality	Priority	Values
BasicInfo_Req	Block of basic information elements	Component << Basic Information Request >>	S	M	Message_Type = CEO
EO_ID	Economic operator identifier code	EOID	S	M	

EO_CODE	Economic operator's confirmation code provided in response to the registration of economic operator	EO_CODE	S	M	
EO_Name1	Economic operator's registered name	Text(100)	S	M	
EO_Name2	Economic operator's alternative or abridged name	Text(100)	S	O	
EO_street	Economic operator's street name and house number (or road number and kilometer)	Text(300)	S	M	
EO_municipality	Economic operator's municipality (city, town or village)	Text(100)	S	M	
EO_postcode	Economic operator's postal code	Text(50)	S	M	'n/a' is permitted value if no postal code has been assigned
EO_A_info	Additional information on economic operator's address (e.g. location in the shopping mall or industrial area)	Text(100)	S	O	
EO_CountryReg	Economic operator's country of registration	Country	S	M	See Country
EO_Email	Economic operator's email address – used to inform about registration process, incl. subsequent changes	Text(80) (Regex protected)	S	M	
VAT_R	Indication of the VAT registration status	Boolean	S	M	0 – No VAT registration 1 – VAT number exists
VAT_N	Economic operator's VAT number	Text(20)	S	M, if VAT_R = 1	
TAX_N	Economic operator's tax registration number	Text(20)	S	M, if VAT_R = 0	

EO_ExciseNumber1	Indication if the economic operator has an excise number issued by the competent authority for the purpose of identification of persons/premises	Boolean	S	M	0 – No SEED number 1 – SEED number exists
EO_ExciseNumber2	Economic operator's excise number issued by the competent authority for the purpose of identification of persons/premises	SEED	S	M, if EO_ExciseNumber 1 = 1	
OtherEOID_R	Indication if the economic operator has been allocated an identifier by another ID Issuer	Boolean	S	M	0 – No 1 – Yes
OtherEOID_N	Economic operator identifier codes allocated by other ID Issuers	EOID	M	M, if OtherEOID_R = 1	
Reg_3RD	Indication if the registration is made on behalf of a retail outlet operator not otherwise involved in the tobacco trade	Boolean	S	M	0 – No 1 – Yes
Reg_EOID	Identifier of the economic operator that acts on behalf of a retail outlet operator not otherwise involved in the tobacco trade	EOID	S	M, if Reg_3RD = 1	
Extensibility	Optional extensibility field	Text(5000)	S	O	

3.4.2.3 Response

correction of information concerning the economic operator – resp ɔnse					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Resp	Block of basic information elements	Component << Basic Information Response >>	S	M	Message_Type = CEO
Extensibility	Optional extensibility field	Text(5000)	S	O	

3.4.2.4 Request sample

```
{
  "Message_Type": "CEO",
  "Code": "873345b2-882f-4064-91f0-90669b46c30a",
  "EO_ID": "QCUKR+1AB020054",
  "EO_CODE": "873345b2-882f-4064-91f0-90669b46c30a",
  "EO_Name1": "registerationname",
  "EO_Name2": "",
  "EO_street": "address 1",
  "EO_municipality": "myNewCity",
  "EO_postcode": "321321",
  "EO_CountryReg": 27,
  "EO_Email": "email@test.com",
  "VAT_R": 1,
  "VAT_N": "VATNumber 1",
  "TAX_N": "Tax",
  "EO_ExciseNumber1": 1,
  "EO_ExciseNumber2": "LA111FD",
  "OtherEOID_R": false,
  "OtherEOID_N": [ "" ],
  "Reg_3RD": false,
  "Reg_EOID": ""
}
```


3.4.2.5 Successful response sample

HTTP Status 202


```
{
    "Code": "873345b2-882f-4064-91f0-90669b46c30a",
    "Message_Type": "CEO",
    "Error": false,
    "Errors": null,
    "Checksum": "G6HF5H"
}
```


3.4.2.6 Error response sample

Processing errors

HTTP status		
<< Common response code >>		

3.4.3 DEO – (1.3) De-registration of economic operator identifier code.

3.4.3.1 Description

De-registers a previously known operator identifier for a given EO_ID.

The de-registration of an economic operator identifier code shall lead to the automatic de-registration of related facility identifier codes and machine identifier codes by the ID issuer including fixed and mobile parts.

3.4.3.2 Description of the fields

De-registration of economic operator – request					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Req	Block of basic information elements	Component << Basic Information Request >>	S	M	Message_Type = DEO
EO_ID	Economic operator identifier code	EOID	S	M	
EO_CODE	Economic operator's confirmation code provided in response to the registration of economic operator	EO_CODE	S	M	
Reg_3RD	Indication if the registration is made on behalf of a retail outlet operator not otherwise involved in the tobacco trade	Boolean	S	M	0 – No 1 – Yes
Reg_EOID	Identifier of the economic operator that acts on behalf of a retail outlet operator not otherwise involved in the tobacco trade	EOID	S	M, if Reg_3RD = 1	
Extensibility	Optional extensibility field	Text(5000)	S	O	

3.4.3.3 Response

correction of information concerning the economic operator – resp ɔnse					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Resp	Block of basic information elements	Component << Basic Information Response >>	S	M	Message_Type = DEO
Extensibility	Optional extensibility field	Text(5000)	S	O	

3.4.3.4 Request sample

```
{
    "EO_ID": "QCUKR+1AB020054",
    "EO_CODE": "873345b2-882f-4064-91f0-90669b46c30a ",
    "Reg_3RD": false,
    "Reg_EOID": "Machine Id A",
```

```
    "Code": "873345b2-882f-4064-91f0-90669b46c30a",
    "Message_Type": "DEO"
}
```

3.4.3.5 Successful response sample

HTTP Status 202

```
{
    "Code": "873345b2-882f-4064-91f0-90669b46c30a",
    "Message_Type": "DEO",
    "Error": false,
    "Errors": null,
    "Checksum": "G6HF5H"
}
```

3.4.3.6 Error response sample

Processing errors

HTTP status		
<< Common response code >>		

3.4.4 RFA – (1.4) Request for a facility identifier code

3.4.4.1 Description

Add a previously unsent / registered facility. Defined as unseen by the existence of the facility id in the repository.

In case that the facility was acquired from another operator and that had a facility identifier code, this must be reported and recorded by the ID Issuer by means of the new fields "PrevFID_B" and "PrevFID_ID".

3.4.4.2 Description of the fields

Request:

Registration of facility – request					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Req	Block of basic information elements	Component << Basic Information Request >>	S	M	Message_Type = RFA

EO_ID	Economic operator identifier code	EOID	S	M	
EO_CODE	Economic operator's confirmation code provided in response to the registration of economic operator	EO_CODE	S	M	
F_street	Facility's street name and house number (or road number and kilometer)	Text(300)	S	M	
F_municipality	Facility's municipality (city, town or village)	Text(100)	S	M	
F_postcode	Facility's postal code	Text(50)	S	M	'n/a' is permitted value if no postal code has been assigned
F_A_info	Additional information on facility's address (e.g. location in the shopping mall or industrial area)	Text(100)	S	S	
F_Country	Facility's country	Country	S	M	See Country
F_Type	Type of facility	Integer	S	M	See FacilityType
F_Type_Other	Description of other facility type	Text(5000)	S	M, if F_Type = 4	
F_Status	Indication if a part of the facility has a bonded warehouse status	Boolean	S	M	0 – No 1 – Yes
F_ExciseNumber1	Indication if the facility has an excise number issued by the competent authority for the purpose of identification of persons/premises	Boolean	S	M	0 – No SEED number 1 – SEED number exists
F_ExciseNumber2	Facility's excise number issued by the competent authority for the purpose of identification of persons/premises	SEED	S	M, if F_ExciseNumber1 = 1	

OtherFID_R	Indication if the facility has been allocated an identifier by another ID Issuer	Boolean	S	M	0 – No 1 – Yes (possible only for non-EU facilities)
OtherFID_N	Facility identifier codes allocated by other ID Issuers	FID	M	M, if OtherFID_R = 1	List of FIDs
PrevFID_B	Indication if the facility was acquired from another operator and had already a facility identifier code	Boolean	S	M	0 – No (first time registration) 1 – Yes
PrevFID_ID	Previous facility identifier used by the former operator of the facility	FID	S	M, if PrevFID_B = 1	
Reg_3RD	Indication if the registration is made on behalf of a retail outlet operator not otherwise involved in the tobacco trade	Boolean	S	M	0 – No 1 – Yes (possible only if F_Type = 3)
Reg_EOID	Identifier of the economic operator that acts on behalf of a retail outlet operator not otherwise involved in the tobacco trade	EOID	S	M, if Reg_3RD = 1	
Extensibility	Optional extensibility field	Text(5000)	S	O	

3.4.4.3 Response

registration of facility – response					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Resp	Block of basic information elements	Component << Basic Information Response >>	S	M	Message_Type = RFA
F_ID	Facility's identifier registered	FID	S	O	Present if synchronous implementation
Extensibility	Optional extensibility field	Text(5000)	S	O	

3.4.4.4 Request sample

```
{
"EO_ID":"QCUKR+1AB020054",
"EO_CODE":"873345b2-882f-4064-91f0-90669b46c30a",
"F_street”: street sample 123",
"F_municipality": "myCity",
"F_postcode": "444555",
"F_Country":"ES",
"F_Type":2,
"F_Status": false,
"F_ExciseNumber1": false,
"OtherFID_R": false,
"PrevFID_B": true,
"PrevFID_ID": "XXXXX4444555RRT",
"Reg_3RD": false,
"Message_Type":"RFA"
}
```

3.4.4.5 Successful response sample

HTTP Status 202

```
{
    "Code": "873345b2-882f-4064-91f0-90669b46c30a",
    "Message_Type": "RFA",
    “FID”:”SAMPLEFID123”,
    "Error": false,
    "Errors": null,
    "Checksum": "G6HF5H"
}
```

3.4.4.6 Error response sample

Processing errors

HTTP status		
<< Common response code >>		

3.4.5 CFA – (1.5) Correction of information concerning the facility identifier code

3.4.5.1 Description

Submit the information of a facility known to the repository in order to update one or more properties. This information in entirety will overwrite the previous data held regarding the master data of this facility. Links (for example dispatches) to / from this F_ID will be maintained.

3.4.5.2 Description of the fields

correction of information concerning the facility – request					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Req	Block of basic information elements	Component << Basic Information Request >>	S	M	Message_Type = CFA
EO_ID	Economic operator identifier code	EOID	S	M	
EO_CODE	Economic operator's confirmation code provided in response to the registration of economic operator	EO_CODE	S	M	
F_ID	Facility identifier code	FID	S	M	
F_street	Facility's street name and house number (or road number and kilometer)	Text(300)	S	M	
F_municipality	Facility's municipality (city, town or village)	Text(100)	S	M	
F_postcode	Facility's postal code	Text(50)	S	M	'n/a' is permitted value if no postal code has been assigned
F_A_info	Additional information on facility's address (e.g. location in the shopping mall or industrial area)	Text(100)	S	S	
F_Country	Facility's country	Country	S	M	See Country
F_Type	Type of facility	Integer	S	M	See FacilityType
F_Type_Other	Description of other facility type	Text(5000)	S	M, if F_Type = 4	
F_Status	Indication if a part of the facility has a bonded warehouse status	Boolean	S	M	0 – No 1 – Yes

F_ExciseNumber 1	Indication if the facility has an excise number issued by the competent authority for the purpose of identification of persons/premises	Boolean	S	M	0 – No SEED number 1 – SEED number exists
F_ExciseNumber 2	Facility's excise number issued by the competent authority for the purpose of identification of persons/premises	SEED	S	M, if F_ExciseNumber 1 = 1	
OtherFID_R	Indication if the facility has been allocated an identifier by another ID Issuer	Boolean	S	M	0 – No 1 – Yes (possible only for non-EU facilities)
OtherFID_N	Facility identifier codes allocated by other ID Issuers	FID	M	M, if OtherFID _R = 1	List of FIDs
PrevFID_B	Indication if the facility was acquired from another operator and had already a facility identifier code	Boolean	S	M	0 – No (first time registration) 1 – Yes
PrevFID_ID	Previous facility identifier used by the former operator of the facility	FID	S	M, if PrevFID_B = 1	
Reg_3RD	Indication if the registration is made on behalf of a retail outlet operator not otherwise involved in the tobacco trade	Boolean	S	M	0 – No 1 – Yes (possible only if F_Type = 3)
Reg_EOID	Identifier of the economic operator that acts on behalf of a retail outlet operator not otherwise involved in the tobacco trade	EOID	S	M, if Reg_3RD = 1	
Extensibility	Optional extensibility field	Text(5000)	S	O	

3.4.5.3 Response

correction of information concerning the facility – response					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Resp	Block of basic information elements	Component << Basic Information Response >>	S	M	Message_Type = CFA
Extensibility	Optional extensibility field	Text(5000)	S	O	

3.4.5.4 Request sample

```
{
"EO_ID":"QCUKR+1AB020054",
"EO_CODE":"873345b2-882f-4064-91f0-90669b46c30a ",
"F_ID":"QCUKR<1AB020054000048",
"F_street": "Sample Street 111",
"F_municipality": "City",
"F_postcode": "555777",
"F_Country":"FR",
"F_Type":1,
"F_Status": false,
"F_ExciseNumber1": false,
"OtherFID_R": false,
"PrevFID_B": false,
"Reg_3RD": false,
"Message_Type":"CFA"
}
```

3.4.5.5 Successful response sample

HTTP Status 202

```
{
    "Code": "873345b2-882f-4064-91f0-90669b46c30a",
    "Message_Type": "CFA",
    "Error": false,
    "Errors": null,
    "Checksum": "G6HF5H"
}
```

3.4.5.6 Error response sample

Processing errors

HTTP status		
<< Common response code >>		

3.4.6 DFA – (1.6) De-registration of facility identifier code

3.4.6.1 Description

De-registers a previously known facility for a given F_ID.

The de-registration of a facility identifier code shall lead to the automatic de-registration of related machine identifier codes by the ID issuer and related machine parts (regardless if they are fixed or mobile).

3.4.6.2 Description of the fields

de-registration of facility – request					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Req	Block of basic information elements	Component << Basic Information Request >>	S	M	Message_Type = DFA
EO_ID	Economic operator identifier code	EOID	S	M	
EO_CODE	Economic operator's confirmation code provided in response to the registration of economic operator	EO_CODE	S	M	
F_ID	Facility identifier code	FID	S	M	
Reg_3RD	Indication if the deregistration is made on behalf of a retail outlet operator not otherwise involved in the tobacco trade	Boolean	S	M	0 – No 1 – Yes
Reg_EOID	Identifier of the economic operator that acts on behalf of a retail outlet operator not otherwise involved in the tobacco trade	EOID	S	M, if Reg_3RD = 1	
Extensibility	Optional extensibility field	Text(5000)	S	O	

3.4.6.3 Response:

de-registration of facility – response					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Resp	Block of basic information elements	Component << Basic Information Response >>	S	M	Message_Type = DFA
Extensibility	Optional extensibility field	Text(5000)	S	O	

3.4.6.4 Request sample

```
{
"EO_ID":"QCUKR+1AB020054",
"EO_CODE": "873345b2-882f-4064-91f0-90669b46c30a",
"F_ID":"QCUKR<1AB020054000048",
"Reg_3RD": false,
"Message_Type": "DFA"
}
```

3.4.6.5 Successful response sample

HTTP Status 202

```
{
    "Code": "873345b2-882f-4064-91f0-90669b46c30a",
    "Message_Type": "DFA",
    "Error": false,
    "Errors": null,
    "Checksum": "G6HF5H"
}
```

3.4.6.6 Error response sample

Processing errors

HTTP status		
<< Common response code >>		

3.4.7 RMA – (1.7) Request for a machine identifier code

3.4.7.1 Description

Submit the information for the first registration of a machine.

As of Data Dictionary 2.0, this request supports two different types of Machines. “Entire machines” and “Machine Parts”.

The Economic Operator shall register first the Machine Parts and then the entire Machine that contains the list of associated Machine Parts.

Note that Machine and Machine Parts are identified by the same “Identifier Code” which is the “Machine ID” type.

However, these correspond to different types of entities, as **Entire Machines can be used to request UIDs (upUI codes) but Machine Parts shall not be allowed when requesting for unique identifiers.** This is validated and further described on the ISU (2.1) message.

3.4.7.2 Description of the fields

Registration of manufacturing machine – request					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Req	Block of basic information elements	Component << Basic Information Request >>	S	M	Message_Type = RMA
EO_ID	Economic operator identifier code	EOID	S	M	
EO_CODE	Economic operator's confirmation code provided in response to the registration of economic operator	EO_CODE	S	M	
F_ID	Facility identifier code	FID	S	M	
PrevMID_B	Indication if the object of this request was already registered, e.g. in relation to another machine identifier code	Boolean	S	M	0 – No (first time registration) 1 – Yes
PrevMID_ID	Previous machine identifier used for the object of this request	MID	S	M, if PrevMID_B = 1	
M_entirety	Indication if this request concerns the machine (v. a part of thereof)	Boolean	S	M	0 – No (machine part) 1 – Yes (machine)
P_Producer	Part's producer	Text(20)	S	M, if M_entirety = 0	
P_Model	Part's model	Text(20)	S	M, if M_entirety = 0	
P_Number	Part's serial number	Text(20)	S	M, if M_entirety = 0	
P_Mobile	Indication if this part is intended to be used with multiple machines (fixed v. mobile part)	Boolean	S	M, if M_entirety = 0	0 – No (fixed part) 1 – Yes (mobile part)
P_ATD1	Indication if an anti-tampering device in the	Boolean	S	M, if M_entirety = 0	0 – No 1 – Yes

	sense of Article 2(7) records the functioning of this part				
P_ATD2	Anti-tampering device's serial number	Text(100)	S	M, if M_entirety = 0 and P_ATD1 = 1	
P_Description	Part's description explaining its technical function	Text(500)	S	O	
M_Producer	Machine producer	Text(20)	S	M, if M_entirety = 1	
M_Model	Machine model	Text(20)	S	M, if M_entirety = 1	
M_Number	Machine serial number	Text(20)	S	M, if M_entirety = 1	
M_parts	Indication if the machine consists of multiple separately identifiable parts	Boolean	S	M, if M_entirety = 1	
M_plist	List of the identifiable parts	MID	M	M (limited to 1000 MID), if M_entirety = 1 and M_parts = 1	List of MIDs (parts)
M_ATD	Serial number of the anti-tampering device in the sense of Article 2(7)	Text(100)	S	M, if M_entirety = 1 and M_parts = 0	
M_Capacity	Maximum capacity over 24hour production cycle expressed in unit packets	Integer	S	M, if M_entirety = 1	
Extensibility	Optional extensibility field	Text(5000)	S	O	

3.4.7.3 Response

registration of manufacturing machine – response					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Resp	Block of basic information elements	Component << Basic Information Response >>	S	M	Message_Type = RMA

M_ID	Machine identifier received from the RMA request made to the code issuer.	MID	S	M	
Extensibility	Optional extensibility field	Text(5000)	S	O	

3.4.7.4 Request sample

Machine part request sample:

```
{
    "EO_ID": "QCUKR+1AB020054",
    "EO_CODE": "873345b2-882f-4064-91f0-90669b46c30a",
    "F_ID": "QCUKR<1AB020054000048",
    "PrevMID_B": false,
    "M_entirety": false,
    "P_Producer": "Producer",
    "P_Model": "Model",
    "P_Number": "SN123213",
    "P_Mobile": false,
    "P_ATD1": true,
    "P_ATD2": "XXXYYYSN",
    "Message_Type": "RMA"
}
```

Entire Machine request sample:

```
{
    "EO_ID": "QCUKR+1AB020054",
    "EO_CODE": "873345b2-882f-4064-91f0-90669b46c30a",
    "F_ID": "QCUKR<1AB020054000048",
    "PrevMID_B": false,
    "M_entirety": true,
    "M_Producer": "MProducer",
    "M_Model": "MModel",
    "M_Number": "MSN123213",
    "M_parts": true,
    "M_plist": ["QCUKR5566YYTTSSaa1", " QCUKR5566YYTTSSaa2"],
    "M_Capacity": 20000,
    "Message_Type": "RMA"
}
```

3.4.7.5 Successful response sample

HTTP Status 202

```
{
    "Code": "873345b2-882f-4064-91f0-90669b46c30a",
    "Message_Type": "RMA",
    "MID": "SAMPLEMID123",
    "Error": false,
    "Errors": null,
    "Checksum": "G6HF5H"
}
```

3.4.7.6 Error response sample

Processing errors

HTTP status		
<< Common response code >>		

3.4.8 CMA – (1.8) Correction of information concerning the machine identifier code

3.4.8.1 Description

Submit the information of a machine known to the repository in order to update one or more properties. This information in entirety will overwrite the previous data held regarding the master data of this machine.

EXCEPTION: It must be not allowed to modify an existing M-ID from entire machine to part and vice-versa. The field "M_entirety" cannot be modified.

3.4.8.2 Description of the fields

Field	Description	Data Type	Cardinality	Priority	Values
BasicInfo_Req	Block of basic information elements	Component << Basic Information Request >>	S	M	Message_Type = CMA
EO_ID	Economic operator identifier code	EOID	S	M	
EO_CODE	Economic operator's confirmation code provided in response to the registration of economic operator	EO_CODE	S	M	
F_ID	Facility identifier code	FID	S	M	
M_ID	Machine identifier code	MID	S	M	
PrevMID_B	Indication if the object of this request was already registered, e.g. in relation to another machine identifier code	Boolean	S	M	0 – No (first time registration) 1 – Yes

PrevMID_ID	Previous machine identifier used for the object of this request	MID	S	M, if PrevMID_B = 1	
M_entirety	Indication if this request concerns the machine (v. a part of thereof)	Boolean	S	M	0 – No (machine part) 1 – Yes (machine)
P_Producer	Part's producer	Text(20)	S	M, if M_entirety = 0	
P_Model	Part's model	Text(20)	S	M, if M_entirety = 0	
P_Number	Part's serial number	Text(20)	S	M, if M_entirety = 0	
P_Mobile	Indication if this part is intended to be used with multiple machines (fixed v. mobile part)	Boolean	S	M, if M_entirety = 0	0 – No (fixed part) 1 – Yes (mobile part)
P_ATD1	Indication if an anti-tampering device in the sense of Article 2(7) records the functioning of this part	Boolean	S	M, if M_entirety = 0	0 – No 1 – Yes
P_ATD2	Anti-tampering device's serial number	Text(100)	S	M, if M_entirety = 0 and P_ATD1 = 1	
P_Description	Part's description explaining its technical function	Text(500)	S	O	
M_Producer	Machine producer	Text(20)	S	M, if M_entirety = 1	
M_Model	Machine model	Text(20)	S	M, if M_entirety = 1	
M_Number	Machine serial number	Text(20)	S	M, if M_entirety = 1	
M_parts	Indication if the machine consists of multiple separately identifiable parts	Boolean	S	M, if M_entirety = 1	
M_plist	List of the identifiable parts	MID	M	M (limit 1000 MIDs), if M_entirety = 1 and M_parts = 1	List of MIDs (parts)
M_ATD	Serial number of the anti-tampering device in the sense of Article 2(7)	Text(100)	S	M, if M_entirety = 1 and M_parts = 0	
M_Capacity	Maximum capacity over 24hour production cycle expressed in unit packets	Integer	S	M, if M_entirety = 1	

Extensibility	Optional extensibility field	Text(5000)	S	O	

3.4.8.3 Response:

correction of information concerning the manufacturing machine – response					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Resp	Block of basic information elements	Component << Basic Information Response >>	S	M	Message_Type = CMA
Extensibility	Optional extensibility field	Text(5000)	S	O	

3.4.8.4 Request sample

```
{
    "EO_ID": "QCUKR+1AB020054",
    "EO_CODE": "873345b2-882f-4064-91f0-90669b46c30a",
    "F_ID": "QCUKR<1AB020054000048",
    "M_ID": "QCUKR>1AB020054000012" ,
    "PrevMID_B": 0,
    "M_entirety": 1,
    "M_Producer": "Producer1",
    "M_Model": "model1",
    "M_Number": "MachineNumber",
    "M_parts": 0,
    "M_ATD": "1A2B3c",
    "M_Capacity": 533,
    "Message_Type": "CMA"
}
```

3.4.8.5 Successful response sample

HTTP Status 202

```
{
    "Code": "873345b2-882f-4064-91f0-90669b46c30a",
    "Message_Type": "CMA",
    "Error": false,
    "Errors": null,
    "Checksum": "G6HF5H"
}
```

3.4.8.6 Error response sample

Processing errors

HTTP status		
<< Common response code >>		

3.4.9 DMA – (1.9) De-registration of machine identifier code

3.4.9.1 Description

De-registers a previously known machine for a given M_ID.

All associated Machine Parts of type "FIXED" will be automatically de-registered as well.

Machine Parts of type "Mobile" shall NOT be automatically de-registered.

3.4.9.2 Description of the fields

de-registration of manufacturing machine – request					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Req	Block of basic information elements	Component << Basic Information Request >>	S	M	Message_Type = DMA
EO_ID	Economic operator identifier code	EOID	S	M	
EO_CODE	Economic operator's confirmation code provided in response to the registration of economic operator	EO_CODE	S	M	
F_ID	Facility identifier code	FID	S	M	
M_ID	Machine identifier code	MID	S	M	
Extensibility	Optional extensibility field	Text(5000)	S	O	

3.4.9.3 Response

De-registration of manufacturing machine – response					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Resp	Block of basic information elements	Component << Basic Information Response >>	S	M	Message_Type = DMA
Extensibility	Optional extensibility field	Text(5000)	S	O	

3.4.9.4 Request sample

```
{
"EO_ID":"QCUKR+1AB020054",
"EO_CODE": "873345b2-882f-4064-91f0-90669b46c30a",
"F_ID": "QCUKR<1AB020054000048",
"M_ID": "QCUKR>1AB020054000012" ,
"Code": "873345b2-882f-4064-91f0-90669b46c30a",
"Message_Type":"DMA"
}
```

3.4.9.5 Successful response sample

HTTP Status 202

```
{
    "Code": "873345b2-882f-4064-91f0-90669b46c30a",
    "Message_Type": "DMA",
    "Error": false,
    "Errors": null,
    "Checksum": "G6HF5H
}
```

3.4.9.6 Error response sample

Processing errors

HTTP status		
<< Common response code >>		

3.4.10 ICV – Validate existence of EOID, FID and the MID.

3.4.10.1 Description

Provides the capability for ID Issuers and Primary repositories to check the existence and the activation status of Identifier codes (EOID, FID and MID) and the respective relations.

For the specific case of MIDs with the addition of Machine Parts, the interface will also return if the queried MID is a machine part or not.

3.4.10.2 Description of the fields

Validate existence of EO-ID, F-ID and the M-ID. – request					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Req	Block of basic information elements	Component << Basic Information Request >>	S	M	Message_Type = ICV
EO_IDS	A list of EOIDs to check for existence	EOID	M	O	
F_IDS	A list of FIDs to check for existence	FID	M	O	

M_IDS	A list of MIDs to check for existence	MID	M	O	
R_EOF	A list of relation of EOID and FID to check for existence	Text (Array limit = 2. String text limit = 5000)	M	O	
R_EOFM	A list of relation of EOID, FID and MID to check for existence	Text (Array limit = 3. String text limit = 5000)	M	O	
ICV_Type		Integer	S	O	• 1– request the Activation information, only available for ID Issuers • 2 – request if MID(s) are Entire Machines or Machine Parts • Other- only existence is provided

3.4.10.3 Response

Validate existence of EO-ID, F-ID and the M-ID.– response					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Resp	Block of basic information elements	Component << Basic Information Response >>	S	M	Message_Type = ICV
EO_IDS_EXIST	List of EO_IDs that exist	Boolean	M	O	0 – False 1 - True
F_IDS_EXIST	List of FO_IDs that exist	Boolean	M	O	0 – False 1 - True
M_IDS_EXIST	List of MO_IDs that exist	Boolean	M	O	0 – False 1 - True
EO_IDS_ACTIVE	List of EO_IDs that are active	Boolean	M	M – when ICV_TYPE = 1 only available for ID Issuer	0 – False 1 - True
F_IDS_ACTIVE	List of FO_IDs that are active	Boolean	M	M – when ICV_TYPE = 1 only available for ID Issuer	0 – False 1 - True
M_IDS_ACTIVE	List of MO_IDs that are active	Boolean	M	M – when ICV_TYPE = 1 only available for ID Issuer	0 – False 1 - True

M_IDS_PARTS	List of MO_IDs that are parts	Boolean	M	M – When ICV_Type = 2 only available for ID Issuer	0 – False (the MID is NOT a Machine Part) 1 – True (the MID IS a Machine Part)
R_EOF_EXIST	List of R_EOF that exist	Boolean	M	O	0 – False 1 - True
R_EOFM_EXIST	List of R_EOFM that exist	Boolean	M	O	0 – False 1 - True

3.4.10.4 Request sample

```
{
"Message_Type":"ICV",
"ICV_Type":1,
"EO_IDS": ["QCUKR+1AB020054", "QCBDR+1DE020055"],
"F_IDS": ["QCUKR<1AB020054000048", "QCUKR<1AB020054000049"],
"M_IDS": ["QCUKR>1AB020054000012", "QCUKR>1AB020054000013"],
"R_EOF": [
    ["QCUKR+1AB020054", "QCUKR<1AB020054000048"],
    ["QCUKR+1AB020054", "QCUKR>1AB020054000012"]],
"R_EOFM": [
    ["QCUKR+1AB020054", "QCUKR<1AB020054000048","QCUKR>1AB020054000012"],
    ["QCUKR+1AB020054", "QCUKR>1AB020054000012","QCUKR>1AB020054000012"]],
"Code": null
}
```

3.4.10.5 Successful response sample

HTTP Status 202 (Primary repository)

```
{
    "Code": "873345b2-882f-4064-91f0-90669b46c30a",
    "Message_Type": "ICV",
    "EO_IDS_EXIST": [true, true],
    "F_IDS_EXIST ": [true, true],
    "M_IDS_EXIST ": [true, false],
    "R_EOF_EXIST ": [true, false],
    "R_EOFM_EXIST ": [true, false],
    "Error": false,
    "Errors": null,
    "Checksum": "G6HF5H"
}
```

Example here shows that the 2nd MID is the only one that does not exist in the EU wide registry. Also that the to the economic operator to facility relation is wrong on the 2nd. Also that the facility to machine relation is detected as wrong on the 2nd.

3.4.10.6 Successful response sample ICV_Type = 1

HTTP Status 202 (ID Issuer)


```
{
    "Code": "873345b2-882f-4064-91f0-90669b46c30a",
    "Message_Type": "ICV",
    "EO_IDS_EXIST": [true, true],
    "F_IDS_EXIST ": [true, true],
    "M_IDS_EXIST ": [true, false],
    "EO_IDS_ACTIVE": [true, true],
    "F_IDS_ACTIVE ": [true, true],
    "M_IDS_ACTIVE ": [true, false],
    "R_EOF_EXIST ": [true, false],
    "R_EOFM_EXIST ": [true, false],
    "Error": false,
    "Errors": null,
    "Checksum": "G6HF5H"
}
```


3.4.10.7 Error response sample

Processing errors

HTTP status		
<< Common response code >>		

3.5 Unique identifiers Messages

3.5.1 ISU – (2.1) Request for unit level UIs

3.5.1.1 Description

Request for reporting the issuance of serial numbers at unit packet level.

Note that with the addition of Machine Parts, which share the same format and definition of "Machine identifier code" as "Entire Machines", MID, ID Issuers **SHALL NOT accept any request of unit level UIs for Machine Identifiers which correspond to "Machine Parts",** providing an error to the Economic Operator so that upUIs are requested with a Machine ID that corresponds to an "Entire Machine" instead.

Note that as of v2.0, ID Issuers shall reply to the Manufacturer/Importer request with 2 lists of codes instead of one:

- *upUI(i): The current format delivered, which will be used by the Economic Operator to add the timestamp and convert it into the upUI(L) format transmitted in the 3.1 message*

- *upUI(s): The Human Readable format, which needs to be printed in human readable form in the product and transmitted as part of the 3.1 message.*

3.5.1.2 Description of the fields

Request for unit level UIs – request					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Req	Block of basic information elements	Component << Basic Information Request >>	S	M	Message_Type = ISU
EO_ID	Economic operator identifier code of the submitting entity (either EU manufacturer or EU importer)	EOID	S	M	
F_ID	Facility identifier code	FID	S	M	
Process_Type	Indication if the production process involves machinery	Boolean	S	M	0 – No (only for fully hand made products) 1 – Yes
M_ID	Machine identifier code	MID	S	M, if Process_Type = 1	
P_Type	Type of tobacco product	Integer	S	M	See TobaccoProductType
P_OtherType	Description of other type of tobacco product	Text(200)	S	M, if P_Type = 12 (other tobacco product)	
P_CN	Combined Nomenclature (CN) code	Text(200)	S	O	
P_weight	Average gross weight of unit packet, including packaging, in grams with 0,1 gram accuracy	Decimal	S	M	
P_Brand	Brand of tobacco product under which the product will be marketed on its intended market	Text(200)	S	M	
P_SubType_Exist	Indicates if the product "subtype name" exists. Subtype name provides further product identification beyond a product's brand name.	Boolean	S	M	0 – No 1 – Yes
P_SubType_Name	The product "subtype name" (if any) as marketed on its intended market	Text(200)	S	M, if P_SubType_Exist = 1	
P_units	The number of individual units in the unit packet (number of sticks in the package).	Integer	S	M, if P_Type = 1 or 2 or 3	
TP_ID	The identification number of the product used in the EU-CEG system.	TPID	S	M, if Intended_Market	

Request for unit level UIs – request					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
				is an EU country	
TP_PN	Tobacco product number used in the EU-CEG system (EAN or GTIN or SKU or UPC)	PN	S	M, if Intended _ Market is a Member State O, if Intended _ Market is a third country	
Intended_Market	Intended country of retail sale.	Country	S	M	
Intended_Route1	Indication if the product is intended to be moved across country boarders with terrestrial transport.	Boolean	S	M	0 – No 1 – Yes
Intended_Route2	The first country of terrestrial transport after the product leaves the Member State of manufacturing or the Member State of importation.	Country	S	M, if Intended _Route1 = 1	
Import	Indication if the product is imported into the EU	Boolean	S	M	0 – No 1 – Yes
Req_Quantity	Requested quantity of unit packet level UIs	Integer	S	M	
P_OtherID	Optional Product ID	Text(20)	S	O	

3.5.1.3 Response

Request for unit level UIs – response					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Resp	Block of basic information elements	Component << Basic Information Response >>	S	M	Message_Type = ISU

3.5.1.4 Request sample

```
{
"EO_ID":"QCUKR+1AB020054",
"F_ID": "QCUKR<1AB020054000049"
"Process_Type":false,
"M_ID":"QCUKR123ABCMIDDD",
"P_Type":2,
"P_OtherType":null,
"P_CN": "FG7H68FHF"
"P_Brand":"Product brand A",
"P_SubType_Exist": true,
"P_SubType_Name": "Product brand A's Name in market",
"P_Weight":10.0,
"TP_ID":"1234",
"TP_PN":"1234",
"Intended_Market":"BG",
"Intended_Route1":true,
"Intended_Route2":"BG",
```

```
"Import":false,
"Req_Quantity":2000,
"P_OtherID":"GTINSAMPLE",
"Message_Type":"ISU"
}
```

3.5.1.5 Successful response sample

HTTP Status 202

```
{
    "Code": "873345b2-882f-4064-91f0-90669b46c30a",
    "Message_Type": "ISU",
    "Error": false,
    "Errors": null,
    "Checksum": "G6HF5H"
}
```

3.5.1.6 Error response sample

Processing errors

HTTP status		
<< Common response code >>		

Error body sample

```
{
    "Code": "873345b2-882f-4064-91f0-90669b46c30a",
    "Message_Type": null,
    "Error": true,
    "Errors": [
        {
            "Error_InternalID": "yndkFz7TBEO706frD38hzA",
            "Error_Code": "INVALID_REQUEST_FORMAT",
            "Error_Descr": "The EconomicOperatorIdentifier field is required."
        }
    ]
}
```

3.5.2 IRU – Message to report the issuance of serial numbers at unit packet level

3.5.2.1 Description

Request for reporting the issuance of serial numbers at unit packet level.

As per the changes on the new regulation, this message (which is sent from the ID Issuers to the Router, from the Router to the Primary Repositories and from the Primary Repositories to the Secondary Repository) has been modified to include the new list of upUI(s) that must be reported along with the upUI(i) list.

In order to improve the system stability, it has been decided that a given IRU message will be limited to 20.000 UIs, divided into 10.000 upUI(i) and 10.000 upUI(s). Economic Operators can order more UIs, as usual, but the order will be broken into several IRU messages.
Note that prior to v2.0 the upUI list of UIs sent was in reality of type "upUI(i)". So upUI(i) is the current format (prior to 2.0) transmitted and upUI(s) is the Human Readable.

The relationship of upUI_i with upUI_s will be **in order**, meaning that the first upUI_i will be matched with the first upUI_s, the second upUI_i with the second upUI_s, so on and so forth. This has been chosen instead of a tuple (upUI, upUI_s) to be consistent with the Activation message (EUA - 3.1) and other representation of array matchings in the regulation.

3.5.2.2 Description of the fields

request for reporting the issuance of serial numbers at unit packet level – request					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Req	Block of basic information elements	Component << Basic Information Request >>	S	M	Message_Type = IRU
Event_Time	Intended time of event occurrence	Time(s)	S	M	
Message_Time_long	Message sending Time	Time(L)	S	M	
EO_ID	Economic operator identifier code of the submitting entity (either EU manufacturer or EU importer)	EOID	S	M	
F_ID	Facility identifier code	FID	S	M	
Process_Type	Indication if the production process involves machinery	Boolean	S	M	0 – No (only for fully hand made products) 1 – Yes
M_ID	Machine identifier code	MID	S	M, if Process_Type = 1	
P_Type	Type of tobacco product	Integer	S	M	See TobaccoProductType
P_OtherType	Description of other type of tobacco product	Text(200)	S	M, if P_Type = 12 (other tobacco product)	
P_CN	Combined Nomenclature (CN) code	Text(200)	S	O	
P_Brand	Brand of tobacco product	Text(200)	S	M	
P_weight	Average gross weight of unit packet, including packaging, in grams with 0,1 gram accuracy	Decimal	S	M	

request for reporting the issuance of serial numbers at unit packet level – request					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
P_SubType_Exist	Indicates if the product "subtype name" exists. Subtype name provides further product identification beyond a product's brand name.	Boolean	S	M	0 – No 1 – Yes
P_SubType_Name	The product "subtype name" (if any) as marketed on its intended market	Text(200)	S	M, if P_SubType_Exist = 1	
P_units	The number of individual units in the unit packet (number of sticks in the package).	Integer	S	M, if P_Type = 1 or 2 or 3	
TP_ID	The identification number of the product used in the EU-CEG system.	TPID	S	M, if Intended_Market is an EU country	
TP_PN	Tobacco product number used in the EU-CEG system (EAN or GTIN or SKU or UPC)	PN	S	M, if Intended_ Market is a Member State O, if Intended_ Market is a third country	
Intended_Market	Intended country of retail sale.	Country	S	M	
Intended_Route1	Indication if the product is intended to be moved across country boarders with terrestrial transport.	Boolean	S	M	0 – No 1 – Yes
Intended_Route2	The first country of terrestrial transport after the product leaves the Member State of manufacturing or the Member State of importation.	Country	S	M, if Intended_Route1 = 1	
Import	Indication if the product is imported into the EU	Boolean	S	M	0 – No 1 – Yes
Req_Quantity	Requested quantity of unit packet level UIs – for the current IRU message	Integer	S	M	
Order_Req_Quantity	Total Order Request quantity of unit packet level UIs.	Integer	S	O	
Order_number	Optional EO Request Order Number	Text(50)	S	O	
P_OtherID	Optional Product ID	Text(20)	S	O	
upUI_i	List of unit packet level UI(i) issued by the ID Issuer.	upUI(i)	M (limit 10.000)	M	
upUI_s	List of unit packet level UI(s) issued by the ID Issuer.	upUI(s)	M (limit 10.000)	M	

3.5.2.3 Response

request for reporting the issuance of serial numbers at unit packet level – response					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Resp	Block of basic information elements	Component << Basic Information Response >>	S	M	Message_Type = IRU
RecallExpiry_Time	Calculation of the Expiry date	Time(L)	S	M	

3.5.2.4 Business Validation

	IRU (2.1)
Business rule validation	
Entity Validation	
VAL_ENT_EXIST_EOID	EO_ID
VAL_ENT_EXIST_FID	F_ID
VAL_ENT_EXIST_MID	M_ID
VAL_ENT_ACTIVE_EOID	EO_ID
VAL_ENT_ACTIVE_FID	F_ID
VAL_ENT_ACTIVE_MID	M_ID
VAL_ENT_REL_EOID_FID	EO_ID, F_ID * Note that the case of import will not require this validation.
VAL_ENT_REL_ FID_MID	F_ID, M_ID
VAL_ENT_MPART	M_ID

3.5.2.5 UI duplicate validation

IRU offline validation is performed just after the online validation and takes a few minutes. During the offline validation, the Secondary repository checks the unicity of the UI against the complete database. In case of success, the IRU message is forwarded to the primary repository
A detection of duplicate UI, at any list (upUI(i) or upUI(s)), is considered as exceptional event and is managed on an operational exception. Dentsu Operation team contacts the ID Issuer team in order to address the issue. The IRU will NOT BE FORWARDED to the primary repository in order to avoid any propagation of the duplicates by generating potential side effects on the primary repositories.

3.5.2.6 Machine Part validation

When requesting unit level UIs there is the possibility to indicate the machine where UIs will be printed. With the inclusion of machine parts, there can be a request for UIs indicating a machine part and not a machine. A validation will take place, so the orders include only machines and not machine parts (VAL_ENT_MPART).

3.5.2.7 Request sample

```
{
    "EO_ID":"QCUKR+1AB020054",
    "F_ID": "QCUKR<1AB020054000049",
    "Event_Time":"24032014",
    "Message_Time_Long":"2024-03-20T14:16:45.123Z",
    "Process_Type":true,
    "M_ID":"QCUKRSAMPLEMID",
    "P_Type":2,
    "P_CN": "FG7H68FHF"
    "P_Brand":"Product brand A",
    "P_Weight":10.0,
    "P_SubType_Exist": false,
    "P_units": 20,
    "TP_ID":"1234",
    "TP_PN":"1234",
    "Intended_Market":"BG",
    "Intended_Route1":1,
    "Intended_Route2":"BG",
    "Import":false,
    "Req_Quantity":2,
    "upUI_i":["DANXXXXXXXXXXXX1PR0123456789","DANXXXXXXXXXXXX2PR0123456789"],
    "upUI_s":["DANXXXXXXXXXXXX1","DANXXXXXXXXXXXX2"],
    "Message_Type":"IRU"
}
```

3.5.2.8 Successful response sample

HTTP Status 202

```
{
    "Code": "873345b2-882f-4064-91f0-90669b46c30a",
    "Message_Type": "IRU",
    "RecallExpiry_Time":"19092014",
    "Error": false,
    "Errors": null,
    "Checksum": "G6HF5H"
}
```

3.5.2.9 Error response sample

Processing errors

HTTP status	Error Code	Error Description
<< Common response code >>		
400	EOID_NOT_EXIST_OR_ACTIVE	VAL_ENT_EXIST_EOID
400	FID_NOT_EXIST_OR_ACTIVE	VAL_ENT_EXIST_FID
400	MID_NOT_EXIST_OR_ACTIVE	VAL_ENT_EXIST_MID
400	EOID_NOT_EXIST_OR_ACTIVE	VAL_ENT_ACTIVE_EOID
400	FID_NOT_EXIST_OR_ACTIVE	VAL_ENT_ACTIVE_FID
400	MID_NOT_EXIST_OR_ACTIVE	VAL_ENT_ACTIVE_MID
400	FID_NOT_RELATED_TO_EOID	VAL_ENT_REL_EOID_FID
400	MID_NOT_RELATED_TO_FID	VAL_ENT_REL_ FID_MID
400	MID_MACHINE_PART	VAL_ENT_MPART

Error body sample

```
{
    "Code": null,
    "Message_Type": null,
    "Error": true,
    "Errors": [
        {
            "Error_InternalID": "yndkFz7TBEO706frD38hzA",
            "Error_Code": "INVALID_REQUEST_FORMAT",
            "Error_Descr": "The EconomicOperatorIdentifier field is required."
        }
    ]
}
```


3.5.3 IRUD – Message to report the issuance of serial numbers at unit packet level callback

3.5.3.1 Description

This IRUD callback message is a response to the original IRU message indicating the delivery status of IRU message.

3.5.3.2 Description of the fields

IRUD – request					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
Message_Type	The identifier of the type of message	Text	S	M	Message_Type = IRUD
IRU_Code	IRU recallCode		S	M	
IRU_Status	The status of the delivery of a specific IRU message	Boolean	S	M	0 – False 1 – True
IRU_Status_Description	Description of the status or the error message	Text	S	O	
Error	Indicates the failure of the message reception	Boolean	S	M	0 – No 1- Yes
Errors	Array containing Error_Code, Error_Descr, InternalId	Text	S	M if Error = 1	

3.5.3.3 Response

IRUD – response					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
Message_Type	The identifier of the type of message	Text	S	M	Message_Type = IRUD
Code	Unique identifier of the message. Used for recall too.	Text(50)	S	M	
Error	Indicates the failure of the message reception	Boolean	S	M	0 – No 1- Yes
Errors	Array containing Error_Code, Error_Descr, InternalId	Text	S	M if Error = 1	

3.5.3.4 Request sample

```
{
    "Code": "873345b2-882f-4064-91f0-90669b46c30b",
    "Message_Type": "IRUD",
    "IRU_Code": "873345b2-882f-4064-91f0-90669b46c30a"
    "IRU_Status": 0,
    "IRU_Status_Description": "optional description",
    "Error": false,
    "Errors": null,
```

3.5.3.5 Successful response sample

HTTP Status 200

```
{
    "Message_Type": "IRUD",
    "Code": "873345b2-882f-4064-91f0-90669b46c30a",
    "Error": false,
    "Errors": null,
    "Checksum": "G6HF5H"
}
```

3.5.4 ISA – (2.2) Request for aggregated level UIs

3.5.4.1 Description

Request for reporting the issuance of serial numbers at aggregated level

3.5.4.2 Description of the fields

request for reporting the issuance of serial numbers at aggregated level – request					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Req	Block of basic information elements	Component << Basic Information Request >>	S	M	Message_Type = ISA
EO_ID	Economic operator identifier code of the submitting entity	EOID	S	M	
F_ID	Facility identifier code	FID	S	M	
Req_Quantity	Requested quantity of aggregated level UIs	Integer	S	M	

3.5.4.3 Response

request for reporting the issuance of serial numbers at aggregated level – response					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Resp	Block of basic information elements	Component << Basic Information Response >>	S	M	Message_Type = ISA

3.5.4.4 Request sample

```
{
  "EO_ID": "QCUKR+1AB020054",
  "F_ID": "QCUKR<1AB020054000049",
  "Req_Quantity": 2,
  "Message_Type": "ISA",
}
```

3.5.4.5 Successful response sample

HTTP Status 202

```
{
    "Code": "873345b2-882f-4064-91f0-90669b46c30a",
    "Message_Type": "ISA",
    "Error": false,
    "Errors": null,
    "Checksum": "G6HF5H"
}
```

3.5.4.6 Error response sample

Processing errors

HTTP status		
<< Common response code >>		

3.5.5 IRA – Request for reporting the issuance of serial numbers at aggregated level

3.5.5.1 Description

Request for reporting the issuance of serial numbers at aggregated level

3.5.5.2 Description of the fields

request for reporting the issuance of serial numbers at aggregated level – request					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Req	Block of basic information elements	Component << Basic Information Request >>	S	M	Message_Type = IRA
Event_Time	Intended time of event occurrence	Time(s)	S	M	
Message_Time_Long	Message sending Time	Time(L)	S	M	

EO_ID	Economic operator identifier code of the submitting entity	EOID	S	M	
F_ID	Facility identifier code	FID	S	M	
Req_Quantity	Requested quantity of aggregated level UIs	Integer	S	M	
aUI	List of aggregated level UIs	aUI	M	M	

3.5.5.3 Response

request for reporting the issuance of serial numbers at aggregated level – response					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Resp	Block of basic information elements	Component << Basic Information Response >>	S	M	Message_Type = IRA

3.5.5.4 Business Validation

	IRA (2.2)
Business rule validation	
Entity Validation	
VAL_ENT_EXIST_EOID	EO_ID
VAL_ENT_EXIST_FID	F_ID
VAL_ENT_ACTIVE_EOID	EO_ID
VAL_ENT_ACTIVE_FID	F_ID
VAL_ENT_REL_EOID_FID	EO_ID, F_ID
VAL_ENT_REL_ FID_MID	EO_ID, F_ID

3.5.5.5 Request sample

```
{
  "EO_ID": "QCUKR+1AB020054",
  "F_ID": "QCUKR<1AB020054000049",
  "Event_Time" : "19032014",
  "Message_Time_Long":"2019-03-20T14:16:45Z",
  "Req_Quantity": 2,
  "aUI": ["DANXXXXXXXXXXXX1FA000001", " DANXXXXXXXXXXX2FA000001" ],
  "Message_Type": "IRA"
}
```

3.5.5.6 Successful response sample

HTTP Status 202

```
{
    "Code": "873345b2-882f-4064-91f0-90669b46c30a",
    "Message_Type": "IRA",
    "Error": false,
    "Errors": null,
    "Checksum": "G6HF5H"
}
```

3.5.5.7 Error response sample

HTTP status	Error Code	Error Description
<< Common response code >>		
400	EOID_NOT_EXIST_OR_ACTIVE	VAL_ENT_EXIST_EOID
400	FID_NOT_EXIST_OR_ACTIVE	VAL_ENT_EXIST_FID
400	EOID_NOT_EXIST_OR_ACTIVE	VAL_ENT_ACTIVE_EOID
400	FID_NOT_EXIST_OR_ACTIVE	VAL_ENT_ACTIVE_FID
400	FID_NOT_RELATED_TO_EOID	VAL_ENT_REL_EOID_FID

3.5.6 IDA – (2.3) Request for deactivation of UIs

3.5.6.1 Description

Changes the status of the UIs listed in the request to "deactivated" The hierarchy below these UIs will be managed depending on the deactivation reason for the parent UI.

3.5.6.1.1 Product deactivation

If the deactivation reason was Deact_Reason1 = 1 (Product destroyed) or 2 (Product stolen) then the full hierarchy is deactivated.

It is only possible to declare IDA with Deact_Reason1 = 2 for upUIs as long as the upUIs have been produced (meaning, they have been associated with a successful 3.1 Activation EUA message prior to the declaration of the IDA message). If this is not the case, a UI_SEQUENCE_ERROR has to be returned by the repositories system, as product that is not produced yet cannot be physically stolen.

Example of a Pallet product deactivation:

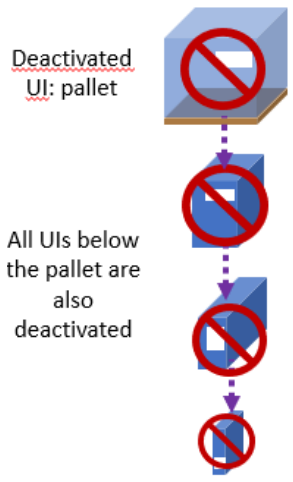


Figure 5 Implicit disaggregation triggered by a IDA Deact_Reason1 = 1 or 2

3.5.6.1.2 UI deactivation

If the deactivation reason was Deact_Reason1 = 3 (UI Destroyed),4 (UI Stolen) ,5 (UI Unused) or 6 (Other), then only the explicitly mentioned UIs are deactivated and therefore the hierarchy related UIs (implicit UIs) would still be existing in the Secondary as active. It is the responsibility of the Economic Operator to report the stolen UIs as stolen when they are aware of it (send a deactivation message for the stolen UIs)

Note that in these cases the Deactivation will also trigger an implicit disaggregation.

Example of UI deactivation for a pallet

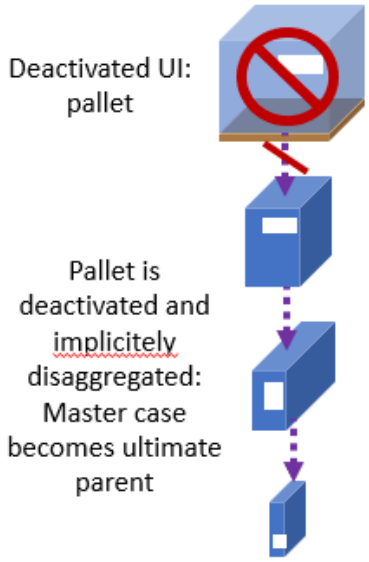


Figure 6 Implicit disaggregation triggered by an IDA Deact_Reason1 = 3,4,5 or 6

3.5.6.1.3 Deactivation upUI

The deactivation event 2.3 to allow the deactivation of the UI not only based on the upUI(s) but as well on the upUI(L) and upUI(i).
The receiving system (Primary repository, Router and Secondary repository) implements the existence validation on the different UI: upUI(s), upUI(L) and upUI(i).

For upUIs that are GENERATED by the ID Issuer via IRU message to the Repositories system but have not received an Activation message (3.1 – EUA) only the upUI(i) representation of the upUI will be used in message 2.3 – Deactivation. This means that before producing the upUIs, the upUI(s) (Human Readable) representation of the upUI **cannot be used** to deactivate the upUI, as this has not been applied yet to any product. If an IDA is received with the upUI(s) before the product has been applied, the repositories system will reply with a "UI_NOT_EXIST" error.

After the upUI has been produced (declared as part of a 3.1 Activation message), then the 3 representations may be used to deactivate it (upUI(s), upUI(i), upUI(L)). Note that this is the case before the v2.0 of the specifications.

Legal basis:
The message 2.3 is clearly structured to permit the deactivation of UIs at any moment of their lifecycle, including when UIs are unused, i.e. not applied. In this sense, the addition of different "representations" of the same UI will be a purely technical extension. To recall, Annex II enables Dentsu "to extend the message content for strictly technical purposes to secure smooth functioning of the tobacco products traceability system".

3.5.6.2 Description of the fields

request for the deactivation of UIs – request					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Req	Block of basic information elements	Component << Basic Information Request >>	S	M	Message_Type = IDA
Event_Time	Intended time of event occurrence	Time(s)	S	M	
Message_Time_long	Message sending Time	Time(L)	S	M	
EO_ID	Economic operator identifier code of the submitting entity	EOID	S	M	
Deact_Type	Deactivation of unit packet or aggregated level UIs	Integer	S	M	1 – Unit pack level UIs 2 – Aggregated level UIs
Deact_Reason1	Identification of the reason for deactivation	Integer	S	M	See DeactivationReasonType
Deact_Reason2	Description of other reason	Text(5000)	S	M, if Deact_Reason1 = 6 (other reason)	
Deact_Reason3	Additional description of the reason	Text(Limited to the set of known deactivation_types)	S	O	
Deact_upUI	List of unit packet level UIs to be deactivated	upUI(s) or upUI(L) or upUI(i)	M	M, if Deact_Type = 1	
Deact_aUI	List of aggregated level UIs to be deactivated	aUI	M	M, if Deact_Type = 2	

3.5.6.3 Response

request for the deactivation of UIs – response					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Resp	Block of basic information elements	Component << Basic Information Response >>	S	M	Message_Type = IDA

3.5.6.4 Business Validation

	IDA (2.3)
Business rule validation	
UI creation	
VAL_UI_EXIST_APP	Deact_upUI,
VAL_UI_EXIST_UPUI_SEQ	Deact_upUI,
VAL_UI_EXIST_AUI_SEQ	Deact_aUI
Entity Validation	
VAL_ENT_EXIST_EOID	EO_ID
VAL_ENT_ACTIVE_EOID	EO_ID
Sequence Validation	
VAL_UI_ORD_DEACTIVATED	Deact_upUI, Deact_aUI

Automatic deactivation

The deactivation event (IDA 2.3) should be accepted (https status 200) for UI that have been expired (automatically deactivated)

3.5.6.5 Implicit disaggregation trigger

The deactivation event can trigger an implicit disaggregation when a child UI is identified as part of the event.

3.5.6.6 Request sample

```
{
  "EO_ID": "QCUKR+1AB020054",
  "Event_Time" : "19032014",
  "Message_Time_Long":"2019-03-20T14:16:45Z",
  "Deact_Type": 1,
  "Deact_Reason1": 1,
  "Deact_Reason2": "reason one",
  "Deact_Reason3": "reason two",
  "Deact_upUI": [ "DANXXXXXXXXXXXX1PR0123456789" ],
  "Deact_apUI": [],
  "Message_Type": "IDA"
}
```

3.5.6.7 Successful response sample

HTTP Status 202

```
{
   "Code": "873345b2-882f-4064-91f0-90669b46c30a",
    "Message_Type": "IDA",
    "Error": false,
    "Errors": null,
    "Checksum": "G6HF5H"
}
```

3.5.6.8 Error response sample

Processing errors

HTTP status	Error Code	Error Description
<< Common response code >>		
400	UIS_APPLICATION_ERROR	VAL_UI_EXIST_APP
400	UI_NOT_EXIST	VAL_UI_EXIST_UPUI_SEQ
400	UI_NOT_EXIST	VAL_UI_EXIST_AUI_SEQ
400	EOID_NOT_EXIST_OR_ACTIVE	VAL_ENT_EXIST_EOID
400	EOID_NOT_EXIST_OR_ACTIVE	VAL_ENT_ACTIVE_EOID
400		VAL_UI_ORD_DEACTIVATED

3.5.7 ICM – Validate the delivery of an IRU message.

3.5.7.1 Description

This optional message allows the ID Issuer to retrieve the state of the delivery of a specific IRU message.

3.5.7.2 Description of the fields

Validate the delivery of an IRU.– request					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Req	Block of basic information elements	Component << Basic Information Request >>	S	M	Message_Type = ICM
IRU_Code	IRU recallCode		S	M	

3.5.7.3 Response

Validate the delivery of an IRU.–.– response					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Resp	Block of basic information elements	Component << Basic Information Response >>	S	M	Message_Type = ICM
IRU_Code	IRU recallCode		S	M	
IRU_Status	The status of the delivery of a specific IRU message	Boolean	S	M	0 – False 1 – True

IRU_Status_Description	Description of the status or the error message	Text	S	O	

3.5.7.4 Request sample

```
{
"Message_Type":"ICM"
"IRU_Code": "873345b2-882f-4064-91f0-90669b46c30a"
}
```

3.5.7.5 Successful response sample

HTTP Status 202

```
{
    "Code": "873345b2-882f-4064-91f0-90669b46c30b",
    "Message_Type": "ICM",
    "IRU_Code": "873345b2-882f-4064-91f0-90669b46c30a"
    "IRU_Status": 0,
    " IRU_Status_Description": "optional description",
    "Error": false,
    "Errors": null,
    "Checksum": "G6HF5H"
}
```

3.5.7.6 Error response sample

Processing errors

HTTP status	Error Code	Error Description
<< Common response code >>		

3.5.8 IRR – (2.4) Request for reactivation of UIs for products reported as stolen but recovered

3.5.8.1 Description

Cancels a previous Deactivation of UIs due to "Product Stolen".

The hierarchy below these UIs will be reactivated as well, since "Product Stolen" was a recursive deactivation.

This message is only permitted if in a preceding message type 2.3, field Deact_Reason1 = 2 and such message was positively acknowledged.

The reactivation can be done at a upUI or aUI level and it implies the UIs will be considered as "received" into the IRR's location (the FID declared in the IRR message). In this regard, the IRR behaves like an Arrival Return – ERP Return.

Note that there is no location validation on the reactivation event, meaning that the UIs can be reactivated at a different location where they were reported as stolen (or they can also be reactivated if they were reported as stolen in transit). Note that this is the same behaviour as Arrivals Return – ERP Return.

3.5.8.2 Description of the fields

request for the reactivation of UIs – request					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Req	Block of basic information elements	Component << Basic Information Request >>	S	M	Message_Type = IRR
Event_Time	Intended time of event occurrence	Time(s)	S	M	
Message_Time_long	Message sending Time	Time(L)	S	M	
EO_ID	Economic operator identifier code of the submitting entity	EOID	S	M	
F_ID	Facility identifier code (facility of recovery)	FID	S	M	
React_Type	Reactivation of unit packet or aggregated level UIs	Integer	S	M	1 – Unit packet level UIs 2 – Aggregated level UIs
React_Reason	Description of the context of reactivation	Text(5000)	S	O	
React_upUI	List of unit packet level UIs to be reactivated	upUI(L) or upUI(i) or upUI(s)	M	M, if React_Type = 1	
React_aUI	List of aggregated level UIs to be reactivated	aUI	M	M, if React_Type = 2	

3.5.8.3 Response

request for the reactivation of UIs – response					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Resp	Block of basic information elements	Component << Basic Information Response >>	S	M	Message_Type = IRR

3.5.8.4 Business Validation

	IRR (2.4)
Business rule validation	
UI creation	
VAL_UI_EXIST_APP	React_upUI,
VAL_UI_EXIST_UPUI	React_upUI,
VAL_UI_EXIST_AUI	React_aUI

Entity Validation	
VAL_ENT_EXIST_EOID	EO_ID
VAL_ENT_ACTIVE_EOID	EO_ID
VAL_ENT_EXIST_FID	F_ID
VAL_ENT_ACTIVE_FID	F_ID
Sequence Validation	
VAL_UI_ORD_REACTIVATION_NOT_ALLOWED	React_upUI, React_aUI

3.5.8.5 Implicit disaggregation trigger

The reactivation event may trigger implicit disaggregation in the case that the stolen goods were reported higher up in the hierarchy and the recovered goods are reported lower in such hierarchy.

3.5.8.6 Request sample

```
{
  "EO_ID": "QCUKR+1AB020054",
  "F_ID": " QCUKR<1AB020054000048",
  "Event_Time" : "19032014",
  "Message_Time_Long":"2019-03-20T14:16:45Z",
  "React_Type": 1,
  "React_Reason": "reason one",
  "React_upUI": [ "DANXXXXXXXXXXXX1PR0123456789" ],
  "React_apUI": [],
  "Message_Type": "IRR"
}
```

3.5.8.7 Successful response sample

HTTP Status 202

```
{
   "Code": "873345b2-882f-4064-91f0-90669b46c30a",
    "Message_Type": "IRR",
    "Error": false,
    "Errors": null,
    "Checksum": "G6HF5H"
}
```

3.5.8.8 Error response sample

Processing errors

HTTP status	Error Code	Error Description
<< Common response code >>		
400	UIS_APPLICATION_ERROR	VAL_UI_EXIST_APP
400	UI_NOT_EXIST	VAL_UI_EXIST_UPUI
400	UI_NOT_EXIST	VAL_UI_EXIST_AUI
400	EOID_NOT_EXIST_OR_ACTIVE	VAL_ENT_EXIST_EOID
400	EOID_NOT_EXIST_OR_ACTIVE	VAL_ENT_ACTIVE_EOID
400	FID_NOT_EXIST_OR_ACTIVE	VAL_ENT_EXIST_FID

400	FID_NOT_EXIST_OR_ACTIVE	VAL_ENT_ACTIVE_FID
400	FAILED_VALIDATION	VAL_UI_ORD_REACTIVATION_NOT_ALLOWED

3.6 Reporting operational events (product movement information)

3.6.1 EUA – (3.1) Application of unit level UIs on unit packets

3.6.1.1 Description

Event notification when the code is applied / printed on unit packets.

Note on the new validation "VAL_UI_HR_EXIST". It is **only possible to perform this validation for codes issued after the go-live date of the v2.0 of the specifications.** Therefore, this validation shall only be performed by the Primary Repositories and Secondary Repository when the IRU message contains the upUI(s) list when received from the ID Issuers, otherwise it is not possible to perform the validation.

3.6.1.2 Description of the fields

upUI application event					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Req	Block of basic information elements	Component << Basic Information Request >>	S	M	Message_Type = EUA
Event_Time	Time of event occurrence	Time(s)	S	M	
Message_Time_long	Message sending Time	Time(L)	S	M	
EO_ID	Economic operator identifier code of the submitting entity	EOID	S	M	
F_ID	Facility identifier code	FID	S	M	
upUI_1	List of unit packet level UIs to be recorded (full length)	upUI(L)	M	M	
upUI_2	List of corresponding unit packet level UIs to be recorded (as visible in human readable format) indicated in the same order as upUI_1	upUI(s)	M	M	
upUI_comment	Comments by the reporting entity	Text(5000)	S	O	

3.6.1.3 Business validation

	EUA (3.1)
Technical validation	
VAL_UI_MULT_MSG	upUI_1, upUI_2
Business rule validation	
UI creation	
VAL_UI_EXIST_APP	upUI_1
VAL_UI_DUPLICATE_APP	upUI_1
VAL_UI_EXPIRY	upUI_1
VAL_UI_HR_EXIST	upUI_2
Entity Validation	
VAL_ENT_EXIST_EOID	EO_ID
VAL_ENT_EXIST_FID	F_ID
VAL_ENT_ACTIVE_EOID	EO_ID
VAL_ENT_ACTIVE_FID	F_ID
Sequence Validation	
VAL_UI_FID_APP	F_ID with information for upUI_1
VAL_UI_ORD_REACTIVATION	upUI_1
Message Timing	
VAL_EVT_24H	Event_Time

3.6.1.4 Sequence validation

The following table represents the authorized transitions for a UI and specifically the previous message for the UI.

	IRU 2.1
Message Received	
EUA 3.1	Yes
EUA 3.1 Import	Yes

No	Next message not allowed for the UI (including different aggregatio
Yes	Next Message allowed
Yes (with Location Validation)	Next Message allowed with location validation

The EUA (3.1) application event must be preceded by an IRU event that reports the issued UIs.

3.6.1.5 Response

upUI application event – response					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Resp	Block of basic information elements	Component << Basic Information Response >>	S	M	Message_Type = EUA

3.6.1.6 Request sample

```
{
  "EO_ID": "QCUKR+1AB020054",
  "F_ID": "QCUKR<1AB020054000049",
  "Event_Time": "19032014",
  "Message_Time_Long":"2019-03-20T14:16:45Z",
  "upUI_1": [
    "DANXXXXXXXXXXX1PR012345678919030110",
    "DANXXXXXXXXXXX2PR012345678919030110"
  ],
  "upUI_2": [
    "DANXXXXXXXXXXX1PR0123456789",
    "DANXXXXXXXXXXX2PR0123456789"
  ],
  "upUI_comment": "Comments",
  "Message_Type": "EUA"
}
```

3.6.1.7 Successful response sample

HTTP Status 202

```
{
   "Code": "873345b2-882f-4064-91f0-90669b46c30a",
    "Message_Type": "EUA",
    "Error": false,
    "Errors": null,
    "Checksum": "G6HF5H"
}
```

3.6.1.8 Error response sample

HTTP status	Error Code	Error Description
<< Common response code >>		
400	MULTIPLE_UID	VAL_UI_MULT_MSG
400	UIS_APPLICATION_ERROR	VAL_UI_EXIST_APP
400	UIS_APPLICATION_ERROR	VAL_UI_HR_EXIST
400	UIS_APPLICATION_ERROR	VAL_UI_DUPLICATE_APP
400	UI_NOT_EXIST	VAL_UI_EXPIRY
400	EOID_NOT_EXIST_OR_ACTIVE	VAL_ENT_EXIST_EOID
400	FID_NOT_EXIST_OR_ACTIVE	VAL_ENT_EXIST_FID
400	EOID_NOT_EXIST_OR_ACTIVE	VAL_ENT_ACTIVE_EOID
400	FID_NOT_EXIST_OR_ACTIVE	VAL_ENT_ACTIVE_FID
400	FID_MISMATCH	VAL_UI_FID_APP
400	UI_DEACTIVATED	VAL_UI_ORD_REACTIVATION
299	OPERATION_WITHIN_24_HOURS	VAL_EVT_24H

3.6.2 EPA – (3.2) Application of aggregated level UIs on aggregated packaging

3.6.2.1 Description

Event notification when the code is applied / printed on an aggregation container. This also records the items that are aggregated into this container.

3.6.2.2 Description of the fields

Application of aggregated level UIs on aggregated packaging - request					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Req	Block of basic information elements	Component << Basic Information Request >>	S	M	Message_Type = EPA
EO_ID	Economic operator identifier code of the submitting entity	EOID	S	M	
F_ID	Facility identifier code	FID	S	M	
Event_Time	Time of event occurrence	Time(s)	S	M	
Message_Time_long	Message sending Time	Time(L)	S	M	
aUI	Aggregated level UI	aUI	S	M	
Aggregation_Type	Identification of aggregation type	Integer	S	M	1 – aggregation of only unit packet level UIs 2 – aggregation of only aggregated level UIs 3 – aggregation of both unit packet and aggregated level UIs
Aggregated_UIs1	List of unit packet level UIs subject to aggregation	upUI(L)	M	M, if Aggregation_Type = 1 or 3	
Aggregated_UIs2	List of aggregated level UIs subject to further aggregation	aUI	M	M, if Aggregation_Type = 2 or 3	
aUI_comment	Comments by the reporting entity	Text(5000)	S	O	
Information	Indicates the request of additional optional information	Boolean	S	O	0 – No 1- Yes

3.6.2.3 Business validation

	EPA (3.2)
Technical validation	
VAL_UI_MULT_MSG	aUI, Aggregated_UIs1
VAL_FIE_REF	aUI, Aggregated_UIs1. Error Descr for Circular Reference issue: The message contains UI values that form a circular reference
Business rule validation	

UI creation	
VAL_UI_EXIST_UPUI	Aggregated_UIs1
VAL_UI_EXIST_AUI	aUI, Aggregated_UIs2
VAL_UI_EXIST_UPUI_SEQ	Aggregated_UIs1
VAL_UI_EXIST_AUI_SEQ	Aggregated_UIs2
VAL_UI_EXPIRY	aUI, Aggregated_UIs1
Entity Validation	
VAL_ENT_EXIST_EOID	EO_ID
VAL_ENT_EXIST_FID	F_ID
VAL_ENT_ACTIVE_EOID	EO_ID
Sequence Validation	
VAL_UI_ORD_DEACTIVATED	aUI, Aggregated_UIs2
VAL_UI_ORD_AGG_MULT	aUI, Aggregated_UIs2
VAL_UI_ORD_IMPLDISAGG	aUI, Aggregated_UIs2
VAL_UI_ORD_AGG_FID	F_ID for Aggregated_UIs1 and Aggregated_UIs2
Message Timing	
VAL_EVT_24H	Event_Time

3.6.2.4 Sequence Validation

The following table represents the authorized transitions for a UI and specifically the previous message for the UI.

Message Received	IRA 2.2	IRR 2.4	EUA 3.1	EUA 3.1 Import	EPA 3.2 parent UI	EPA 3.2 parent UI Import	EPA 3.2 Child	ERP 3.4	ERP 3.4 (Return)	EUD 3.6
EPA 3.2 parent UI	Yes	Yes	No	No	No	No	Yes	Yes	Yes	Yes
EPA 3.2 child UI (upUI)	No	Yes	Yes	Yes	No	No	Yes	Yes	Yes	No
EPA 3.2 child UI (aUI)	No	Yes	No	No	Yes	Yes	Yes	Yes	Yes	No

No	Next message not allowed for the UI (including different aggregatior
Yes	Next Message allowed
Yes (with Location Validation)	Next Message allowed with location validation

A parent aUI can be

- Self generated parent UI. This event will correspond to the initial commissioning of the aUI.

- ID Issuer generated aUI reported using an IRA (2.2) event.
- As a aUI that is re-used. This aUI must be preceded by a EUD (3.6) (explicit) disaggregation event. Note that in the case of a reuse, the location validation is not performed. In other words, the aUI can be re aggregated in any location.
- Reactivated aUIs by message 2.4 can be re-aggregated with different children after it's reactivation.
- Reactivated aUIs by message 2.4 can be used as children of new aggregations after it's reactivation
- Reactivated upUIs by message 2.4 can be aggregated to new parents after it's reactivation.

A Child UI can be reported if it is present in the location of the aggregation.

- A upUI can be applied, following a EUA (3.1) event.
- A upUI can be present in the location following a disaggregation event of a previous aggregation.
- An aUI can be present in the location following a disaggregation event of a previous aggregation where it was reported as child UI.
- A upUI or aUI that are still part of a valid aggregation as child UIs. The reporting of the aggregation event will trigger an implicit disaggregation of the initial aggregation.
- A upUI or aUI that are arrived ERP (3.4) event.

Location validation on the child UI Is performed for the aggregation as these UI should be in the location of the aggregation.

3.6.2.4.1 Import

Location validation exception for Imports Due to the fact that product movement outside of the EU are not subject to being reported into the EU tobacco track and trace system (i.e. Dispatch/Arrivals), in the event that an Economic Operator requires to modify the hierarchy of the goods before the import into EU, there will be Aggregation (3.2 – EPA) and Disaggregation (3.6 – EUD) messages on different locations (Facility IDs) without a Dispatch/Arrival movement between them. For these cases, the Location validation (VAL_UI_ORG_AGG_FID) will not be applicable (so all products BEFORE the first Arrival 3.4 into the EU may be modified in terms of their logistic hierarchy).

3.6.2.5 Implicit disaggregation trigger

This event can trigger an implicit disaggregation when a child UI is identified as part of the event.

Example: Implicit Disaggregation occurring when re-Aggregating

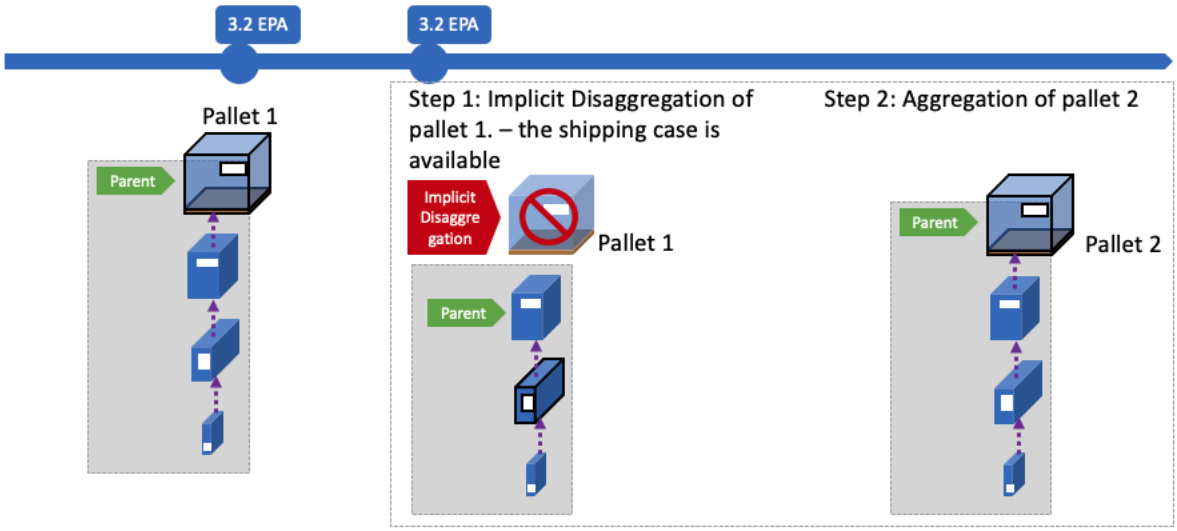


Figure 7 Implicit disaggregation triggered by an EPA (3.2) event

The second aggregation event (EPA 3.2) result in the creation of the new pallet with the same content as pallet 1 that is implicitly disaggregated.

3.6.2.6 Response

Application of aggregated level UIs on aggregated packaging – response					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Resp	Block of basic information elements	Component << Basic Information Response >>	S	M	Message_Type = EPA
Basic Information Block	Additional optional acknowledgment Information	Component << Basic Information Block >>	S	O	

3.6.2.7 Request sample

```
{
  "EO_ID": "QCUKR+1AB020054",
  "F_ID": "QCUKR<1AB020054000049",
  "Event_Time": "19032014",
  "Message_Time_Long":"2019-03-20T14:16:45Z",
  "Aggregation_Type": "1",
  "aUI" : "DANXXXXXXXXXXXX1FA00000119030110",
  "Aggregated_UIs1": ["DANXXXXXXXXXXX1PR012345678919030110",
"DANXXXXXXXXXXX2PR012345678919030110",
"DANXXXXXXXXXXX3PR012345678919030110",
"DANXXXXXXXXXX10FA00000119030110"],
  "Aggregated_UIs2": ["DANXXXXXXXXXX10FA00000119030110"],
  "aUI_comment": "Comments",
  "Message_Type": "EPA"}
```

3.6.2.8 Successful response sample

HTTP Status 202


```
{
    "Code": "873345b2-882f-4064-91f0-90669b46c30a",
    "Message_Type": "EPA",
    "Error": false,
    "Errors": null,
    "Checksum": "G6HF5H"
}
```


3.6.2.9 Error response sample

HTTP status	Error Code	Error Description
<< Common response code >>		
400	UIS_APPLICATION_ERROR	VAL_UI_EXIST_UPUI
400	UI_NOT_EXIST	VAL_UI_EXIST_AUI
400	UI_NOT_VALID	VAL_UI_EXIST_UPUI_SEQ
400	UI_NOT_EXIST	VAL_UI_EXIST_AUI_SEQ
400	UI_EXPIRED	VAL_UI_EXPIRY
400	EOID_NOT_EXIST_OR_ACTIVE	VAL_ENT_EXIST_EOID
400	FID_NOT_EXIST_OR_ACTIVE	VAL_ENT_EXIST_FID
400	EOID_NOT_EXIST_OR_ACTIVE	VAL_ENT_ACTIVE_EOID
400	FID_NOT_EXIST_OR_ACTIVE	VAL_UI_ORD_DEACTIVATED
400	MULTIPLE_AGGREGATION	VAL_UI_ORD_AGG_MULT
400	UI_ALREADY_DISAGGREGATED	VAL_UI_ORD_IMPLDISAGG
400	LOCATION_MISMATCH	VAL_UI_ORD_AGG_FID
299	OPERATION_WITHIN_24_HOURS	VAL_EVT_24H
400	FAILED_VALIDATION	VAL_FIE_REF

3.6.3 EDP – (3.3) Dispatch of tobacco products from a facility

3.6.3.1 Description

Record that the UIs listed in the call have been dispatched from the economic identifier.

3.6.3.2 Description of the fields

Dispatch of tobacco products from a facility event					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Req	Block of basic information elements	Component << Basic Information Request >>	S	M	Message_Type = EDP
EO_ID	Economic operator identifier code of the submitting entity	EOID	S	M	
Event_Time	Time of event occurrence	Time (s)	S	M	
Message_Time_long	Message sending Time	Time (L)	S	M	

Dispatch of tobacco products from a facility event					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
F_ID	Dispatch facility identifier code	FID	S	M	
Destination_ID1	Indication if the destination facility is located on the EU territory and if it is a vending machine (VM)	Integer	S	M	1 – Non EU dest. 2 – EU destination other than VM – fixed quantity delivery 3 – EU VM(s) 4 – EU destination other than VM – delivery with VV
Destination_ID2	Destination facility identifier code	FID	S	M, if Destination_ID1 = 2	
Destination_ID3	Destination facility identifier code(s) – possible multiple vending machines	FID	M (limited to 1000 FID)	M, if Destination_ID1 = 3	
Destination_ID4	Destination id facility codes	FID	M (limited to 1000 FID)	M, if Destination_ID1 = 4	
Destination_ID5	Destination facility's street name and house number (or road number and kilometer)	Text(300)	S	M, if Destination_ID 1 = 1	
Destination_ID6	Destination facility's municipality (city, town or village)	Text(100)	S	M, if Destination_ID 1 = 1	
Destination_ID7	Destination facility's postal code	Text(50)	S	M, if Destination_ID 1 = 1	'n/a' is permitted value if no postal code has been assigned
Destination_ID8	Destination facility's country	Country	S	M, if Destination_ID 1 = 1	
Transport_mode	Mode of transport by which the product leaves the facility, see: Commission Regulation (EC) No 684/2009, Annex II, Code List 7	Integer	S	M	See TransportMode
Transport_vehicle	Identification of the mode of transport (i.e. number plates, train number, plane/flight number, ship name or other identification)	Text(5000)	S	M	'n/a' is permitted value if Transport_mode = 0 and product movement takes place between adjacent facilities and is delivered manually
Transport_cont1	Indication if the transport is containerised and uses	Boolean	S	M	0 – No 1 – Yes

Dispatch of tobacco products from a facility event					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
	an individual transport unit code (e.g. SSCC)				
Transport_cont2	Individual transport unit code of the container	ITU	S	M, if Transport_cont1 = 1	
Transport_s1	Indication if the dispatch takes place with the logistic/postal operator who operates its own track and trace system accepted by the Member State of the dispatch facility. Only for small quantities of tobacco products (net weight of the products dispatched below 10 kg) destined for exports to third countries	Boolean	S	M	0 – No 1 – Yes
Transport_s2	The logistic operator's tracking number	Text(5000)	S	M, if Transport_s1 = 1	
EMCS	Dispatch under the Excise Movement and Control System (EMCS)	Boolean	S	M	0 – No 1 – Yes
EMCS_ARC	Administrative Reference Code (ARC)	ARC	S	M, if EMCS = 1	
SAAD	Dispatch with a simplified accompanying document, see: Commission Regulation (EEC) No 3649/92	Boolean	S	M	0 – No 1 – Yes
SAAD_number	Reference number of the declaration and/or authorization which has to be given by the competent authority in the Member State of destination before the movement starts	T Text(5000)	S	M, if SAAD = 1	
Exp_Declaration	Indication if the Movement Reference Number (MRN) has been issued by the customs office	Boolean	S	M	0 – No 1 – Yes
Exp_ DeclarationNumber	Movement Reference Number (MRN)	MRN	S	M, if Exp_Declaration = 1	
UI_Type	Identification of UI types in the dispatch (recorded at the highest level of available aggregation)	Integer	S	M	1 – only unit packet level UIs 2 – only aggregated level UIs 3 – both unit packet and aggregated level UIs

Dispatch of tobacco products from a facility event					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
upUIs	List of unit packet level UIs subject to the dispatch	upUI(L)	M	M, if UI_Type = 1 or 3	
aUIs	List of aggregated level UIs subject to the dispatch	aUI	M	M, if UI_Type = 2 or 3	
Dispatch_comment	Comments by the reporting entity	Text(5000)	S	O	
Information	Indicates the request of additional optional information	Boolean	S	O	0 – No 1- Yes

3.6.3.3 Business validation

	EDP (3.3)
Technical validation	
VAL_MSG_JSON	EXCISE_NUMBER_NOT_VALID
VAL_UI_MULT_MSG	upUIs, aUIs
Business rule validation	
UI creation	
VAL_UI_EXIST_UPUI	upUIs
VAL_UI_EXIST_AUI	aUIs
VAL_UI_EXIST_UPUI_SEQ	upUIs
VAL_UI_EXIST_AUI_SEQ	aUIs
Entity Validation	
VAL_ENT_EXIST_EOID	EO_ID
VAL_ENT_EXIST_FID	F_ID, Destination_ID2, Destination_ID3, Destination_ID4
VAL_ENT_ACTIVE_EOID	EO_ID
VAL_ENT_ACTIVE_FID (Router only)	F_ID, Destination_ID2, Destination_ID3, Destination_ID4
Sequence Validation	
VAL_UI_ORD_DEACTIVATED	upUIs, aUIs
VAL_UI_ORD_DISAGG	aUIs
VAL_UI_ORD_IMPLDISAGG	aUIs
VAL_UI_ORD_DISPATCH	upUIs, aUIs
Message Timing	
VAL_EVT_TIME	Event_Time

3.6.3.4 Sequence validation

The following table represents the authorized transitions for a UI and specifically the previous message for the UI.

The type of the Dispatch event 3.3 EDP refers to the Destination_ID1 field.

- Type 1 – Non EU dest.
- Type 2 – EU destination other than VM – fixed quantity delivery
- Type 3 – EU VM(s)
- Type 4 – EU destination other than VM – delivery with VV

Message Received	IRR 2.4	EUA 3.1	EUA 3.1 Import	EPA 3.2 parent UI	EPA 3.2 parent UI Import	EPA 3.2 Child	ERP 3.4	ERP 3.4 (Return)
EDP 3.3 Export (type 1)	Yes	Yes	No	Yes	No	Yes	Yes	Yes
EDP 3.3 (type 2)	Yes	Yes	No	Yes	No	Yes	Yes	Yes
EDP 3.3 VM (type 3)	Yes	Yes	No	Yes	No	Yes	Yes	Yes
EDP 3.3 VV (type 4)	Yes	Yes	No	Yes	No	Yes	Yes	Yes

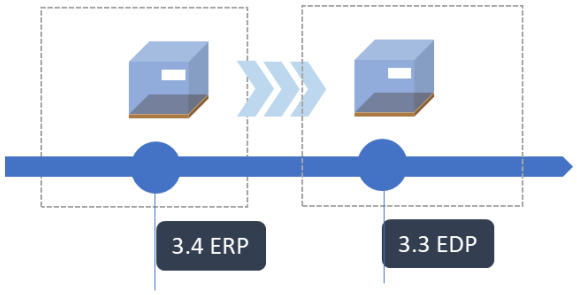
No	Next message not allowed for the UI (including different aggregatior
Yes	Next Message allowed
Yes (with Location Validation)	Next Message allowed with location validation

Products can be dispatched from a location only if they have been applied (EUA 3.1) or aggregated (EPA 3.2) in that specific location or if they have been previously reported as arrived in that location. This means that Dispatch events should follow an Arrival, an Aggregation or an Application message, and the origin of the Dispatch must correspond to the location of previous Arrival, Aggregation or Application event.

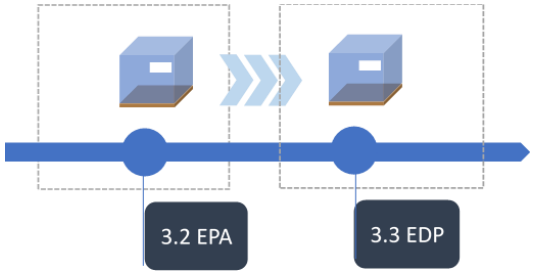
Products can be dispatched after a Reactivation event (2.4) from the location where they have been reactivated.

3.6.3.4.1 Expected sequences

- ERP – 3.4 message > EDP – 3.3 message



- EPA – 3.2 message > EDP – 3.3 message

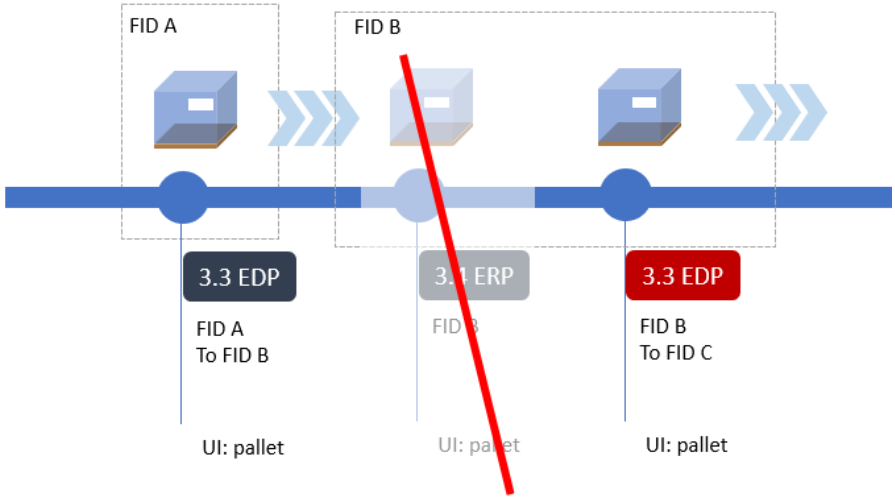


3.6.3.4.2 Example of sequence errors for EDP (3.3)

- UI_SEQUENCE_ERROR:

The error is generated when UIs scanned at Dispatch are not part of a prior Delivery/Arrival, Aggregation or Application message.
The UIs are considered as "in transit" and cannot be dispatched again.
The prior Delivery/Arrival of the UIs in the location might have not be reported or the UIs might have already been dispatched from the location.

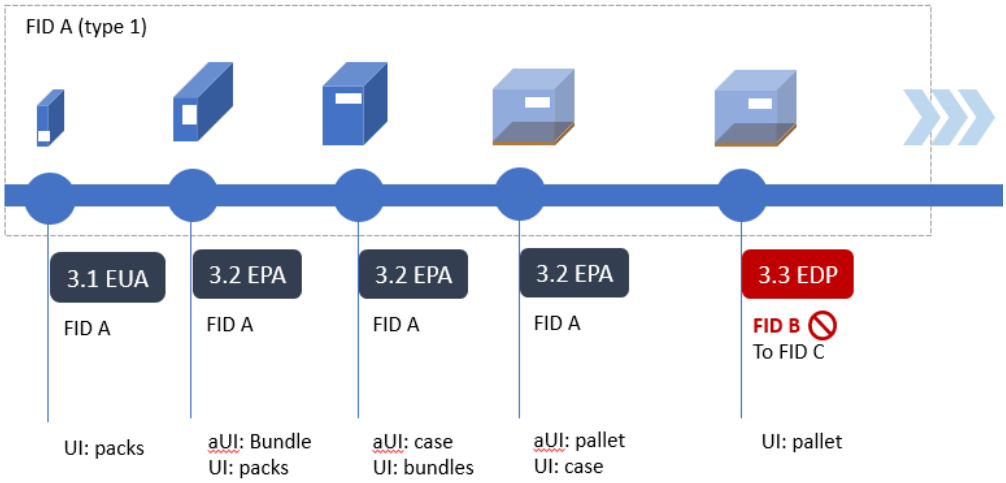
Example of EDP (3.3) with prior ERP (3.4) not reported:



- LOCATION_MISMATCH

The error is generated when the facility of origin of the Dispatch does not match the current location of the UIs reported in the Dispatch message.
There might have been a misreporting in the previous Delivery/Arrival messages leading to a wrong location of the UIs or the current Dispatch does not report the correct facility of origin.

Example of EDP (3.3) with wrong FID:



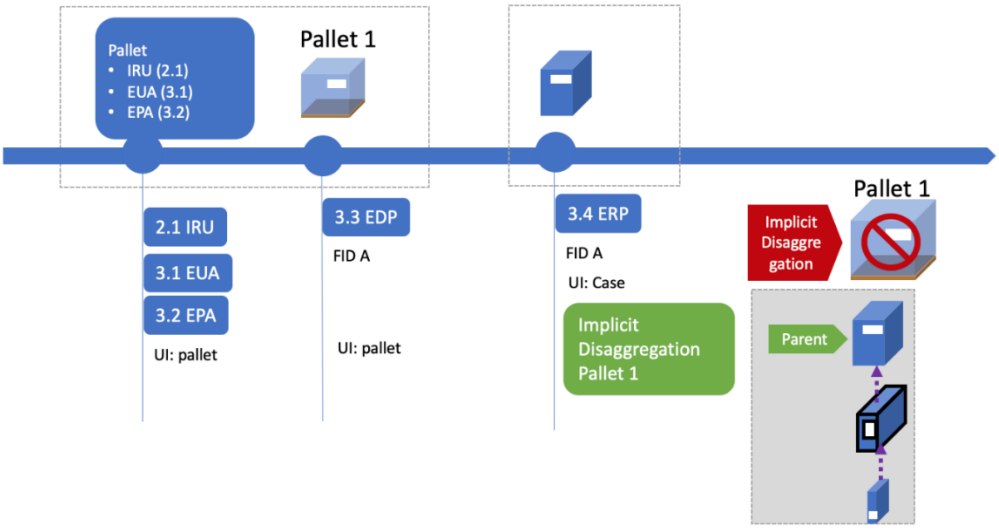
3.6.3.4.3 Import scenario

The imported goods that have been applied and aggregated require to be part of an Arrival (ERP 3.4) event.

3.6.3.5 Implicit disaggregation trigger

This event can trigger an implicit disaggregation when a child UI is identified as part of the event.

Example of an Implicit Disaggregation occurring when Dispatching



3.6.3.6 Response

Dispatch event – response					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Resp	Block of basic information elements	Component << Basic Information Response >>	S	M	Message_Type = EDP

Dispatch event – response					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
Basic Information Block	Additional optional acknowledgment Information	Component << Basic Information Block >>	S	O	

3.6.3.7 Request sample

```
{
  "EO_ID": "QCUKR+1AB020054",
  "F_ID": "QCUKR<1AB020054000049",
  "Event_Time": "19032014",
  "Message_Time_Long":"2019-03-20T14:16:45Z",
  "Destination_ID1": "1",
  "Destination_ID2": "FacilityIdB",
  "Destination_ID3": [ " FacilityIdB ", " FacilityIdB " ],
  "Destination_ID4": [ " FacilityIdB" ],
  "Destination_ID5": "FacilityIdA",
  "Destination_ID6": "Village",
  "Destination_ID7": "6358852",
  "Destination_ID8": "IT",
  "Transport_vehicle": "1",
  "Transport_cont1": 1,
  "Transport_cont2": "1",
  "Transport_s1": 1,
  "Transport_s2": "1",
  "EMCS": false,
  "EMCS_ARC": null,
  "SAAD": 1,
  "SAAD_number": 1,
  "Exp_Declaration": 1,
  "Exp_DeclarationNumber": 1,
  "UI_Type": 3,
  "upUIs": [ "DANXXXXXXXXXXXX1PR012345678919030110",
"DANXXXXXXXXXXXX2PR012345678919030110" ],
  "aUIs": [ "DANXXXXXXXXXXXX1FA00000119030110" ],
  "Dispatch_comment": "Comments",
  "Message_Type": "EDP"
}
```

3.6.3.8 Successful response sample

HTTP Status 202

```
{
    "Code": "873345b2-882f-4064-91f0-90669b46c30a",
    "Message_Type": "EDP",
    "Error": false,
    "Errors": null,
    "Checksum": "G6HF5H"
}
```

3.6.3.9 Error response sample

Processing errors

HTTP status		
<< Common response code >>		
400	EXCISE_NUMBER_NOT_VALID	VAL_FIE_FORMAT
400	MULTIPLE_UID	VAL_UI_MULT_MSG
400	UI_NOT_EXIST UI_NOT_VALID	VAL_UI_EXIST_UPUI
400	UI_NOT_EXIST	VAL_UI_EXIST_AUI
400	UI_NOT_VALID	VAL_UI_EXIST_UPUI_SEQ
400	UI_NOT_EXIST	VAL_UI_EXIST_AUI_SEQ
400	EOID_NOT_EXIST_OR_ACTIVE	VAL_ENT_EXIST_EOID
400	FID_NOT_EXIST_OR_ACTIVE	VAL_ENT_EXIST_FID
400	EOID_NOT_EXIST_OR_ACTIVE	VAL_ENT_ACTIVE_EOID
400	FID_NOT_EXIST_OR_ACTIVE	VAL_ENT_ACTIVE_FID (Router only)
400	UI_DEACTIVATED	VAL_UI_ORD_DEACTIVATED
400	UI_ALREADY_DISAGGREGATED	VAL_UI_ORD_DISAGG or VAL_UI_ORD_IMPLDISAGG
400	LOCATION_MISMATCH	VAL_UI_ORD_DISPATCH
299	SHIPMENT_WITHIN_24_HOURS	VAL_EVT_TIME
400	UI_SEQUENCE_ERROR	VAL_UI_ORD_SEQUENCE

3.6.4 ERP – (3.4) Arrival of tobacco products at a facility

3.6.4.1 Description

Record that the UIs listed in the call have been received to an economic identifier.

3.6.4.2 Description of the fields

Arrival of tobacco products at a facility					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Req	Block of basic information elements	Component << Basic Information Request >>	S	M	Message_Type = ERP
EO_ID	Economic operator identifier code of the submitting entity	EOID	S	M	
F_ID	Arrival facility identifier code	FID	S	M	
Event_Time	Time of event occurrence	Times(s)	S	M	
Message_Time_long	Message sending Time	Times(L)	S	M	
Product_Return	Indication if the arriving products are a return following complete or partial non-delivery	Boolean	S	M	0 – No 1 – Yes
UI_Type	Identification of UI types received (recorded at the highest level of available aggregation)	Integer	S	M	1 – only unit packet level UIs 2 – only aggregated level UIs

Arrival of tobacco products at a facility					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
					3 – both unit packet and aggregated level UIs
upUIs	List of unit packet level UIs received	upUI(L)	M	M, if UI_Type = 1 or 3	
aUIs	List of aggregated level UIs received	aUI	M	M, if UI_Type = 2 or 3	
Arrival_comment	Comments by the reporting entity	Text	S	O	
Information	Indicates the request of additional optional information	Boolean	S	O	0 – No 1- Yes

3.6.4.3 Business Validation

	ERP (3.4)
Technical validation	
VAL_UI_MULT_MSG	upUIs , aUIs
Business rule validation	
UI creation	
VAL_UI_EXIST_UPUI	upUIs
VAL_UI_EXIST_AUI	aUIs
VAL_UI_EXIST_UPUI_SEQ	upUIs
VAL_UI_EXIST_AUI_SEQ	aUIs
Entity Validation	
VAL_ENT_EXIST_EOID	EO_ID
VAL_ENT_EXIST_FID	F_ID
VAL_ENT_ACTIVE_EOID	EO_ID
Sequence Validation	
VAL_UI_ORD_DEACTIVATED	upUIs, aUIs
VAL_UI_ORD_DISAGG	aUIs
VAL_UI_ORD_IMPLDISAGG	aUIs
VAL_UI_ORD_ARRIVAL	upUIs, aUIs
VAL_UI_ORD_ARRIVAL_RETURN	upUIs, aUIs

3.6.4.4 Sequence validation

The following table represents the authorized transitions for a UI and specifically the previous message for the UI.

Message Received	EUA 3.1 Import	EPA 3.2 parent UI Import	EDP 3.3 (type 1) Export	EDP 3.3 (type 2)	EDP 3.3 (types 3) VM	EDP 3.3 (type 4) VV	ETL 3.5	ETL 3.5 Export	EVR 3.7
ERP 3.4	Yes	Yes	No	Yes	No	No	Yes	No	No
ERP 3.4 (Return)	No	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes

No	Next message not allowed for the UI (including different aggregatior
Yes	Next Message allowed
Yes (with Location Validation)	Next Message allowed with location validation

The type return of the ERP (3.4) is based on the Product_Return field

- 0 – No
- 1– The arrival is a type return

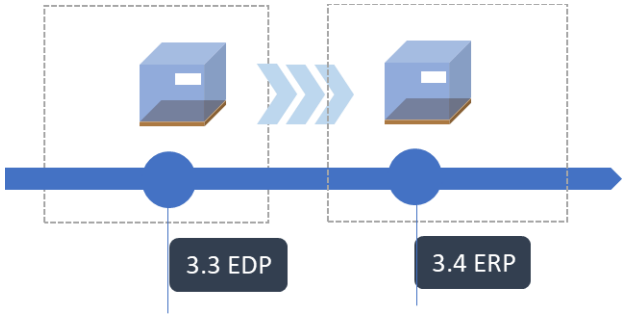
Events must be transmitted in sequence: Arrivals are expected to be reported in proper events sequence, following a dispatch, a transloading, or as a (partial) return from Vending Van delivery or from retail outlets.

3.6.4.4.1 Import scenario

In the case of Imported good, the newly applied (EUA 3.1) or aggregated (EPA 3.2) UIs must be part of an arrival event.

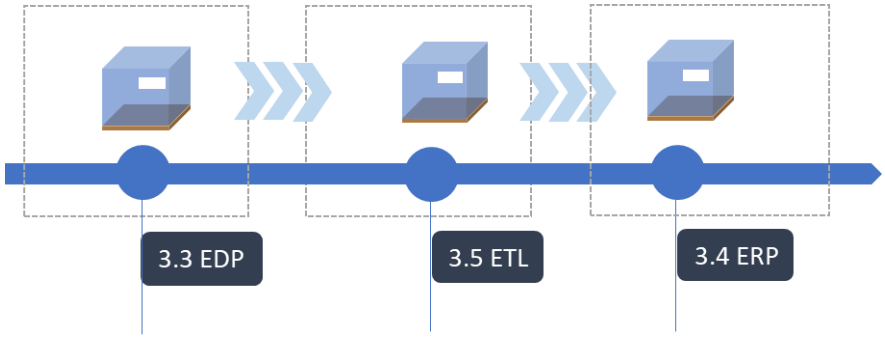
3.6.4.4.2 Arrival after Dispatch

- EDP – 3.3 (type 2) message > ERP – 3.4 message



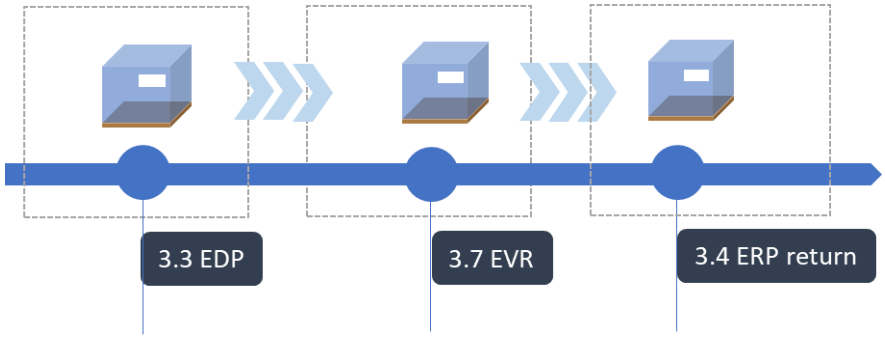
3.6.4.4.3 Arrival after Transloading

- ETL – 3.5 message > ERP – 3.4 message



3.6.4.4.4 Arrival after dispatch carried out by vending van

- EVR – 3.7 messages > ERP of type return – 3.4 message



Events must be reported within respecting Principle 4: All disaggregation must be performed at a location. No Disaggregation are allowed during the transport.
Events must be reported within respecting Principle 5: The reporting on the Arrival should be done on the same UI that have been reported during the Dispatch/Transloading process. This is a consequence of Principle 4. This means that an Arrival Event that contains child UI of UI reported during the Dispatch/Transloading Event will be rejected. The same UI must be reported.
Exception to principle 5: Arrival of type return can be reported at a different level than the previous dispatch/transloading/delivery with VV

3.6.4.4.5 Arrival of type return

The Arrival of type return is the proper reporting event for exported goods, goods in transit (dispatched or in transloading)

3.6.4.4.6 Examples of sequence errors for ERP (3.4)

ARRIVAL_NOTALLOWED
The error is generated because UIs in the Arrival message were not part of a previous EDP (3.3) or ETL (3.5) message. The previous Economic Operator should be contacted to verify its message of Dispatch/Transloading.
Note: in case of product return, the ERP (3.4) can be reported after an EVR (3.7), this will not trigger ARRIVAL_NOTALLOWED.

Example of ERP (3.4) rejected because the previous EDP (3.3) has not been reported:

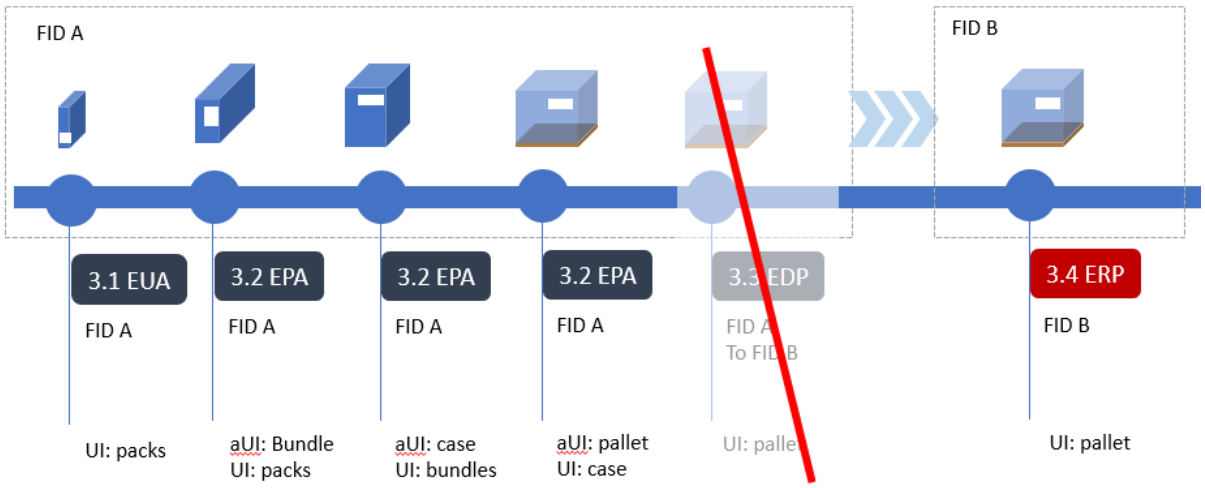


Figure 8 Arrival sequence error ARRIVAL_NOTALLOWED

UI_SEQUENCE_ERROR
The error is generated when UIs scanned at Arrival are not the same that the ones scanned in the prior Dispatch/Transloading event.
According to principle 5, the UIs must be scanned at Arrival at the highest level of aggregation, before being implicitly or explicitly disaggregated.
If the error is generated when Arrival is reported at the highest available aggregation level, the previous Economic Operator should be contacted to verify at which level it reported Dispatch/Transloading.

Exception: Arrivals of type return will not generate this error because implicit disaggregation is allowed for them

Example of ERP (3.4) rejected because it is not reported at the right level of aggregation:

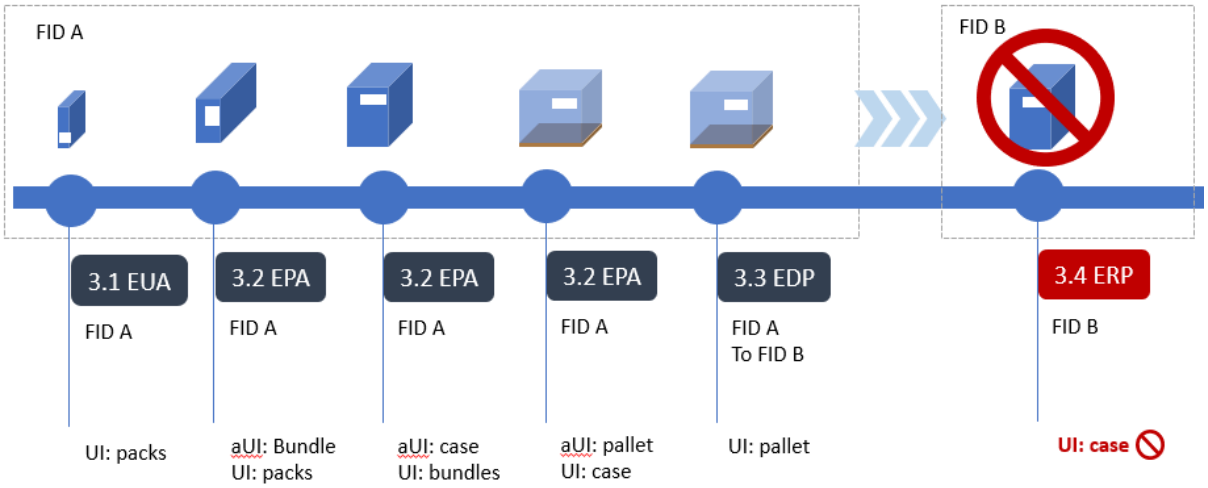


Figure 9 Arrival sequence error UI_SEQUENCE_ERROR

3.6.4.5 Implicit disaggregation trigger

The Arrival of type Return (Product_Return = "true") ERP (3.4) can trigger an implicit disaggregation when a child UI is identified as part of the event.

Note that an arrival message that contains a child UI will cause a sequence validation error UI_SEQUENCE_ERROR if the Product_Return flag is set to false.

1.1.1.6 Response

Arrival of tobacco products at a facility– response					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Resp	Block of basic information elements	Component << Basic Information Response >>	S	M	Message_Type = ERP
Basic Information Block	Additional optional acknowledgment Information	Component << Basic Information Block >>	S	O	

3.6.4.5 Request sample

```
{
  "EO_ID": "QCUKR+1AB020054",
  "F_ID": "QCUKR<1AB020054000049",
  "Event_Time": "19032014",
  "Message_Time_Long":"2019-03-20T14:16:45Z",
  "Product_Return": "true",
  "UI_Type": "1",
  "upUIs": [ "DANXXXXXXXXXXXX1PR012345678919030110", "
DANXXXXXXXXXXX2PR012345678919030110" ],
  "aUIs": [ "DANXXXXXXXXXXXX1PR012345678919030110" ],
  "Arrival_comment": "Comments",
  "Message_Type": "ERP",
  "Code": "873345b2-882f-4064-91f0-90669b46c30a"
}
```

3.6.4.6 Successful response sample

HTTP Status 202

```
{
    "Code": "873345b2-882f-4064-91f0-90669b46c30a",
    "Message_Type": "ERP",
    "Error": false,
    "Errors": null,
    "Checksum": "G6HF5H"
}
```

3.6.4.7 Error response sample

HTTP status	Error Code	Error Description
<< Common response code >>		
400	MULTIPLE_UID	VAL_UI_MULT_MSG
400	UI_NOT_EXIST UI_NOT_VALID	VAL_UI_EXIST_UPUI
400	UI_NOT_EXIST	VAL_UI_EXIST_AUI
400	UI_NOT_VALID	VAL_UI_EXIST_UPUI_SEQ
400	UI_NOT_EXIST	VAL_UI_EXIST_AUI_SEQ
400	EOID_NOT_EXIST_OR_ACTIVE	VAL_ENT_EXIST_EOID
400	FID_NOT_EXIST_OR_ACTIVE	VAL_ENT_EXIST_FID
400	EOID_NOT_EXIST_OR_ACTIVE	VAL_ENT_ACTIVE_EOID
400	UI_DEACTIVATED	VAL_UI_ORD_DEACTIVATED
400	UI_ALREADY_DISAGGREGATED	VAL_UI_ORD_DISAGG or VAL_UI_ORD_IMPLDISAGG
400	ARRIVAL_NOTALLOWED	VAL_UI_ORD_ARRIVAL or VAL_UI_ORD_ARRIVAL_RETURN
400	UI_SEQUENCE_ERROR	VAL_UI_ORD_SEQUENCE

3.6.5 ETL – (3.5) Trans-loading

3.6.5.1 Description

Event to show that UIs have been moved from one transport mechanism to another.

3.6.5.2 Description of the fields

Trans-loading event					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Req	Block of basic information elements	Component << Basic Information Request >>	S	M	Message_Type = ETL
EO_ID	Economic operator identifier code of the submitting entity	EOID	S	M	
Event_Time	Intended time of event occurrence	Time(s)	S	M	
Message_Time_long	Message sending Time	Time(L)	S	M	
Destination_ID1	Indication if the destination facility is located on the EU territory	Integer	S	M	0 – No 1 – Yes
Destination_ID2	Destination facility identifier code	FID	S	M, if Destination_ID 1 = 1	
Destination_ID3	Destination facility's street name and house number (or road number and kilometer)	Text(300)	S	M, if Destination_ID 1 = 0	

Trans-loading event					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
Destination_ID4	Destination facility's municipality (city, town or village)	Text(100)	S	M, if Destination_ID 1 = 0	
Destination_ID5	Destination facility's postal code	Text(50)	S	M, if Destination_ID 1 = 0	'n/a' is permitted value if no postal code has been assigned
Destination_ID6	Destination facility's country	Country	S	M, if Destination_ID 1 = 0	
Transport_mode	Mode of transport to which the product is trans-loaded, see: Commission Regulation (EC) No 684/2009, Annex II, Code List 7	Integer	S	M	See TransportMode
Transport_vehicle	Identification of the vehicle (i.e. number plates, train number, plane/flight number, ship name or other identification)	Text(5000)	S	M	
Transport_cont1	Indication if the transport is containerised and uses an individual transport unit code (e.g. SSCC)	Boolean	S	M	0 – No 1 – Yes
Transport_cont2	Individual transport unit code of the container	ITU	S	M, if Transport_cont1 = 1	
EMCS	Dispatch under the Excise Movement and Control System (EMCS)	Boolean	S	M	0 – No 1 – Yes
EMCS_ARC	Administrative Reference Code (ARC)	ARC	S	M, if EMCS = 1	
UI_Type	Identification of UI types subject to the trans-loading (recorded at the highest level of available aggregation)	Integer	S	M	1 – only unit packet level UIs 2 – only aggregated level UIs 3 – both unit packet and aggregated level UIs
upUIs	List of unit packet level UIs subject to the trans-loading	upUI(L)	M	M, if UI_Type = 1 or 3	
aUIs	List of aggregated level UIs subject to the trans-loading	aUI	M	M, if UI_Type = 2 or 3	
Transloading_comment	Comments by the reporting entity	Text(5000)	S	O	
Information	Indicates the request of additional optional information	Boolean	S	O	0 – No 1- Yes

3.6.5.3 Business validation

	ETL (3.5)
Technical validation	
VAL_UI_MULT_MSG	upUIs, aUIs
Business rule validation	
UI creation	
VAL_UI_EXIST_UPUI	upUIs
VAL_UI_EXIST_AUI	aUIs
VAL_UI_EXIST_UPUI_SEQ	upUIs
VAL_UI_EXIST_AUI_SEQ	aUIs
Entity Validation	
VAL_ENT_EXIST_EOID	EO_ID
VAL_ENT_EXIST_FID	Destination_ID2
VAL_ENT_ACTIVE_EOID	EO_ID
Sequence Validation	
VAL_UI_ORD_DEACTIVATED	upUIs, aUIs
VAL_UI_ORD_DISAGG	aUIs
VAL_UI_ORD_IMPLDISAGG	aUIs
Message Timing	
VAL_EVT_TIME	Event_Time

3.6.5.4 Sequence validation

The field Destination_ID1 in the ETL event indicates if the ETL is aimed at export or EU location.

- 0 – No
- 1 – Yes

Message Received	EDP 3.3 (type 1) Export	EDP 3.3 (type 2)	EDP 3.3 (types 3) VM	EDP 3.3 (type 4) VV	ETL 3.5	ETL 3.5 Export
ETL 3.5	No	Yes	No	No	Yes	No
ETL 3.5 (Export)	Yes	No	No	No	No	Yes

No	Next message not allowed for the UI (including different aggregatior
Yes	Next Message allowed
Yes (with Location Validation)	Next Message allowed with location validation

ETL (3.5) can only be preceded by EDP (3.3) of type 1 or 2 or another ETL (3.5)
The ETL (3.5) event is not subject to any location validation

3.6.5.5 Response

Trans-loading event – response					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Resp	Block of basic information elements	Component << Basic Information Response >>	S	M	Message_Type = ETL
Basic Information Block	Additional optional acknowledgment Information	Component << Basic Information Block >>	S	O	

3.6.5.6 Request sample

```
{
  "EO_ID": "QCUKR+1AB020054",
  "Event_Time": "19032014",
  "Message_Time_Long":"2019-03-20T14:16:45Z",
  "Destination_ID1": 1,
  "Destination_ID2": "FGHZ7G",
  "Destination_ID3": "Address",
  "Destination_ID4": "TownA",
  "Destination_ID5": "12345",
  "Destination_ID6": "DE",
  "Transport_mode": 1,
  "Transport_vehicle": 1,
  "Transport_cont1": 1,
  "Transport_cont2": "code",
  "EMCS": 1,
  "EMCS_ARC": "ref",
  "UI_Type": 1,
  "upUIs": [ "DANXXXXXXXXXXXX1PR012345678919030110", "
DANXXXXXXXXXXXX2PR012345678919030110" ],
  "aUIs": [ "DANXXXXXXXXXXX10FA00000119030110" ],
  "Transloading_comment": "Comments",
  "Message_Type": "ETL",
  "Code": "873345b2-882f-4064-91f0-90669b46c30a"
}
```

3.6.5.7 Successful response sample

HTTP Status 202

```
{
   "Code": "873345b2-882f-4064-91f0-90669b46c30a",
    "Message_Type": "ETL",
    "Error": false,
    "Errors": null,
    "Checksum": "G6HF5H"
}
```

3.6.5.8 Error response sample

HTTP status		
<< Common response code >>		
400	MULTIPLE_UID	VAL_UI_MULT_MSG
400	UI_NOT_EXIST UI_NOT_VALID	VAL_UI_EXIST_UPUI
400	UI_NOT_EXIST	VAL_UI_EXIST_AUI
400	UI_NOT_VALID	VAL_UI_EXIST_UPUI_SEQ
400	UI_NOT_EXIST	VAL_UI_EXIST_AUI_SEQ
400	EOID_NOT_EXIST_OR_ACTIVE	VAL_ENT_EXIST_EOID
400	FID_NOT_EXIST_OR_ACTIVE	VAL_ENT_EXIST_FID
400	EOID_NOT_EXIST_OR_ACTIVE	VAL_ENT_ACTIVE_EOID
400	UI_DEACTIVATED	VAL_UI_ORD_DEACTIVATED
400	UI_ALREADY_DISAGGREGATED	VAL_UI_ORD_DISAGG or VAL_UI_ORD_IMPLDISAGG
299	SHIPMENT_WITHIN_24_HOURS	VAL_EVT_TIME

3.6.6 EUD – (3.6) Disaggregation of aggregated level UIs

3.6.6.1 Description

Event showing that an aggregation no longer exists.

3.6.6.2 Description of the fields

aUI disaggregation event					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Req	Block of basic information elements	Component << Basic Information Request >>	S	M	Message_Type = EUD
EO_ID	Economic operator's identifier	EOID	S	M	
F_ID	Facility's identifier	FID	S	M	
Event_Time	Time of event occurrence	Time(s)	S	M	
Message_Time_long	Message sending Time	Time(L)	S	M	
aUI	Aggregated level UI subject to disaggregation	aUI	S	M	
disaUI_comment	Comments by the reporting entity	Text(5000)	S	O	
Information	Indicates the request of additional optional information (only available on the Router interface)	Boolean	S	O	0 – No 1- Yes

3.6.6.3 Business validation

	EUD (3.6)
Business rule validation	
UI creation	
VAL_UI_EXIST_AUI	aUI
VAL_UI_EXIST_AUI_SEQ	aUI
Entity Validation	
VAL_ENT_EXIST_EOID	EO_ID
VAL_ENT_EXIST_FID	F_ID
VAL_ENT_ACTIVE_EOID	EO_ID
Sequence Validation	
VAL_UI_ORD_DEACTIVATED	aUI
VAL_UI_ORD_AGG_FID	aUI (ONLY for aUI that have not been implicitly disaggregated)

3.6.6.4 Sequence validation

The following table represents the authorized transitions for a UI and specifically the previous message for the UI.

Message Received	IRR 2.4 aUI	EPA 3.2 parent UI	EPA 3.2 parent UI Import	EPA 3.2 Child	EPA 3.2 Child import	ERP 3.4	ERP 3.4 (Return)	EUD 3.6	EUD 3.6 (aUI implicitly disaggregated) - reuse of aUI
EUD 3.6	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes

No	Next message not allowed for the UI (including different aggregatior
Yes	Next Message allowed
Yes (with Location Validation)	Next Message allowed with location validation

3.6.6.4.1

3.6.6.5 Clarification of the location validation

The disaggregation event is subject to the location validation (VAL_UI_ORD_AGG_FID). In other words, the aUI that are been disaggregated must be in the same facility.
There is exception for aUI that have been already disaggregated implicitly.

3.6.6.6 Location validation Exception

Location validation exception for Imports Due to the fact that product movement outside of the EU are not subject to being reported into the EU tobacco track and trace system (i.e. Dispatch/Arrivals), in the event that an Economic Operator requires to modify the hierarchy of the goods before the import into EU, there will be Aggregation (3.2 – EPA) and Disaggregation (3.6 – EUD) messages on different locations (Facility IDs) without a Dispatch/Arrival movement between them. For these cases, the Location validation (VAL_UI_ORG_AGG_FID) will not be applicable (so all products BEFORE the first Arrival 3.4 into the EU may be modified in terms of their logistic hierarchy).

3.6.6.7 Clarification on the impact of Implicit disaggregation

The reporting of a disaggregation event (EUD 3.6) on an aUI that have been previously implicitly disaggregated (by any the reporting of an event that allows the triggering of the implicit disaggregation mechanism) doesn't correspond to any physical movement. Therefore, the location validation control VAL_UI_ORD_AGG_FID will not be applied.

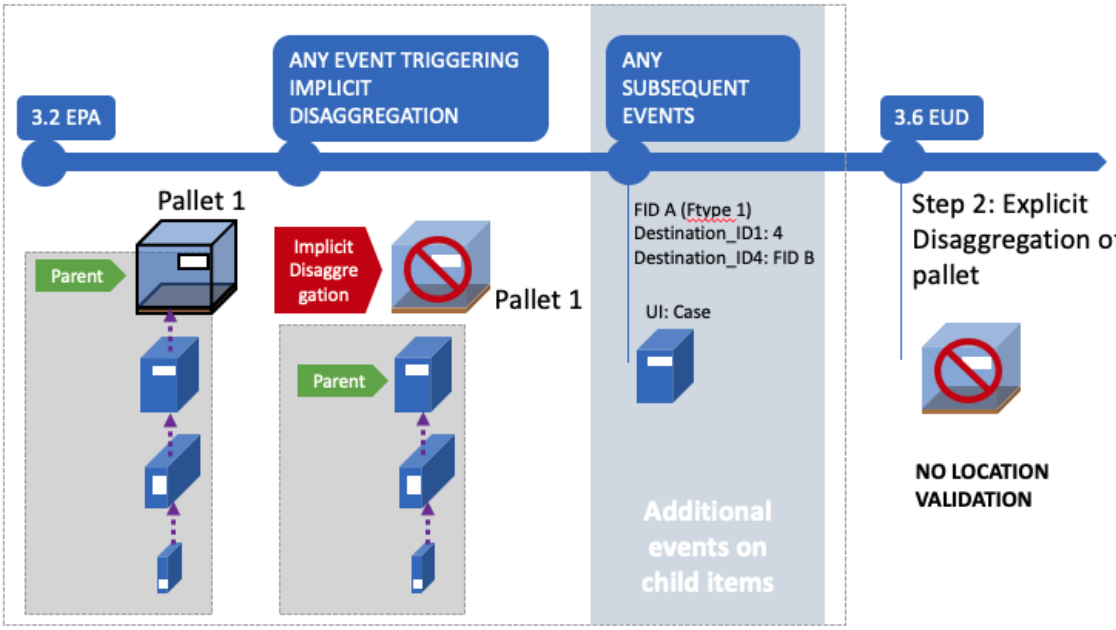


Figure 10 reporting of disaggregation event on aUI that are implicitly disaggregated.

3.6.6.8 Implicit disaggregation trigger

This event can trigger an implicit disaggregation when a child UI is identified as part of the event.

3.6.6.9 Response

aUI disaggregation event– response					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Resp	Block of basic information elements	Component << Basic Information Response >>	S	M	Message_Type = EUD

3.6.6.10 Disaggregation Acknowledgement Confirmation

The ” Information” field should be set to 1. In the response messages the Information block contains the Data_List field that holds the list of “subordinate/child” UIs.

Note: The response is only available on the Router.

3.6.6.11 Request sample

```
{
  "EO_ID": "QCUKR+1AB020054",
  "F_ID": "QCUKR<1AB020054000049",
  "Event_Time": "19032014",
  "Message_Time_Long":"2019-03-20T14:16:45Z",
  "aUI": "DANXXXXXXXXXX10FA00000119030110",
  "disaUI_comment": "Comments",
  "Message_Type": "EUD",
  "Code": "873345b2-882f-4064-91f0-90669b46c30a"
}
```

3.6.6.12 Successful response sample

HTTP Status 202

```
{
   "Code": "873345b2-882f-4064-91f0-90669b46c30a",
    "Message_Type": "EUD",
    "Error": false,
    "Errors": null,
    "Checksum": "G6HF5H"
}
```

3.6.6.13 Error response sample

HTTP status	Error Code	Error Description
<< Common response code >>		
400	UI_NOT_EXIST	VAL_UI_EXIST_AUI
400	UI_NOT_VALID	VAL_UI_EXIST_AUI_SEQ
400	EOID_NOT_EXIST_OR_ACTIVE	VAL_ENT_EXIST_EOID
400	FID_NOT_EXIST_OR_ACTIVE	VAL_ENT_EXIST_FID
400	EOID_NOT_EXIST_OR_ACTIVE	VAL_ENT_ACTIVE_EOID
400	UI_DEACTIVATED	VAL_UI_ORD_DEACTIVATED
400	LOCATION_MISMATCH	VAL_UI_ORD_AGG_FID

3.6.7 EVR – (3.7) Report the delivery carried out with a vending van to retail outlet

3.6.7.1 Description

Event sent when UIs have been distributed via a van delivery.

3.6.7.2 Description of the fields

Vending Van event					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Req	Block of basic information elements	Component << Basic Information Request >>	S	M	Message_Type = EVR
EO_ID	Economic operator identifier code of the submitting entity	EOID	S	M	
F_ID	Facility identifier code of retail outlet	FID	S	M	
Event_Time	Time of event occurrence	Time(s)	S	M	
Message_Time_long	Message sending Time	Time(L)	S	M	
UI_Type	Identification of UI types delivered (recorded at the highest level of available aggregation)	Integer	S	M	1 – only unit packet level UIs 2 – only aggregated level UIs 3 – both unit packet and aggregated level UIs
upUIs	List of unit packet level UIs delivered	upUI(L)	M	M, if UI_Type = 1 or 3	

aUIs	List of aggregated level UIs delivered	aUI	M	M, if UI_Type = 2 or 3	
Delivery_comment	Comments by the reporting entity	Text(5000)	S	O	
Information	Indicates the request of additional optional information	Boolean	S	O	0 – No 1- Yes

3.6.7.3 Business validation

	EVR (3.7)
Technical validation	
VAL_UI_MULT_MSG	upUIs, aUIs
Business rule validation	
UI creation	
VAL_UI_EXIST_UPUI	upUIs
VAL_UI_EXIST_AUI	aUIs
VAL_UI_EXIST_UPUI_SEQ	upUIs
VAL_UI_EXIST_AUI_SEQ	aUIs
Entity Validation	
VAL_ENT_EXIST_EOID	EO_ID
VAL_ENT_EXIST_FID	F_ID
VAL_ENT_ACTIVE_EOID	EO_ID
Sequence Validation	
VAL_UI_ORD_DEACTIVATED	upUIs, aUIs
VAL_UI_ORD_DISAGG	aUIs
VAL_UI_ORD_IMPLDISAGG	aUIs
Message Timing	
VAL_EVT_24H	Event_Time

3.6.7.4 Sequence Validation

The following table represents the authorized transitions for a UI and specifically the previous message for the UI.

Message Received	EDP 3.3 (type 1) Export	EDP 3.3 (type 2)	EDP 3.3 (types 3) VM	EDP 3.3 (type 4) VV
EVR 3.7	No	No	No	Yes

No	Next message not allowed for the UI (including different aggregatior
Yes	Next Message allowed
Yes (with Location Validation)	Next Message allowed with location validation

For each UI in the event, An EVR must be preceded by a Dispatch event (EDP 3.3) of type 4. Any other combination will result in a Sequence Error

Note that the omission of the reporting of the Dispatch event (EDP 3.3) will result in a SEQUENCE_ERROR

3.6.7.5 Implicit disaggregation trigger

This event can trigger an implicit disaggregation when a child UI is identified as part of the event. Implicit Disaggregation occurring when Delivering to Retail Outlet (EVR 3.7)

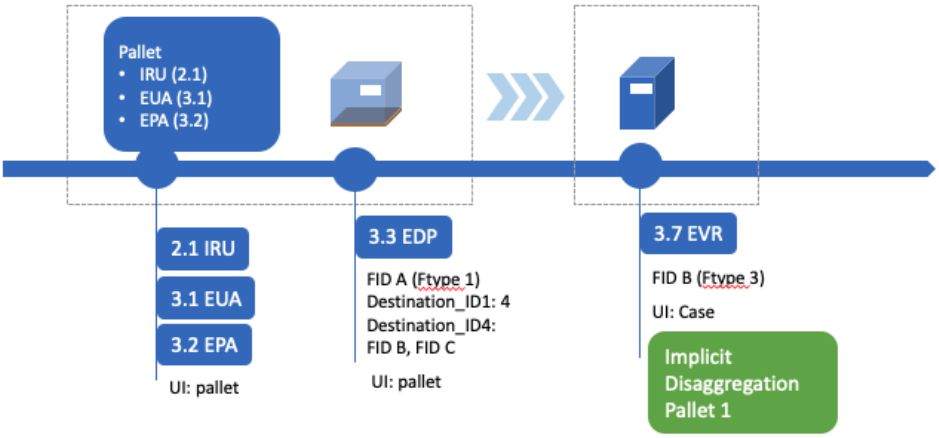


Figure 11 Implicit disaggregation trigger

3.6.7.6 Response

Vending Van event – response					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Resp	Block of basic information elements	Component << Basic Information Response >>	S	M	Message_Type = EVR
Basic Information Block	Additional optional acknowledgment Information	Component << Basic Information Block >>	S	O	

3.6.7.7 Request sample

```
{
  "EO_ID": "QCUKR+1AB020054",
  "F_ID": "QCUKR<1AB020054000049",
  "Event_Time": "19032014",
  "Message_Time_Long":"2019-03-20T14:16:45Z",
  "UI_Type": 1,
  "upUIs": [ "DANXXXXXXXXXXXX1PR012345678919030110", "
DANXXXXXXXXXXXX2PR012345678919030110" ],
  "aUIs": [ "DANXXXXXXXXXXX10FA00000119030110" ],
  "Delivery_comment": "Comments",
  "Message_Type": "EVR",
  "Code": "873345b2-882f-4064-91f0-90669b46c30a"
}
```

3.6.7.8 Successful response sample

HTTP Status 202

```
{
   "Code": "873345b2-882f-4064-91f0-90669b46c30a",
    "Message_Type": "EVR",
    "Error": false,
    "Errors": null,
    "Checksum": "G6HF5H"
}
```

3.6.7.9 Error response sample

HTTP status		
<< Common response code >>		
400	MULTIPLE_UID	VAL_UI_MULT_MSG
400	UI_NOT_EXIST UI_NOT_VALID	VAL_UI_EXIST_UPUI
400	UI_NOT_EXIST	VAL_UI_EXIST_AUI
400	UI_NOT_VALID	VAL_UI_EXIST_UPUI_SEQ
400	UI_NOT_EXIST	VAL_UI_EXIST_AUI_SEQ
400	EOID_NOT_EXIST_OR_ACTIVE	VAL_ENT_EXIST_EOID
400	FID_NOT_EXIST_OR_ACTIVE	VAL_ENT_EXIST_FID
400	EOID_NOT_EXIST_OR_ACTIVE	VAL_ENT_ACTIVE_EOID
400	UI_DEACTIVATED	VAL_UI_ORD_DEACTIVATED
400	UI_ALREADY_DISAGGREGATED	VAL_UI_ORD_DISAGG or VAL_UI_ORD_IMPLDISAGG
299	SHIPMENT_WITHIN_24_HOURS	VAL_EVT_24H

3.6.8 EDX – (3.8) Dispatch of tobacco products from a facility to laboratories, waste disposal centres, national authorities, international governmental organisations, embassies and military bases

3.6.8.1 Description

Record that the UIs listed in the call have been dispatched from the economic identifier to a special destinations.

3.6.8.2 Description of the fields

Dispatch of tobacco products from a facility event					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Req	Block of basic information elements	Component << Basic Information Request >>	S	M	Message_Type = EDX
EO_ID	Economic operator identifier code of the submitting entity	EOID	S	M	

Dispatch of tobacco products from a facility event					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
Event_Time	Time of event occurrence	Time (s)	S	M	
Message_Time_long	Message sending Time	Time (L)	S	M	
F_ID	Dispatch facility identifier code	FID	S	M	
Destination_1	Indication of destination type	Integer	S	M	1 – Laboratory 2 – Waste disposal centre 3 – National authority 4 – International governmental organisation 5 - Embassy 6 – Military base
Destination_2	Destination's street name and house number (or road number and kilometer)	Text(300)	S	M	
Destination_3	Destination's municipality (city, town or village)	Text(100)	S	M	
Destination_4	Destination's postal code	Text(50)	S	M	'n/a' is permitted value if no postal code has been assigned
Destination_5	Destination's country	Country	S	M	
Transport_mode	Mode of transport by which the product leaves the facility, see Annex II, Code List 7 of Commission Regulation (EC) No 684/20091	Integer	S	M	0 – Other 1 – Sea Transport 2 – Rail transport 3 – Road transport 4 – Air transport 5 – Postal consignment 6 – Fixed transport installations 7 – Inland waterway transport
Transport_vehicle	Identification of the vehicle (i.e. number plates, train number, plane/flight number, ship name or other identification)	Text(5000)	S	M	'n/a' is permitted value if Transport_mode = 0 and product movement takes place between adjacent facilities and is

Dispatch of tobacco products from a facility event					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
					delivered manually
Transport_cont 1	Indication if the transport is containerised and uses an individual transport unit code (e.g. SSCC)	Boolean	S	M	0 – No 1 – Yes
Transport_cont 2	Individual transport unit code of the container	ITU	S	M, if Transport_co nt1 = 1	
UI_Type	Identification of UI types in the dispatch (recorded at the highest level of available aggregation)	Integer	S	M	1 – only unit packet level UIs 2 – only aggregated level UIs 3 – both unit packet and aggregated level UIs
upUIs	List of unit packet level UIs subject to the dispatch	upUI(L)	M	M, if UI_Type = 1 or 3	
aUIs	List of aggregated level UIs subject to the dispatch	aUI	M	M, if UI_Type = 2 or 3	
S_Dispatch_co mment	Comments by the reporting entity	Text(5000)	S	O	
Information	Indicates the request of additional optional information	Boolean	S	O	0 – No 1- Yes

3.6.8.3 Business validation

	EDX (3.8)
Technical validation	
VAL_MSG_JSON	EXCISE_NUMBER_NOT_VALID
VAL_UI_MULT_MSG	upUIs, aUIs
Business rule validation	
UI creation	
VAL_UI_EXIST_UPUI	upUIs
VAL_UI_EXIST_AUI	aUIs
VAL_UI_EXIST_UPUI_SEQ	upUIs
VAL_UI_EXIST_AUI_SEQ	aUIs
Entity Validation	
VAL_ENT_EXIST_EOID	EO_ID
VAL_ENT_EXIST_FID	F_ID
VAL_ENT_ACTIVE_EOID	EO_ID
VAL_ENT_ACTIVE_FID (Router only)	F_ID
Sequence Validation	
VAL_UI_ORD_DEACTIVATED	upUIs, aUIs
VAL_UI_ORD_DISAGG	aUIs
VAL_UI_ORD_IMPLDISAGG	aUIs
VAL_UI_ORD_DISPATCH	upUIs, aUIs
Message Timing	
VAL_EVT_TIME	Event_Time

3.6.8.4 Sequence validation

The EDX message shares very similar validations as the EDP (3.3) message.

The most significant exception is that it is not possible to declare ETL (3.5 Transloading) after an EDX 3.8. The only valid product movement events after an EDX are an ERP Return (Arrival of type return) or IDA (deactivation).

The following table represents the authorized transitions for a UI and specifically the previous message for the UI.

Message Received	IRR 2.4	EUA 3.1	EUA 3.1 Import	EPA 3.2 parent UI	EPA 3.2 parent UI Import	EPA 3.2 Child	ERP 3.4	ERP 3.4 (Return)
EDX 3.8	Yes	Yes	No	Yes	No	Yes	Yes	Yes

No	Next message not allowed for the UI (including different aggregatior
Yes	Next Message allowed
Yes (with Location Validation)	Next Message allowed with location validation

Products can be dispatched from a location only if they have been applied (EUA 3.1) or aggregated (EPA 3.2) in that specific location or if they have been previously reported as arrived in that location. This means that Dispatch events should follow an Arrival, an Aggregation or an Application message, and the origin of the Dispatch must correspond to the location of previous Arrival, Aggregation or Application event.

3.6.8.5 Implicit disaggregation trigger

This event can trigger an implicit disaggregation when a child UI is identified as part of the event.

3.6.8.6 Response

Dispatch event – response					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Resp	Block of basic information elements	Component << Basic Information Response >>	S	M	Message_Type = EDX

Dispatch event – response					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
Basic Information Block	Additional optional acknowledgment Information	Component << Basic Information Block >>	S	O	

3.6.8.7 Request sample

```
{
  "EO_ID": "QCUKR+1AB020054",
  "F_ID": "QCUKR<1AB020054000049",
  "Event_Time": "19032014",
  "Message_Time_Long":"2019-03-20T14:16:45Z",
  "Destination_1": "1",
  "Destination_2": "Street A",
  "Destination_3": "City A",
  "Destination_4": "n/a",
  "Destination_5": "BE",
  "Transport_mode": "3",
  "Transport_vehicle": "AB12345AB",
  "Transport_cont1": 1,
  "Transport_cont2": "1",
  "UI_Type": 3,
  "upUIs": [ "DANXXXXXXXXXXXX1PR012345678919030110",
"DANXXXXXXXXXXXX2PR012345678919030110" ],
  "aUIs": [ "DANXXXXXXXXXXXX1FA00000119030110" ],
  "S_Dispatch_comment": "Comments",
  "Message_Type": " EDX"
}
```

3.6.8.8 Successful response sample

HTTP Status 202

```
{
    "Code": "873345b2-882f-4064-91f0-90669b46c30a",
    "Message_Type": "EDX",
    "Error": false,
    "Errors": null,
    "Checksum": "G6HF5H"
}
```

3.6.8.9 Error response sample

Processing errors

HTTP status		
<< Common response code >>		
400	EXCISE_NUMBER_NOT_VALID	VAL_FIE_FORMAT
400	MULTIPLE_UID	VAL_UI_MULT_MSG
400	UI_NOT_EXIST UI_NOT_VALID	VAL_UI_EXIST_UPUI
400	UI_NOT_EXIST	VAL_UI_EXIST_AUI
400	UI_NOT_VALID	VAL_UI_EXIST_UPUI_SEQ
400	UI_NOT_EXIST	VAL_UI_EXIST_AUI_SEQ
400	EOID_NOT_EXIST_OR_ACTIVE	VAL_ENT_EXIST_EOID
400	FID_NOT_EXIST_OR_ACTIVE	VAL_ENT_EXIST_FID

400	EOID_NOT_EXIST_OR_ACTIVE	VAL_ENT_ACTIVE_EOID
400	FID_NOT_EXIST_OR_ACTIVE	VAL_ENT_ACTIVE_FID (Router only)
400	UI_DEACTIVATED	VAL_UI_ORD_DEACTIVATED
400	UI_ALREADY_DISAGGREGATED	VAL_UI_ORD_DISAGG or VAL_UI_ORD_IMPLDISAGG
400	LOCATION_MISMATCH	VAL_UI_ORD_DISPATCH
299	SHIPMENT_WITHIN_24_HOURS	VAL_EVT_TIME
400	UI_SEQUENCE_ERROR	VAL_UI_ORD_SEQUENCE

3.7 Reporting transactional events (trade information)

3.7.1 EIV – (4.1) Issuing of the invoice

3.7.1.1 Description

Added invoice details to a UI.

3.7.1.2 Description of the fields

Invoice reporting					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Req	Block of basic information elements	Component << Basic Information Request >>	S	M	Message_Type = EIV
EO_ID	Economic operator identifier code of the submitting entity	EOID	S	M	
Event_Time	Time of event occurrence	Time(s)	S	M	
Message_Time_long	Message sending Time	Time(L)	S	M	
Invoice_Type1	Type of the invoice	Integer	S	M	See InvoiceType
Invoice_Type2	Description of the other type of the invoice	Text(5000)	S	M, if Invoice_Type1 = 3	
Invoice_Number	Number of the invoice	Text(5000)	S	M	
Invoice_Date	Date of the invoice	Date	S	M	
Invoice_Seller	Identity of the seller	EOID	S	M	
Invoice_Buyer1	Identification if the buyer is located in the EU	Boolean	S	M	0 – No 1 – Yes
Invoice_Buyer2	Identity of the buyer	EOID	S	M, if Invoice_Buyer1 = 1	
Buyer_Name	Buyer's registered legal name	Text(5000)	S	M, if Invoice_Buyer1 = 0	
Buyer_Address_1	Buyer's street name and house number (or road number and kilometer)	Text(300)	S	M, if Invoice_Buyer1 = 0	
Buyer_Address_2	Buyer's municipality (city, town or village)	Text(100)	S	M, if Invoice_Buyer1 = 0	

Invoice reporting					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
Buyer_Address_3	Buyer's postal code	Text(50)	S	M, if Invoice_Buyer1 = 0	'n/a' is permitted value if no postal code has been assigned
Buyer_CountryReg	Buyer's country of registration	Country	S	M, if Invoice_Buyer1 = 0	
Buyer_TAX_N	Buyer's tax registration number	Text(5000)	S	M, if Invoice_Buyer1 = 0	
First_Seller_EU	Identification if the invoice is issued by the first seller in the EU, i.e. the EU manufacturer or the importer, and the product is destined for the EU market	Boolean	S	M	0 – No 1 – Yes
Product_Items_1	List of TPIDs corresponding to the product items listed on the invoice	TPID	M	M, if First_Seller_EU = 1	
Product_Items_2	List of product numbers corresponding to the product items listed on the invoice (in the same order as product_Items_1)	PN	M	M, if First_Seller_EU = 1	
Product_Price	Net unit packet price per each pair of TPID and product number (in the same order as product_Items_1)	Decimal	M	M, if First_Seller_EU = 1	
Invoice_Net	Total net amount of the invoice	Decimal	S	M	
Invoice_Currency	Currency of the invoice	Currency	S	M	
UI_Type	Identification of UI types covered by the invoice (recorded at the highest level of available aggregation)	Integer	S	M	1 – only unit packet level UIs 2 – only aggregated level UIs 3 – both unit packet and aggregated level UIs
upUIs	List of unit packet level UIs covered by the invoice	upUI(L)	M	M, if UI_Type = 1 or 3	
aUIs	List of aggregated level UIs covered by the invoice	aUI	M	M, if UI_Type = 2 or 3	
Invoice_comment	Comments by the reporting entity	Text(5000)	S	O	

3.7.1.3 Business validation

	EIV (4.1)
Technical validation	
VAL_EOID_SELLER	Invoice_Seller , EOID field must be equal to the seller EOID (EOID = Invoice_Seller)

3.7.1.4 Response

Invoice reporting– response					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Resp	Block of basic information elements	Component << Basic Information Response >>	S	M	Message_Type = EIV

3.7.1.5 Request sample

```
{
  "EO_ID": "QCUKR+1AB020054",
  "Event_Time": "19032014",
  "Message_Time_Long":"2019-03-20T14:16:45Z",
  "Invoice_Type1": 1,
  "Invoice_Type2": "other type",
  "Invoice_Number": "INV000001",
  "Invoice_Date": "2018-08-23T07:32:20.7878086+00:00",
  "Invoice_Seller": "SellerId",
  "Invoice_Buyer1": false,
  "Invoice_Buyer2": null,
  "Buyer_Name": "Buyer1",
  "Buyer_Address_1": "BuyerAddress",
  "Buyer_Address_2": "Municipality",
  "Buyer_Address_3": "PostalCode",
  "Buyer_CountryReg": "LU",
  "Buyer_TAX_N": "TAX0001",
  "First_Seller_EU": 1,
  "Product_Items_1": [ "11111-1111111","11111-1111112" ],
  "Product_Items_2": [ "01234567891234","01234567891235"  ],
  "Product_Price": [ "16.99", "19.99" ],
  "Invoice_Net": 10099.99,
  "Invoice_Currency": "EUR",
  "UI_Type": 1,
  "upUIs": [ "DANXXXXXXXXXXXX1PR012345678919030110",
"DANXXXXXXXXXXXX1PR012345678919030110" ],
  "aUIs": [ "DANXXXXXXXXXXX10FA00000119030110" ],
  "Invoice_comment": "Comments",
  "Message_Type": "EIV",
  "Code": "873345b2-882f-4064-91f0-90669b46c30a"
}
```

3.7.1.6 Successful response sample

HTTP Status 202

```
{
   "Code": "873345b2-882f-4064-91f0-90669b46c30a",
    "Message_Type": "EIV",
    "Error": false,
```

```
    "Errors": null,
    "Checksum": "G6HF5H"
}
```

3.7.1.7 Error response sample

Processing errors

HTTP status	Error Code	Error Description
<< Common response code >>		
400	FAILED_VALIDATION	VAL_EOID_SELLER

3.7.2 EPO – (4.2) Issuing of the order number

3.7.2.1 Description

Adds a purchase order event to a UI.

3.7.2.2 Description of the fields

Purchase order event					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Req	Block of basic information elements	Component << Basic Information Request >>	S	M	Message_Type = EPO
EO_ID	Economic operator identifier code of the submitting entity	EOID	S	M	
Event_Time	Time of event occurrence	Time(s)	S	M	
Message_Time_long	Message sending Time	Time(L)	S	M	
Order_Number	Number of the purchase order	Text	S	M	
Order_Date	Date of the purchase order	Date	S	M	
UI_Type	Identification of UI types covered by the purchase order (recorded at the highest level of available aggregation)	Integer	S	M	1 – only unit packet level UIs 2 – only aggregated level UIs 3 – both unit packet and aggregated level UIs
upUIs	List of unit packet level UIs covered by the purchase order	upUI(L)	M	M, if UI_Type = 1 or 3	
aUIs	List of aggregated level UIs covered by the purchase order	aUI	M	M, if UI_Type = 2 or 3	
Order_comment	Description of the reason for delayed recording of the purchase order	Text(5000)	S	O	

3.7.2.3 Response

Purchase order – response					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Resp	Block of basic information elements	Component << Basic Information Response >>	S	M	Message_Type = EPO

3.7.2.4 Request sample

```
{
  "EO_ID": "QCUKR+1AB020054",
  "Event_Time": "19032014",
  "Message_Time_Long":"2019-03-20T14:16:45Z",
  "Order_Number": "1234",
  "Order_Date": "2018-08-23T07:32:20.7878086+00:00",
  "UI_Type": 1,
  "upUIs": [ "DANXXXXXXXXXXXX1PR012345678919030110",
"DANXXXXXXXXXXXX1PR012345678919030110" ],
  "aUIs": [ "DANXXXXXXXXXXX10FA00000119030110" ],
  "Order_comment": "Comments",
  "Message_Type": "EPO",
  "Code": "873345b2-882f-4064-91f0-90669b46c30a"
}
```

3.7.2.5 Successful response sample

HTTP Status 202

```
{
    "Code": "873345b2-882f-4064-91f0-90669b46c30a",
    "Message_Type": "EPO",
    "Error": false,
    "Errors": null,
    "Checksum": "G6HF5H"
}
```

3.7.2.6 Error response sample

Processing errors

HTTP status	Error Code	Error Description
<< Common response code >>		

3.7.3 EPR – (4.3) Receipt of the payment

3.7.3.1 Description

Adds a payment record event to a UI.

3.7.3.2 Description of the fields

Payment record event					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Req	Block of basic information elements	Component << Basic Information Request >>	S	M	Message_Type = EPR
EO_ID	Economic operator identifier code of the submitting entity	EOID	S	M	
Event_Time	Time of event occurrence	Time(s)	S	M	
Message_Time_long	Message sending Time	Time(L)	S	M	
Payment_Date	Date of the payment receipt	Date	S	M	
Payment_Type	Type of payment	Integer	S	M	See PaymentType
Payment_Amount	Amount of the payment	Decimal	S	M	
Payment_Currency	Currency of the payment	Currency	S	M	
Payment_Payer1	Identification if the payer is located in the EU	Boolean	S	M	0 – No 1 – Yes
Payment_Payer2	Identity of the payer	EOID	S	M, if Payment_Payer1 = 1	
Payer_Name	Payer's registered legal name	Text(5000)	S	M, if Payment_Payer1= 0	
Payer_Address_1	Payer's street name and house number (or road number and kilometer)	Text(300)	S	M, if Payment_Payer1= 0	
Payer_Address_2	Payer's municipality (city, town or village)	Text(100)	S	M, if Payment_Payer1= 0	
Payer_Address_3	Payer's postal code	Text(50)	S	M, if Payment_Payer1= 0	'n/a' is permitted value if no postal code has been assigned
Payer_CountryReg	Payer's country of registration	Country	S	M, if Payment_Payer1 = 0	
Payer_TAX_N	Payer's tax registration number	Text(5000)	S	M, if Payment_Payer1 = 0	
Payment_Recipient	Identity of the recipient	EIOD	S	M	
Payment_Invoice	Indication if the payment corresponds to the existing invoice	Boolean	S	M	0 – No 1 – Yes
Invoice_Paid	Number of the invoice paid with the payment	Text(5000)	S	M, if Payment_Invoice = 1	

Payment record event					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
UI_Type	Identification of UI types covered by the payment (recorded at the highest level of available aggregation)	Integer	S	M, if Payment_Invoice = 0	1 – only unit packet level UIs 2 – only aggregated level UIs 3 – both unit packet and aggregated level UIs
upUIs	List of unit packet level UIs covered by the payment	upUI(L)	M	M, if AND Payment_Invoice = 0 UI_Type = 1 or 3	
aUIs	List of aggregated level UIs covered by the payment	aUI	M	M, if AND Payment_Invoice = 0 UI_Type = 2 or 3	
Payment_comment	Comments by the reporting entity	Text(5000)	S	O	

3.7.3.3 Business validation

	EPR (4.3)
Technical validation	
VAL_EOID_PAYMENT_RECIPIENT	Payment_Recipient, Technical validation recipient identity of the payment (EOID = Payment_Recipient)

3.7.3.4 Response

Payment record – response					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Resp	Block of basic information elements	Component << Basic Information Response >>	S	M	Message_Type = EPR

3.7.3.5 Request sample

```
{
  "EO_ID": "QCUKR+1AB020054",
  "Event_Time": "19032014",
  "Message_Time_Long":"2019-03-20T14:16:45Z",
  "Payment_Date": "2018-08-23T07:32:20.7878086+00:00",
  "Payment_Type": 1,
  "InvoiceType": 1,
  "UI_Type": 1,
  "Payment_Amount": 1.99,
  "Payment_Currency": "EUR",
  "Payment_Payer1": true,
  "Payment_Payer2": "PayerId",
  "Payer_Name": "PayerName",
  "Payer_Address": "Address",
  "Payer_CountryReg": "UK",
  "Payer_TAX_N": "TaxId",
  "Payment_Recipient": "PaymentRecipient",
  "Payment_Invoice": 1,
```

```
  "Invoice_Paid":  "test" ,
  "upUIs": [ "DANXXXXXXXXXXXX1PR012345678919030110",
"DANXXXXXXXXXXXX2PR012345678919030110" ],
  "aUIs": [ "DANXXXXXXXXXXX10FA00000119030110",
"DANXXXXXXXXXXX20FA00000119030110" ],
  "Payment_comment": "Comment",
  "Message_Type": "EPR",
  "Code": "873345b2-882f-4064-91f0-90669b46c30a"
}
```


3.7.3.6 Successful response sample

HTTP Status 202


```
{
   "Code": "873345b2-882f-4064-91f0-90669b46c30a",
    "Message_Type": "EPR",
    "Error": false,
    "Errors": null,
    "Checksum": "G6HF5H"
}
```


3.7.3.7 Error response sample

Processing errors

HTTP status		
<< Common response code >>		
400	FAILED_VALIDATION	VAL_EOID_PAYMENT_RECIPIENT

3.8 Recall an Event

3.8.1 RCL – (5.0) Recalls of requests, operational and transactional messages

3.8.1.1 Description

Given a recall id (“Code” in the return of any message) The caller can mark that event invalid.
This is possible for message types 2.1, 2.2, 2.3, 3.1 to 3.8, 4.1, 4.2 and 4.3)

Note that message 2.3 must meet certain conditions to be able to be recalled:

- Condition 1: Deact_Reason1 must be other than “2 – Product stolen”
- Condition 2: Less than 24h must’ve passed since the original 2.3 message was reported to the system. This time will be measured using the “Reception Time” (Time at which the message is received

at the entry system: Router / Primary) (The Secondary Repository will not perform the 24h validation and accept it as decided by the Primary Repository / Router to avoid the edge case when close to 24h)

3.8.1.2 Sequence Validation

For recall reason 1, previous messages can only be 3.3 EDP, 3.5 ETL and 3.8 EDX

	EUA 3.1	EPA 3.2	EDP 3.3	ERP 3.4	ETL 3.5	EUD 3.6	EVR 3.7	EDX 3.8
RCL 5.0 (Recall Reason = 1)	No	No	Yes	No	Yes	No	No	Yes

3.8.1.3 Description of the fields

Recall – request					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Req	Block of basic information elements	Component << Basic Information Request >>	S	M	Message_Type = RCL
EO_ID	Economic operator identifier code of the submitting entity	EOID	S	M	
Message_Time_long	Message sending Time	Time(L)	S	M	
Recall_CODE	Message recall code provided to the message sender in the acknowledgement of the original message to be recalled	UUID	S	M	Recall_CODE
Recall_Reason 1	Reason for recalling the original message	Integer	S	M	See RecallReasonType
Recall_Reason 2	Description of the reason for recalling the original message	Text(5000)	S	M, if Recall_ Reason 1 = 3 (other reason)	
Recall_Reason 3	Any additional explanations on the reason for recalling the original message	Text(5000)	S	O	

3.8.1.4 Response

Recall – response					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Resp	Block of basic information elements	Component << Basic Information Response >>	S	M	Message_Type = RCL

3.8.1.5 Request sample

```
{
  "EO_ID":"QCUKR+1AB020054",
  "Message_Time_Long":"2019-03-20T14:16:45Z",
  "RecallReason1":1,
  "RecallReason2":1,
  "RecallReason3":"Comments",
  "Message_Type":"RCL",
  "Code": "873345b2-882f-4064-91f0-90669b46c30a"
}
```

3.8.1.6 Successful response sample

HTTP Status 202

```
{
    "Code": "873345b2-882f-4064-91f0-90669b46c30a",
    "Message_Type": "RCL",
    "Error": false,
    "Errors": null,
    "Checksum": "G6HF5H"
}
```

3.8.1.7 Error response sample

HTTP status	Error Code	Error Description
<< Common response code >>		
400	RECALL_AFTER_ONE_WORKING_DAY	For requests of unit level or aggregated level UIs (ISU, ISA), recalls can be performed up to one working day after the original message.
400	RECALL_NOT_LAST_EVENT	Recall code not found: ERROR: The provided recall code does not exist. Recallcode not the last event ERROR: Please note that a recall can only be performed on valid messages that referred to UIs which were not later used in other messages.

400	RECALL_EXPIRED	Error provided when attempting to Recall a 2.3 message (with Deact_Reason1 != 2) after 24h have passed.

3.9 Flat file and registry file upload initiation service

3.9.1 ULO – Flat file and registry File upload

3.9.1.1 Description

This initial ULO request allows the caller to gain permission and details in order to upload a file.

3.9.1.2 Description of the fields

Flat file initiation – request					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
Message_Type	The identifier of the type of message	Text	S	M	Message_Type = ULO
ID_Issuer	ID Issuer code compliant with ISO/IEC 15459 regulation	IIID	S	M	
File_Type	The type of the file intended to be uploaded	int	S	M	1– Registry file 2 – Flat file
Callback_Url	The URL on the ID Issuer side that will be called asynchronously	Text(5000)	S	O	

3.9.1.3 Response

Flat file initiation – response					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
Message_Type	The identifier of the type of message	Text	S	M	Message_Type = ULO
Code	Unique identifier of the message. Used for recall too.	Text(50)	S	M	
Upload_Url	The URL that the file should be HTTP put to	Text(5000)	S	M	
Error	Indicates the failure of the message reception	Boolean	S	M	0 – No 1- Yes
Errors	Array containing Error_Code, Error_Descr, InternalId	Text	S	M if Error = 1	

3.9.1.4 Request sample

```
{
    "Message_Type": "ULO",
    "ID_Issuer": "IID"
    "File_Type": 1
}
```

3.9.1.5 Successful response sample

HTTP Status 200


```
{
    "Message_Type": "ULO",
    "Upload_Url ": "https://test.s3.eu-west-1.amazonaws.com/9adda342-012c-
46e6-b5f9-18bc73a693d7?X-Amz-Expires=299&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-
Amz-Credential=AKIAID6ZB7LZNC6M6BBA/20190218/eu-west-1/s3/aws4_request&X-Amz-
Date=20190218T135040Z&X-Amz-SignedHeaders=host&X-Amz-
Signature=eb4133f0e2f5e283c65d8b169b378ae7b6946570d485e95e976c605ae4d5ed47",
    "Code": "873345b2-882f-4064-91f0-90669b46c30a",
    "Error": false,
    "Errors": null,
    "Checksum": "G6HF5H"
}
```


3.9.2 ULOD – Flat file and registry File callback

3.9.2.1 Description

This ULOD request is a response to the original ULO message

3.9.2.2 Description of the fields

Flat file initiation – request					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
Message_Type	The identifier of the type of message	Text	S	M	Message_Type = ULOD
Code	Unique identifier of the message. Used for recall too.	Text(50)	S	M	
Error	Indicates the failure of the message reception	Boolean	S	M	0 – No 1- Yes
Errors	Array containing Error_Code, Error_Descr, InternalId	Text	S	M if Error = 1	

3.9.2.3 Response

Flat file initiation – response					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
Message_Type	The identifier of the type of message	Text	S	M	Message_Type = ULOD
Code	Unique identifier of the message. Used for recall too.	Text(50)	S	M	
Error	Indicates the failure of the message reception	Boolean	S	M	0 – No 1- Yes
Errors	Array containing Error_Code, Error_Descr, InternalId	Text	S	M if Error = 1	

3.9.2.4 Request sample

```
{
    "Message_Type": "ULOD",
    "Code": "873345b2-882f-4064-91f0-90669b46c30a",
    "Error": false,
    "Errors": null
}
```

3.9.2.5 Successful response sample

HTTP Status 200

```
{
    "Message_Type": "ULOD",
    "Code": "873345b2-882f-4064-91f0-90669b46c30a",
    "Error": false,
    "Errors": null,
    "Checksum": "G6HF5H"
}
```

3.9.3 PLO – Partial Flat file and registry transmission

3.9.3.1 Description

This message enables the ID issuer to update the Register and the FlatFile in an incremental manner and in a synchronous way.

3.9.3.2 Description of the fields

Flat file initiation – request					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
Message_Type	The identifier of the type of message	Text	S	M	Message_Type = PLO
ID_Issuer	The identifier of the economic operator sending the message	IIID	S	M	Note : Checked on token too
File_Type	The type of the file intended to be uploaded	int	S	M	1– Machines.csv 2 – Facilities.csv 3 - EconomicIdentifiers.csv 4- MachineLookup.csv. 5- ProductLookup.csv 6- RegularExpression.csv
File_Content	Content of the csv file	Text(50)	S	M	

3.9.3.3 How to report lists

In order to be able to still use the comma ",” when generating a value (for example Machine Identifiers) and be able to report a list of values inside a CSV transmitted by JSON (for example, the new Machine Parts list M_plist) please use the character "#” to separate the values (See the sample below).

3.9.3.4 Response:

Flat file initiation – response					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
Message_Type	The identifier of the type of message	Text	S	M	Message_Type = PLO
Code	Unique identifier of the message. Used for recall too.	Text(50)	S	M	
Error	Indicates the failure of the message reception	Boolean	S	M	0 – No 1- Yes
Errors	Array containing Error_Code, Error_Descr, InternalId	Text	S	M if Error = 1	

3.9.3.5 Request sample

```
{
    "Message_Type": "PLO",
    "ID_Issuer": "IDISSUERCODE",
    "File_Type": 1,
    "File_Content":["CSVVALUE1; CSVVALUE2; CSVVALUE3",
        "CSVVALUE1; CSVLISTVALUE1#CSVLISTVALUE2#CSVLISTVALUE3; CSVVALUE3",
        "CSVVALUE1; CSVVALUE2; CSVVALUE3",
        "CSVVALUE1; CSVVALUE2; CSVVALUE3"]
}
```

Note that in the sample the part with CSVLISTVALUE1#CSVLISTVALUE2#CSVLISTVALUE3 represents for example the M_plist, which is a list inside the CSV. Use « # » to separate the list values, the character « ; » is reserved to separate cells in the CSV. The character « , » may be used to identify a M_ID.

3.9.3.6 Successful response sample

HTTP Status 202

```
{
    "Message_Type": "PLO",
    "Code": "873345b2-882f-4064-91f0-90669b46c30a",
    "Error": false,
    "Errors": null,
    "Checksum": "G6HF5H"
}
```

3.10 Connectivity Test Message

3.10.1 CTM – Connectivity Test Messages

3.10.1.1 Description

The connectivity test message is sent by the Router or Secondary in order to test the connectivity.

3.10.1.2 Description of the fields

Recall – request					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Req	Block of basic information elements	Component << Basic Information Request >>	S	M	Message_Type = CTM

3.10.1.3 Response

Recall – response					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Resp	Block of basic information elements	Component << Basic Information Response >>	S	M	Message_Type = CTM

3.10.1.4 Request sample

```
{
"Message_Type":"CTM",
"Code": null
}
```

3.10.1.5 Successful response sample

HTTP Status 202

```
{
    "Code": "873345b2-882f-4064-91f0-90669b46c30a",
    "Message_Type": "CTM",
    "Error": false,
    "Errors": null,
    "Checksum": "G6HF5H"
}
```

3.10.1.6 Error response sample

HTTP status	Error Code	Error Description
<< Common response code >>		

3.11 Query API interface

3.11.1 LUQ – Query Data

3.11.1.1 Description

Allows to query different types of data via the API interface. Which data depends on the permissions of the entity authenticated in the API (ID Issuer / Primary Repository / Competent Authority).

Query Type ID 1 (Economic Operator ID), 2 (Facility ID) and 3 (Machine ID) are available to ID Issuers, Primary Repository providers and Competent Authorities.

Query Type ID 4 (Events), 5 (Unique Identifiers) and 6 (Vehicles) are only available to Competent Authorities.

Query Types 4 and 5 can use the field "Query_Param" to request the full list of UIs by adding {"ListEventUI":"true"}. The default response will assume {"ListEventUI":"false"}.

3.11.1.2 Query Type ID definition

Query Type ID	Description
1	EO Query
2	Facility Query
3	Machine Query
4	Event Query
5	UI query
6	Vehicle Query

3.11.1.3 Description of the fields (Request)

Query Message – request					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Req	Block of basic information elements	Component << Basic Information Request >>	S	M	Message_Type = LUQ
Query_UserID	Unique user identifier	Text(5000)	S	M	OAUTH2 ClientID (for ID Issuers / Primary Providers) or User Identification (for competent

					authorities, i.e. email)
Query_Type	Query type description	Integer	S	M	Query Type ID (1,2,3,4,5,6) see 3.11.1.2
Query_Elements	List of elements	EconomicOperatorID (if Query_Type = 1) FacilityID (if Query_Type = 2) MachineID (if Query_Type = 3) Event (if Query_Type = 4) UniqueIdentifier (if Query_Type = 5) Vehicle (if Query_Type = 6)	M	M	
Query_Param	Query Parameter, list of key value pair.	Text(Dictionary of strings)	M	O	Filters only available for Query Types 4 and 5.

3.11.1.4 Response

Query Message – response				
	Description	**Data Type**	**Cardinality**	**Values**
BasicInfo_Resp	Block of basic information elements	Component << Basic Information Response >>	S	Message_Type = LUQ
Query_Result	JSON containing the response	Text(5000)	S	Depends on the Query Type, see individual examples below

3.11.1.5 Request / Response by Query Type ID

3.11.1.5.1 Query Type ID = 1 (Economic Operator ID)

This request allows to query all the information of message 1.1 of Annex II related to an Economic Operator ID. Up to 20 Economic Operator IDs can be queried in one request.

The request is available to ID Issuers, Primary Repository Providers and Competent Authorities.

Request

Query Message – request					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Req	Block of basic information elements	Component << Basic Information Request >>	S	M	Message_Type = LUQ
Query_UserID	Unique user identifier	Text(5000)	S	M	OAUTH2 ClientID (for ID Issuers / Primary Providers) or User Identification (for competent authorities, i.e. email)
Query_Type	Query type description	Integer	S	M	1
Query_Elements	List of elements	EconomicOperatorID	M	M	Limited to 20 EOs

Response

Note that even if the regulation has updated the Address fields, there will be several entries with the old format in the system, so the Query API returns the ones available on the system.

EO Element JSON Object				
Field	**Description**	**Data Type**	**Cardinality**	**Values**
EOID	Original UI that is used to perform the search	EOID	S	
EO_street	Economic operator's street name and house number (or road number and kilometer)	Text(300)	S	
EO_municipality	Economic operator's municipality (city, town or village)	Text(100)	S	
EO_postcode	Economic operator's postal code	Text(50)	S	
EO_A_info	Additional information on economic operator's address (e.g. location in the shopping mall or industrial area)	Text(100)	S	
EO_Name1	Economic operator's registered name	Text(100)	S	

EO_Name2	Economic operator's alternative or abridged name	Text(100)	S	
EO_Address_StreetOne	Street part of the Address	Text(5000)	S	
EO_Address_StreetTwo	Second Element of the Street part of the Address	Text(5000)	S	
EO_Address_City	City	Text(5000)	S	
EO_Address_PostCode	PostalCode information	Text(5000)	S	
EO_CountryReg	Economic operator's country of registration	Country	S	See Country
EO_Email	Economic operator's email address; used to inform about registration process, incl. subsequent changes and other required correspondence	Text(80) (Regex protected)	S	
VAT_R	Indication of the VAT registration status	Boolean	S	0–No VAT registration 1– VAT number exists
VAT_N	Economic operator's VAT number	Text(20)	S	
TAX_N	Economic operator's tax registration number	Text(20)	S	
EO_ExciseNumber1	Indication if the economic operator has an excise number issued by the competent authority for the purpose of identification of persons/premises	Boolean	S	0– No SEED number 1– SEED number exists
EO_ExciseNumber2	Economic operator's excise number issued by the competent authority for the purpose of identification of persons/premises	SEED	S	
OtherEOID_R	Indication if the economic operator has been allocated an identifier by another ID Issuer	Boolean	S	0– No 1– Yes
OtherEOID_N	Economic operator identifier codes allocated by other ID Issuers	EOID	M	
Reg_3RD	Indication if the registration is made on behalf of a retail outlet operator not otherwise involved in the tobacco trade	Boolean	S	0– No 1– Yes
Reg_EOID	Identifier of the economic operator that acts on behalf of a retail outlet operator not otherwise involved in the tobacco trade	EOID	S	
EO_OtherID	Optional identifier	Text(50)	S	

Query_Type 1 request sample

```
{
    "Message_Type": "LUQ",
    "Query_Elements": ["QCCCT5606221025744"],
    "Query_Type" : "1",
    "Query_UserID" : "3rsubbrvk13sojihem2pqo14bn"

}
```

Query_Type 1 Successful response sample

HTTP Status 200

```
{
    "Query_Result": [
        {
            "EOID": "QCCCT5606221025744",
            "EO_Name1": "JOSE GARCIA",
            "EO_Name2": "EO NAME 123",
            "EO_Address_StreetOne": "Av. Street name , 16",
            "EO_Address_StreetTwo": null,
            "EO_Address_City": null,
            "EO_Address_PostCode": null,
            "EO_CountryReg": "BE",
            "EO_Email": "samplemail@sample.com",
            "VAT_R": 0,
            "VAT_N": null,
            "TAX_N": "TAXN123123123",
            "EO_ExciseNumber1": 0,
            "EO_ExciseNumber2": null,
            "OtherEOID_R": 0,
            "OtherEOID_N": [],
            "Reg_3RD": 0,
            "Reg_EOID": null,
            "EO_OtherID": null,
            "EO_street": null,
            "EO_postcode": null,
            "EO_A_info": null,
           "EO_municipality":null
        }
    ],
    "Code": "3585e3da-75aa-54a2-9b51-2f1243ed5fa9",
    "Message_Type": null,
    "Error": 0,
    "Errors": null,
    "Checksum": null
}
```

3.11.1.5.2 Query Type ID = 2 (Facility ID)

This request allows to query all the information of message 1.4 of Annex II related to a Facility ID. Up to 20 Facility IDs can be queried in one request.

The request is available to ID Issuers, Primary Repository Providers and Competent Authorities.

Request

Query Message – request					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Req	Block of basic information elements	Component << Basic Information Request >>	S	M	Message_Type = LUQ
Query_UserID	Unique user identifier	Text(5000)	S	M	OAUTH2 ClientID (for ID Issuers / Primary Providers) or User Identification (for competent authorities, i.e. email)
Query_Type	Query type description	Integer	S	M	2
Query_Elements	List of elements	FacilityID	M	M	Limited to 20 FIDs

Response

Facility Element JSON Object				
Field	**Description**	**Data Type**	**Cardinality**	**Values**
F_ID	Facility identifier code	FID	S	
EO_ID	Economic operator identifier code	EOID	S	
F_street	Facility's street name and house number (or road number and kilometer)	Text(300)	S	
F_municipality	Facility's municipality (city, town or village)	Text(100)	S	
F_postcode	Facility's postal code	Text(50)	S	'n/a' is permitted value if no postal code has been assigned
F_A_info	Additional information on facility's address (e.g. location in the shopping mall or industrial area)	Text(100)	S	
PrevFID_B	Indication if the facility was acquired from another operator and had already a facility identifier code	Boolean	S	0 – No (first time registration) 1 – Yes
PrevFID_ID	Previous facility identifier used by the former operator of the facility	FID	S	

F_Address_StreetOne	Facility's address – Street part of the Address	Text(5000)	S	
F_Address_StreetTwo	Facility's address – Second Element of the Street part of the Address	Text(5000)	S	
F_Address_City	Facility's address – City	Text(5000)	S	
F_Address_PostCode	Facility's address – PostalCode information	Text(5000)	S	
F_Country	Facility's country	Country	S	See Country
F_Type	Type of facility	Integer	S	See FacilityType
F_Type_Other	Description of other facility type	Text(5000)	S	
F_Status	Indication if a part of the facility has a bonded warehouse status	Boolean	S	0– No 1– Yes
F_ExciseNumber1	Indication if the facility has an excise number issued by the competent authority for the purpose of identification of persons/premises	Boolean	S	0– No SEED number 1– SEED number exists
F_ExciseNumber2	Facility's excise number issued by the competent authority for the purpose of identification of persons/premises	SEED	S	
OtherFID_R	Indication if the facility has been allocated an identifier by another ID Issuer	Boolean	S	0– No 1– Yes (possible only for non-EU facilities)
OtherFID_N	Facility identifier codes allocated by other ID Issuers	FID	M	List of FIDs
Reg_3RD	Indication if the registration is made on behalf of a retail outlet operator not otherwise involved in the tobacco trade	Boolean	S	0 – No 1 – Yes (possible only if F_Type = 3)
Reg_EOID	Identifier of the economic operator that acts on behalf of a retail outlet operator not otherwise involved in the tobacco trade	EOID	S	
Latitude	GPS coordinates, latitude value	Decimal	S	

Longitude	GPS coordinates, longitude value	Decimal	S	
Extensibility	Optional extensibility field	Text(5000)	S	

Query_Type 2 request sample

```
{
    "Message_Type": "LUQ",
    "Query_Elements": ["QCXXX<1DE1416673857877"],
    "Query_Type" : "2",
    "Query_UserID" : "xxxxxxxxxyyyyyyyyyzzzzz",
    "Code" : "335d0201-3d3b-43d7-a1af-e3fa3b51b3c8"
}
```

Query_Type 2 Successful response sample

HTTP Status 200

```
{
"Query_Result": [
        {
            "F_ID": "QCXXX<1DE1416673857877",
            "EO_ID": "QXXXXX555YYYY483857877",
            "F_Address_StreetOne": "Sample Str.,13,111111,City",
            "F_Address_StreetTwo": null,
            "F_Address_City": null,
            "F_Address_PostCode": null,
            "F_Country": "FR",
            "F_Type": 3,
            "F_Type_Other": null,
            "F_Status": 0,
            "F_ExciseNumber1": 0,
            "F_ExciseNumber2": null,
            "OtherFID_R": 0,
            "OtherFID_N": [],
            "Reg_3RD": 0,
            "Reg_EOID": null,
            "F_street": null,
            "F_postcode": null,
            "F_A_info": null,
            "F_municipality": null,
            "PrevFID_B": null,
            "PrevFID_ID": null,
            "Latitude": 33.7584,
            "Longitude": -75.2358,
            "Extensibility": null
        }
    ],
    "Code": "7e89d3bc-0c39-573d-9e1c-616f3b8267f5",
    "Message_Type": null,
    "Error": 0,
    "Errors": null,
    "Checksum": null
}
```

3.11.1.5.3 Query Type ID = 3 (Machine ID)

This request allows to query all the information of message 1.7 of Annex II related to a Machine ID. Up to 20 Facility IDs can be queried in one request.

The request is available to ID Issuers, Primary Repository Providers and Competent Authorities.

Request

Query Message – request					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Req	Block of basic information elements	Component << Basic Information Request >>	S	M	Message_Type = LUQ
Query_UserID	Unique user identifier	Text(5000)	S	M	OAUTH2 ClientID (for ID Issuers / Primary Providers) or User Identification (for competent authorities, i.e. email)
Query_Type	Query type description	Integer	S	M	3
Query_Elements	List of elements	MachineID	M	M	Limited to 20 MIDs

Response

Machine Element JSON Object				
Field	**Description**	**Data Type**	**Cardinality**	**Values**
M_ID	Machine Id	MID	S	
EO_ID	Economic operator identifier code	EOID	S	
F_ID	Facility identifier code	FID	S	
PrevMID_B	Indication if the object of this request was already registered, e.g. in relation to another machine identifier code	Boolean	S	0 – No (first time registration) 1 - Yes
PrevMID_ID	Previous machine identifier used for the object of this request	MID	S	

M_entirety	Indication if this request concerns the machine (v. a part of thereof)	Boolean	S	0 – No (machine part) 1 – Yes (machine)
P_Producer	Part's producer	Text(20)	S	
P_Model	Part's model	Text(20)	S	
P_Number	Part's serial number	Text(20)	S	
P_Mobile	Indication if this part is intended to be used with multiple machines (fixed v. mobile part)	Boolean	S	0 – No (fixed part) 1 – Yes (mobile part)
P_ATD1	Indication if an anti-tampering device in the sense of Article 2(7) records the functioning of this part	Boolean	S	0 – No 1 – Yes
P_ATD2	Anti-tampering's device serial number	Text(100)	S	
P_Description	Part's description explaining its technical function	Text(500)	S	
M_Producer	Machine producer	Text(20)	S	
M_Model	Machine model	Text(20)	S	
M_Number	Machine serial number	Text(20)	S	
M_parts	Indication if the machine consists of multiple separately identifiable parts	Boolean	S	0 - No 1 - Yes
M_plist	List of the identifiable parts	MID	M	List of MIDs (parts)
M_ATD	Serial number of the anti-tampering device in the sense of Article 2(7)	Text(100)	S	
M_Capacity	Maximum capacity over 24hour production cycle expressed in unit packets	Integer	S	

Query_Type 3 request sample

```
{
    "Message_Type": "LUQ",
    "Query_Elements": ["XXXX560112512312312312 3"],
    "Query_Type" : "3",
    "Query_UserID" : "xxxxxxxxxyyyyyyyyyzzzzz",
    "Code" : "335d0201-3d3b-43d7-a1af-e3fa3b51b3c8"
}
```

Query_Type 3 Successful response sample

HTTP Status 200

```
{
   "Query_Result": [
        {
            "M_ID": "XXXX560112512312312312 3",
            "EOID": null,
            "F_ID": "XXXX5605555595686",
            "M_Producer": "GD",
            "M_Model": "GD X3",
            "M_Number": "071366506",
            "M_Capacity": 720000,
            "PrevMID_B": 0,
            "PrevMID_ID": null,
            "M_entirety": 1,
```

```
            "P_Producer": null,
            "P_Model": null,
            "P_Number": null,
            "P_Mobile": null,
            "P_ATD1": null,
            "P_ATD2": null,
            "P_Description": null,
            "M_parts": 0,
            "M_plist": [],
            "M_ATD": "071366506",
            "Active": 1,
            "Technical_Owner": "071366506",

        }
    ],
    "Code": "7e89d3bc-0c39-573d-9e1c-616f3b8267f5",
    "Message_Type": null,
    "Error": 0,
    "Errors": null,
    "Checksum": null
}
```


3.11.1.5.4 Query Type ID = 4 (Event)

This request allows to query information related to an event by recall code. Up to 20 events can be queried in one request.

The request is only available to Competent Authorities.

Request

Query Message – request					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Req	Block of basic information elements	Component << Basic Information Request >>	S	M	Message_Type = LUQ
Query_UserID	Unique user identifier	Text(5000)	S	M	OAUTH2 ClientID (for ID Issuers / Primary Providers)
Query_Type	Query type description	Integer	S	M	4
Query_Elements	List of elements	EventID	M	M	Limited to 20 Events
Query_Param	Query Parameter, list of key value pair.	Text(Dictionary of strings)	M	O	{"ListEventUI":"true"} {"ListEventUI":"false"}

Response

Event Element JSON Object

Field	Description	Data Type	Cardinality	Values
RecallCode	recallCode of the Event	Text(5000)	S	
Event_Type	Type of the Event	Text	S	
Event_Time	Time of event occurrence	Time(s)	S	
EO_ID	Economic operator identifier code of the submitting entity	EOID	S	
F_ID	Dispatch facility identifier code	FID	S	
upUIs	List of unit packet level UIs subject to the dispatch	upUI(L)	M	
aUIs	List of aggregated level UIs subject to the dispatch	aUI	M	
Available when EDP (3.3 dispatch)				
Destination_ID1	Indication if the destination facility is located on the EU territory and if it is a vending machine (VM)	Integer	S	1 – Non EU dest. 2 – EU destination other than VM – fixed quantity delivery 3 – EU VM(s) 4 – EU destination other than VM – delivery with VV
Destination_ID2	Destination facility identifier code	FID	S	
Destination_ID3	Destination facility identifier code(s) – possible multiple vending machines	FID	M	
Destination_ID4	Destination id facility codes	FID	M	
Destination_ID5	Destination facility's street name and house number (or road number and kilometer)	Text(300)	S	
Destination_ID6	Destination facility's municipality (city, town or village)	Text(100)	S	
Destination_ID7	Destination facility's postal code	Text(50)	S	'n/a' is permitted value if no postal code has been assigned
Destination_ID8	Destination facility's country	Country	S	
Destination_ Address_StreetOne	Destination facility's full address - Street part of the Address	Text(5000)	S	

Destination_ Address_StreetTwo	Destination facility's full address - Second Element of the Street part of the Address	Text(5000)	S	
Destination_ Address_City	Destination facility's full address - City	Text(5000)	S	
Destination_ Address_PostCode	Destination facility's full address - PostalCode information	Text(5000)	S	
Transport_mode	Mode of transport by which the product leaves the facility, see: Commission Regulation (EC) No 684/2009, Annex II, Code List 7	Integer	S	See TransportMode
Transport_vehicle	Identification of the mode of transport (i.e. number plates, train number, plane/flight number, ship name or other identification)	Text(5000)	S	'n/a' is permitted value if Transport_mode = 0 and product movement takes place between adjacent facilities and is delivered manually
Available when Disaggregation Event				
disaUI_comment	Comments by the reporting entity	Text(5000)	S	
Available when Deactivation Event				
Deact_Reason1	Identification of the reason for deactivation	Integer	S	See DeactivationReasonType
Deact_Reason2	Description of other reason	Text(5000)	S	
Deact_Reason3	Additional description of the reason	Text(Limited to the set of known deactivation_types)	S	
Available when Transloading Event				
Destination_ID1	Indication if the destination facility is located on the EU territory	Integer	S	0 – No 1 – Yes
Destination_ID2	Destination facility identifier code	FID	S	
Destination_ID3	Destination facility's street name and house number (or road number and kilometer)	Text(300)	S	
Destination_ID4	Destination facility's municipality (city, town or village)	Text(100)	S	
Destination_ID5	Destination facility's postal code	Text(50)	S	

Destination_ID6	Destination facility's country	Country	S	
Transport_mode	Mode of transport to which the product is trans-loaded, see: Commission Regulation (EC) No 684/2009, Annex II, Code List 7	Integer	S	TransportMode
Transport_vehicle	Identification of the vehicle (i.e. number plates, train number, plane/flight number, ship name or other identification)	Text(5000)	S	
Transport_cont1	Indication if the transport is containerised and uses an individual transport unit code (e.g. SSCC)	Boolean	Transport_cont1	0 – No 1 – Yes
Transport_cont2	Individual transport unit code of the container	ITU	Transport_cont2	
EMCS	Dispatch under the Excise Movement and Control System (EMCS)	Boolean	EMCS	0 – No 1 – Yes
EMCS_ARC	Administrative Reference Code (ARC)	ARC	EMCS_ARC	
UI_Type	Identification of UI types subject to the trans-loading (recorded at the highest level of available aggregation)	Integer	S	1 – only unit packet level UIs 2 – only aggregated level UIs 3 – both unit packet and aggregated level UIs

Query_Type 4 request sample


```
{
"Message_Type":"LUQ"
"Code": null,
"Query_Type": 4,
"Query_Elements": ["873377b2-882f-5064-91f0-90669b46c30a","875545b2-882f-
5064-91f0-40669b46c30a"],
"Query_Param": {"ListEventUI":"false"}
}
```


Query_Type 4 Successful response sample

HTTP Status 200


```
{
    "Query_Result": [
        {
            "RecallCode": "c7fa0e4b-2f5a-58c2-99f3-316dcb2aacc7",
            "Event_Type": "EDP",
            "Event_Time": "2024-02-05T10:00:00+00:00",
            "EO_ID": "EOID_TEST_PRIMARY_A",
            "F_ID": "FID_TEST_PRIMARY_B",
```

```
            "upUIs": [
                "b2f6f0d004-routing-eo2-3-"
            ],
            "aUIs": [
                "b2f6f0d004-a-routing-eo2-self1",
                "b2f6f0d004-a-routing-eo2-self6",
                "b2f6f0d004-a-routing-eo2-5"
            ],
            "Destination_ID1": "EUOtherThanVM",
            "Destination_ID2": "F_ID_TYPE_4",
            "Destination_ID3": null,
            "Destination_ID4": null,
            "Destination_ID5_Address_StreetOne": null,
            "Destination_ID5_Address_StreetTwo": null,
            "Destination_ID5_Address_City": null,
            "Destination_ID5_Address_PostCode": null,
            "Transport_mode": "Other",
            "Transport_vehicle": "n/a",
            "disaUI_comment": null,
            "Deact_Reason1": null,
            "Deact_Reason2": null,
            "Deact_Reason3": null,
            "aUI": null,
            "EMCS_ARC": null,
            "SAAD_number": null
        }
    ],
    "Code": "e890fdc4-efe2-58aa-abde-eb28ee9d1e7a",
    "Message_Type": null,
    "Error": 0,
    "Errors": null,
    "Checksum": "c019169532b45de15171b1d6d40c91af"
}
```


3.11.1.5.5 Query Type ID = 5 (UniqueIdentifier)

This request allows to query information related to upUIs or aUIs. Up to 20 UIs can be queried in one request.

This request is only available to Competent Authorities

Request

Query Message – request					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Req	Block of basic information elements	Component << Basic Information Request >>	S	M	Message_Type = LUQ
Query_UserID	Unique user identifier	Text(5000)	S	M	OAUTH2 ClientID (for ID Issuers / Primary Providers)
Query_Type	Query type description	Integer	S	M	5
Query_Elements	List of elements	Unique Identifier	M	M	Limited to 20 Unique Identifiers

Response if upUI

Event Element JSON Object				
Field	**Description**	**Data Type**	**Cardinality**	**Values**
Id	Original UI that is used to perform the search		S	
UI_Status	UI identifier State	Integer	S	See UniqueIdentifierState
UI_Status_Description		Text(5000)	S	
Id_Type	Type of UI	Integer	S	upUI = 1 , aUI = 2
upUI_i	upUI(i) as transmitted by ID Issuer	upUI(i)	S	
upUI_s	Human readable UI upUI(s) as transmitted by ID Issuer	upUI(s)	S	
upUI_L	upUI(L) as transmitted by the Primary Repository on message 3.1	upUI(L)	S	
IIID	ID Issuer that generated the UI	IIID	S	
EO_ID	Economic operator identifier code of the submitting entity (either EU manufacturer or EU importer)	EOID	S	
F_ID	Facility identifier code	FID	S	
M_ID	Machine identifier code	MID	S	
P_Type	Type of tobacco product	Integer	S	See TobaccoProductType
P_SubType_Exists	Indicates if the product "subtype name" exists. Subtype name provides further product identification beyond a product's brand name.	Boolean	S	
P_SubType				
P_OtherType	Description of other type of tobacco product	Text(200)	S	
P_CN	Combined Nomenclature (CN) code	Text(200)	S	
P_Brand	Brand of tobacco product	Text(200)	S	

P_weight	Average gross weight of unit packet, including packaging, in grams with 0,1 gram accuracy	Decimal	S	
TP_ID	The identification number of the product used in the EU-CEG system.	TPID	S	
TP_PN	Tobacco product number used in the EU-CEG system	PN	S	
Intended_Market	Intended country of retail sale.	Country	S	
Intended_Route1	Indication if the product is intended to be moved across country boarders with terrestrial transport.	Boolean	S	0 – No 1 – Yes
Intended_Route2	The first country of terrestrial transport after the product leaves the Member State of manufacturing or the Member State of importation.	Country	S	
Import	Indication if the product is imported into the EU	Boolean	S	0 – No 1 – Yes
P_OtherID	Optional Product ID	Text(20)	S	
Event_Aggregations	List of Aggregation	Aggregation Element JSON Object	M	
Event_List	List of Event recallcodes allowing the lookup of the different events	Text(Array of strings; internal limit = 5000)	M	

Response if aUI

Aggregation Element JSON Object					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
RecallCode	RecallCode of the aggregation event	Text(5000)	S	M	
ParentId	Parent Id of the aggregation event	Text(5000)	S	M	
Event_UI	The UI that is part of the aggregation	Text	M	M	
EO_ID	Economic operator identifier code of the submitting entity	EOID	S	M	
F_ID	Facility identifier code	FID	S	M	
Event_Time	Time of event occurrence	Time(s)	S	M	
aUI	Aggregated level UI	aUI	S	M	

Query_Type 5 request sample

```
{
"Message_Type":"LUQ"
"Code": null,
"Query_Type": 5,
"Query_Elements": ["id1","id2"]
}
```

Query_Type 5 Successful response sample

HTTP Status 200


```
{
     "Query_Result": [
          {
               "Id": "027405d5-8367-44bc-ad49-dd1d1d766839",
               "UI_Status": 1,
               "UI_Status_Description": null,
               "Id_Type": 1,
               "upUI_i": "027405d6-8347-44bc-ad49",
               "upUI_s": "027405d6-8347-44bc-ad49",
               "upUI_L": "027405d6-8347-44bc-ad49-dd1d1d766839",
               "IIID": null,
               "EO_ID": "XXXX5606221019279",
               "F_ID": null,
               "M_ID": null,
               "P_Type": null,
               "P_OtherType": null,
               "P_CN": null,
               "P_Brand": null,
               "P_weight": 0.0,
               "TP_ID": null,
               "TP_PN": null,
               "Intended_Market": null,
               "Intended_Route1": 0,
               "Intended_Route2": null,
               "Import": 0,
               "P_OtherID": null,
               "Event_Aggregations": [
                    {
                         "RecallCode": "CODE",
                         "ParentId": "AGGREGATION-STAN1",
                         "Event_UI": null,
                         "EO_ID": "Z25Q1H44IB3002078572YSHREJCOL",
                         "F_ID": null,
                         "Event_Time": "2019-06-04T14:22:42.679+00:00",
                         "aUI": null
                    },
                    {
                         "RecallCode": "fbec47f2-5771-5a29-9d5c-e6ddb4cbca09",
                         "ParentId": "1917-AAE-IFT-upUI-01_aUI",
                         "Event_UI": null,
                         "EO_ID": "I85B2J22DN7823851457TPULHSIDZ",
                         "F_ID": "REGRESSION5860808396MISTRESSE",
                         "Event_Time": "2019-06-26T16:00:00+00:00",
                         "aUI": null
                    },
                    {
                         "RecallCode": "CODE",
                         "ParentId": "AGGREGATION-STAN1",
                         "Event_UI": null,
                         "EO_ID": "Z25Q1H44IB3002078572YSHREJCO",
                         "F_ID": null,
                         "Event_Time": "2019-06-04T13:58:20.254+00:00",
```

```
                        "aUI": null
                    }
                ],
                "Event_List": [
                    "143dfb19-d2f7-5c11-b384-a8bf3feacc3b",
                    "2d836564-b943-502d-9842-ff1d1dda93d1",
                    "4ff9c550-0a07-596c-bbfe-3f17bc5c2fde",
                    "64a6c11b-4773-530e-aa18-9cec6915cb8e",
                    "CODE",
                    "CODE",
                    "d980a80d-05a1-53b4-812c-033b1338a5b1",
                    "d99b6a18-2b59-5877-bf5f-408759b82ee1",
                    "fbec47f2-5771-5a29-9d5c-e6ddb4cbca09"
                ]
                }
    ],
    "Message_Type": null,
    "Error": 0,
    "Errors": null,
    "Checksum": null
}
```


3.11.1.5.6 Query Type ID = 6 (Vehicle)

This request allows to query events by vehicle license plate. Only 1 license plate can be queried in one request.

This request is only available to Competent Authorities.

Request

Query Message – request					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Req	Block of basic information elements	Component << Basic Information Request >>	S	M	Message_Type = LUQ
Query_UserID	Unique user identifier	Text(5000)	S	M	OAUTH2 ClientID (for ID Issuers / Primary Providers)
Query_Type	Query type description	Integer	S	M	6
Query_Elements	List of elements	Vehicle ID	M	M	Limited to 1 Vehicle

Response

Vehicle Element JSON Object

Field	Description	Data Type	Cardinality	Values
Transport_vehicle	Identification of the mode of transport (i.e. number plates, train number, plane/flight number, ship name or other identification)	Text(5000)	S	'n/a' is permitted value if Transport_mode = 0 and product movement takes place between adjacent facilities and is delivered manually
Event_List	List of Event recallcodes allowing the lookup of the different events	Text(Array of strings; internal limit = 5000)	M	

Query_Type 6 request sample

```
{
"Message_Type":"LUQ"
"Code": null,
"Query_Type": 6,
"Query_Elements": ["PLATENUMBER1"]
}
```

Query_Type 6 Successful response sample

HTTP Status 200

Query_Type 6 – Query_Result					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
Query_Result	List of found Vehicle elements	Vehicle Element JSON Object	M	M	

3.11.2 LUP – Download Offline flat file

3.11.2.1 Description

Allows download of the generated Offline flat file. This method is only available to Competent Authorities.

3.11.2.2 Description of the fields

Download Offline flat file – request

Field	Description	Data Type	Cardinality	Priority	Values
BasicInfo_Req	Block of basic information elements	Component << Basic Information Request >>	S	M	Message_Type = LUP
Filter	Configurable filter to allow partial file download	Text(1025)	S	O	

3.11.2.3 Response

Download Offline flat file – response					
	Description	Data Type	Cardinality	Priority	Values
BasicInfo_Resp	Block of basic information elements	Component << Basic Information Response >>	S	M	Message_Type = LUP
URL to download ZIP file	Zip file download URL	Text(5000)	S	M	
Password	The password used to protect the zip file	Text(5000)	S	M	

3.11.2.4 Request sample

upUI

```
{
"Message_Type":"LUP"
"Code": "873345b2-882f-4064-91f0-90669b46c30a",
"Filter": {"IdIssuers" : ["id1","id2"]}
}
```

3.11.2.5 Successful response sample

HTTP Status 200

```
Zip file binary download
```

3.12 Manufacturer interface

3.12.1 LDI Lookup Dispatch Interface

3.12.1.1 Context

Provide the manufacturer the ability to check if dispatches that have been positively accepted by their Primary Repositories has been successfully delivered to the Secondary Repository, allowing for the first Arrival on the Router to work successfully.

3.12.1.2 Scope

This interface ONLY allows the manufacturer to validate the accuracy of the reporting for the dispatches sent to their Primary Repository at Secondary Repository level through the Router.

The interface will not cover and therefore will not provide a response to

- Distributors dispatch events.

3.12.1.3 Approach

The Recall code validation. The Manufacturer will be able to

- Retrieve the **status of the dispatch** on the secondary repository. Allowing the confirmation that the primary has processed the dispatch message and transmitted it successfully to the Secondary repository.
- Confirm the **arrival status** side by "simulating" the next arrival process and provide the router response.

3.12.1.4 Response information

The Traceability response to the manufacturer request over the dispatch

3.12.1.4.1 Dispatch status

	Description
0	The recall code of the dispatch message (3.3) is not present in the Secondary repository. The recall code of the dispatch message (3.3) is present but the messages have been recalled successfully.
1	The recall code of the dispatch message (3.3) is present in the Secondary repository and has been successfully processed.

3.12.1.4.2 Arrival status

The system will execute the simulation of the reception validation controls.
The result of the validation controls will be provided in the arrival status.

	Description
null	When Dispatch_status = 0
0	The execution the reception validation controls encountered failed validation. At this point, the corresponding arrival message will result in validation errors.
1	The recall code of the dispatch message (3.3) is present in the Secondary repository and the simulation of the arrival message did not produce any errors.

3.12.1.5 Daily Limit

The limit per manufacturer is set to 30 000 calls per day.
In case the daily limit per manufacturer is exceeded, the call will return a validation error.

HTTP status		
400	FAILED_VALIDATION	In case the maximum number of requests is reached

3.12.1.6 Description of the fields

Application and aggregation envelop event					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Req	Block of basic information elements	Component << Basic Information Request >>	S	M	Message_Type = LDI
Message_Time_long	Message sending Time	Time(L)	S	M	
Dispatch_Code	Dispatch RecallCode		S	M	
Dispatch_EOID	EOID		S	M	

3.12.1.7 Response

upUI application event – response					
Field	**Description**	**Data Type**	**Cardinality**	**Priority**	**Values**
BasicInfo_Resp	Block of basic information elements	Component << Basic Information Response >>	S	M	Message_Type = LDI
Validation_Time	Validation Timestamp		S	M	
Dispatch_Code	Dispatch RecallCode		S	M	
Dispatch_Status			S	M	
Arrival_Status	Response of the simulated arrival related to the dispatch				

3.12.1.8 Request sample

```
{
    "Message_Type": "LDI",
    "Code": null,
    "Dispatch_Code": "873345b2-882f-4064-91f0-90669b46c30a",
    "Dispatch_EOID": "AAAAAAA",
    "Message_Time_Long":"2019-03-20T14:16:45Z"
}
```

3.12.1.9 Successful response sample

HTTP Status 200


```
{
  "Code": "873345b2-882f-4064-91f0-90669b46c30a",
  "Message_Type": "LDI",
  "Dispatch_Code": "873345b2-882f-4064-91f0-90669b46c30a",
  "Validation_Time":"2019-03-20T14:16:45Z",
  "Dispatch_Status": 1,
  "Arrival_Status": {
        "Error": false,
        "Errors": null
  },
  "Error": false,
  "Errors": null,
  "Checksum": "G6HF5H"
}
```


3.6.4.6 Arrival Not Allowed response sample

Response status: 200


```
{
   "Validation_Time": "2021-03-15T12:08:53.0013823+00:00",
   "Dispatch_Code": "870935a0-64fc-55f8-b669-2e6cc09ed68b",
   "Dispatch_EOID": "I85B2J22DN7823851457TPULHSIDZR2735861466DPINXHLRMJ",
   "Dispatch_Status": 1,
   "Arrival_Status": {
      "Error": 1,
      "Errors": [{
         "Error_Code": "ARRIVAL_NOTALLOWED",
         "Error_Descr": "ERROR: VAL_UI_ORD_ARRIVAL, Arrival not allowed because one or
more of the UIs are already IN_STOCK in this or another location
(IN_STOCK/ERP_DELIVERED)",
         "Error_InternalID": null,
         "Error_Data": null
      }]
   },
   "Code": "e7de7c5c-ce96-5753-a2fb-ed655349ce23",
   "Message_Type": "LDI",
   "Error": 0,
   "Errors": null,
   "Checksum": "4d5ddd222480fc8e2f87b2228a3cdde7"
}
```


3.12.1.10 Error response sample

HTTP status		
<< Common response code >>		
400	FAILED_VALIDATION	In case the maximum number of requests is reached
400	FAILED_VALIDATION	The EOID is not valid.
400	FAILED_VALIDATION	The field Dispatch_Code is linked to an event that is not covered by the LDI scope

3.12.1.11 LDI Validations clarification

3.12.1.11.1 Limit Validation

In order to protect the system a limit of 30k call has been defined per primary repository. In case the daily limit is exceeded, a validation error will be returned (http status 400)

HTTP status		
400	FAILED_VALIDATION	In case the maximum number of requests is reached

3.12.1.11.2 Dispatch_EOID existence check

Validate the existence of the EOID present in the request
In case of failure the answer is a 400

HTTP status		
400	FAILED_VALIDATION	The EOID is not valid.

3.12.1.11.3 EDP RecallCode Existence Validation

The first validation is done on the existence of the EDP in the secondary repository. Condition:

- Recall code must exist
- Recall code must exist as received on secondary side ONLY (not router)
- Recall code must be for a Dispatch event EDP (Message_Type = EDP)
- Recall code has not be part of any recall events

in case one of the previous conditions is not met the LDI will return a success response with the Dispatch_Status = 0

HTTP status		
200	SUCCESS	Dispatch_Status = 0 (no dispatch event 3.3 with that RecallCode has been reported to the secondary repository)

3.12.1.11.4 EDP RecallCode LDI Authorisation

This step consist in checking if the EDP can be disclosed.
A number of limitations are implemented to avoid the manufacturer to access the information internal events as well as events from the supply chain.

Conditions:

The LDI is limited to the Dispatch events from a source FID that belongs to the manufacturer (that is consuming the LDI) and the destination FID does not belong to the Manufacturer (meaning it sends the goods to a location outside of his custody)

- EDP must be for EU and have only one destination field filled (Destination_ID2)
- EDP Source Facility ID must be owned by an EO that is linked to a manufacturer
- EDP Source Facility ID and Destination Facility ID must not be the same

HTTP status		
400	FAILED_VALIDATION	The field Dispatch_Code is linked to an event that is not covered by the LDI scope

Note: These conditions might be updated in order to accommodate specific manufacturer distribution configuration including the use of 3PLs. Any update of these conditions will be communicated through the release notes.

4 EU Wide Registry Data Exchange

4.1 Registry

4.1.1 Economic Identifier

Field	Description	Data Type	Priority	Comments
EO_ID	Economic operator's registered ID	EOID	M	
Issuer	Identification number of the ID Issuer solution that has processed the registration	IIID	M	
EO_street	Economic operator's street name and house number (or road number and kilometer)	Text(300)	M, if registered post DataDictionary v2.0 go-live	
EO_municipality	Economic operator's municipality (city, town or village)	Text(100)	M, if registered post DataDictionary v2.0 go-live	
EO_postcode	Economic operator's postal code	Text(50)	M, if registered post DataDictionary v2.0 go-live	
EO_A_info	Additional information on economic operator's address (e.g. location in the shopping mall or industrial area)	Text(100)	O	
EO_Name1	Economic operator's registered name	Text(100)	M	
EO_Name2	Economic operator's alternative or abridged name	Text(100)	O	
EO_Address_Name	Name part of the Address	Text(5000)	O	
EO_Address_StreetOne	Street part of the Address	Text(5000)	M, if registered pre DataDictionary v2.0 go-live	
EO_Address_StreetTwo	Second Element of the Street part of the Address	Text(5000)	O	
EO_Address_City	City	Text(5000)	M, if registered pre DataDictionary v2.0 go-live	
EO_Address_PostCode	Postal Code information	Text(5000)	O	
EO_CountryReg	Economic operator's country of registration	Country	M	See Country

EO_Email	Economic operator's email address; used to inform about registration process, incl. subsequent changes and other required correspondence	Text(80) (Regex protected)	M	
VAT_R	Indication of the VAT registration status	Boolean	M	0–No VAT registration 1– VAT number exists
VAT_N	Economic operator's VAT number	Text(20)	M if VAT_R = 1	
TAX_N	Economic operator's tax registration number	Text(20)	M if VAT_R = 0	
EO_ExciseNumber1	Indication if the economic operator has an excise number issued by the competent authority for the purpose of identification of persons/premises	Boolean	M	0– No SEED number 1– SEED number exists
EO_ExciseNumber2	Economic operator's excise number issued by the competent authority for the purpose of identification of persons/premises	SEED	M, if EO_Excise Number1 = 1	
OtherEOID_R	Indication if the economic operator has been allocated an identifier by another ID Issuer	Boolean	M	0– No 1– Yes
OtherEOID_N	Economic operator identifier codes allocated by other ID Issuers	Text	M if OtherEOID _R = 1	List of EOIDs
Reg_3RD	Indication if the registration is made on behalf of a retail outlet operator not otherwise involved in the tobacco trade	Boolean	M	0– No 1– Yes
Reg_EOID	Identifier of the economic operator that acts on behalf of a retail outlet operator not otherwise involved in the tobacco trade	EOID	M if Reg_3RD = 1	
EO_OtherID	Optional identifier	Text(50)	O	
EO_Importer_Index	Optional Importer Index	Text(50)	O	
EO_CODE	Economic operator's confirmation code provided in response to the registration of economic operator	EO_CODE	M	
Active	If the EO is active	Boolean	M	
Technical_Owner	The IIID that has the ownership of the record.	IIID	M	

4.1.2 Facility

Field	Description	Data Type	Priority	Comments
EO_ID	Economic operator identifier code	EOID	M	(FK)
F_ID	Facility code from the RFA code issuer call	FID	M	(PK)
F_street	Facility's street name and house number (or road number and kilometer)	Text(300)	M, if registered post DataDictionary v2.0 go-live	
F_municipality	Facility's municipality (city, town or village)	Text(100)	M, if registered post DataDictionary v2.0 go-live	
F_postcode	Facility's postal code	Text(50)	M, if registered post DataDictionary v2.0 go-live	
F_A_info	Additional information on facility's address (e.g. location in the shopping mall or industrial area)	Text(100)	O	
PrevFID_B	Indication if the facility was acquired from another operator and had already a facility identifier code	Boolean	M, if registered post DataDictionary v2.0 go-live	0 – No (first time registration) 1 – Yes
PrevFID_ID	Previous facility identifier used by the former operator of the facility	FID	M, if PrevFID_B = 1	
F_Address_Name	Name of the address	Text(5000)	O	
F_Address_StreetOne	Street part of the Address	Text(5000)	M, if registered pre DataDictionary v2.0 go-live	
F_Address_StreetTwo	Second Element of the Street part of the Address	Text(5000)	O	
F_Address_City	City	Text(5000)	M, if registered pre DataDictionary v2.0 go-live	
F_Address_PostCode	Postal Code information	Text(5000)	O	
F_Country	Facility's country	Country	M	See Country
F_Type	Type of facility	Integer	M	See FacilityType
F_Type_Other	Description of other facility type	Text(5000)	M, if F_Type = 4	
F_Status	Indication if a part of the facility has a bonded warehouse status	Boolean	M	0– No 1– Yes
F_ExciseNumber1	Indication if the facility has an excise number issued by the competent authority for the purpose of identification of persons/premises	Boolean	M	0– No SEED number 1– SEED number exists

F_ExciseNumber2	Facility's excise number issued by the competent authority for the purpose of identification of persons/premises	SEED	M, if F_ExciseNumber1 = 1	
OtherFID_R	Indication if the facility has been allocated an identifier by another ID Issuer	Boolean	M	0– No 1– Yes (possible only for non-EU facilities)
OtherFID_N	Facility identifier codes allocated by other ID Issuers	Text	M if OtherFID_R = 1	List of FID
Reg_3RD	Indication if the registration is made on behalf of a retail outlet operator not otherwise involved in the tobacco trade	Boolean	M	0 – No 1 – Yes (possible only if F_Type = 3)
Reg_EOID	Identifier of the economic operator that acts on behalf of a retail outlet operator not otherwise involved in the tobacco trade	EOID	M if Reg_3RD = 1	
latitude	GPS coordinates, latitude value	Decimal	O	Optional information described in the revised Annex II
longitude	GPS coordinates, longitude value	Decimal	O	Optional information described in the revised Annex II
Active	If the facility is active	Boolean	M	
Technical_Owner	The IIID that has the ownership of the record.	IIID	M	

4.1.3 Manufacturing machine

Field	Description	Data Type	Priority	Comments
M_ID	Machine identifier received from the RMA request made to the code issuer.	MID	M	(PK)
F_ID	Facility identifier code	FID	M	(FK)
PrevMID_B	Indication if the object of this request was already registered, e.g. in relation to another machine identifier code	Boolean	M, if registered post DataDictionary v2.0 go-live	0 – No (first time registration) 1 - Yes
PrevMID_ID	Previous machine identifier used for the object of this request	MID	M, if PrevMID_B = 1	
M_entirety	Indication if this request concerns the machine (v. a part of thereof)	Boolean	M, if registered post DataDictionary v2.0 go-live	0 – No (machine part) 1 – Yes (machine)

P_Producer	Part's producer	Text(20)	M, if M_entirety = 0	
P_Model	Part's model	Text(20)	M, if M_entirety = 0	
P_Number	Part's serial number	Text(20)	M, if M_entirety = 0	
P_Mobile	Indication if this part is intended to be used with multiple machines (fixed v. mobile part)	Boolean	M, if M_entirety = 0	0 – No (fixed part) 1 – Yes (mobile part)
P_ATD1	Indication if an anti-tampering device in the sense of Article 2(7) records the functioning of this part	Boolean	M, if M_entirety = 0	0 – No 1 – Yes
P_ATD2	Anti-tampering's device serial number	Text(100)	M, if M_entirety = 0 and P_ATD1 = 1	
P_Description	Part's description explaining its technical function	Text(500)	O	
M_Producer	Machine producer	Text(20)	M, if M_entirety = 1	
M_Model	Machine model	Text(20)	M, if M_entirety = 1	
M_Number	Machine serial number	Text(20)	M, if M_entirety = 1	
M_parts	Indication if the machine consists of multiple separately identifiable parts	Boolean	M, if M_entirety = 1 and registered post DataDictionary v2.0 go-live	
M_plist	List of the identifiable parts	MID	M, if M_entirety = 1 and M_parts = 1	List of MIDs (parts)
M_ATD	Serial number of the anti-tampering device in the sense of Article 2(7)	Text(100)	M, if M_entirety = 1 and M_parts = 0 and registered post DataDictionary v2.0 go-live	
M_Capacity	Maximum capacity over 24hour production cycle expressed in unit packets	Integer	M, if M_entirety = 1	
Active	If Machine is active	Boolean	M	
Technical_Owner	The IIID that has the ownership of the record.	IIID	M	

4.2 Flat Files

4.2.1 Flat File type I Format

4.2.1.1 Overview

The Flat File type I format contain the following files.

- ProductLookup
- ManufacturerLookup
- RegularExpression

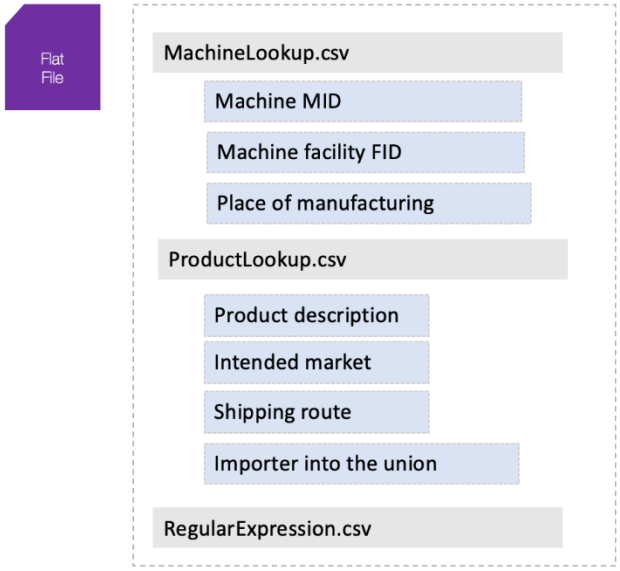


Figure 12 Compact Flat File Structure

4.2.1.2 ProductLookup

Field	Description	Data Type	Priority	Comments
ProductLookupId	The part of the code used for product lookup	Text(20)	M	
TP_ID	Tobacco product identifier used in the EU-CEG system	TPID	M, if Intended_Market is an EU country	
P_Type	Type of tobacco product	int	M	
P_OtherType	Description of other type of tobacco product	Text(200)	M, if P_Type = 12	
P_CN	Combined Nomenclature (CN) code	Text(200)	Optional	
TP_PN	Tobacco product number used in the EU-CEG system	PN	M, if Intended_Market is an EU country	
P_Brand	Brand of tobacco product	Text(200)	M	
P_OtherID	Optional Product ID	Text(20)	O	
P_weight	Average gross weight of unit packet, including packaging, in grams with 0,1 gram accuracy	Decimal	M	
P_SubType_Exist	Indicates if the product "subtype name" exists. Subtype name provides further product	Boolean	M	

	identification beyond a product's brand name.			
P_SubType_Name	The product "subtype name" (if any) as marketed on its intended market	Text(200)	M, if P_SubType_Exist = 1	
P_units	The number of individual units in the unit packet (number of sticks in the package).	Integer	M, if P_Type = 1 or 2 or 3	
Intended_Market	Intended country of retail sale	Country	M	
Intended_Route1	Indication if the product is intended to be moved across country borders with terrestrial/water/air transport	Boolean	M	
IntendedShipmentRoute	The first country of terrestrial/water/air transport after the product leaves the Member State of manufacturing or the Member State of importation established on the basis of a check point on the land border, next seaport or next airport respectively	Country	M, if Intended_Route1 = 1	
ImporterIntoEU	Indication if the product is imported into the EU	Boolean	M	
ImporterEOID	EOID of the importer when applicable	EOID	O	Fully optional field – THIS FIELD can be omitted
Active	If the lookup entry is active	Boolean	M	

4.2.1.3 ManufacturerLookup

Field	Description	Data Type	Priority	Comments
ManufacturerLookupId	The manufacturer ID	Text(20)		
MID	Machine identifier code	MID		MID can be empty for aUI
FID	Factory identifier code	FID		
Active	If the lookup entry is active	Boolean		

4.2.1.4 RegularExpression (Optional)

Field	Description	Data Type	Comments
RegularExpression	The regular expression using tags to flag potential lookup extracted from code. Tags : TPID / MID / FID / IID	Text(5000)	

Short example of a regular expression containing 5 alphanumeric for IID, 3 and 3 alphanumeric for FID & MID, 8 alphanumeric for TPID (might be

the GTIN encoded in 8 symbols), 10 symbols for the "serial number", and 8 digits for the timestamp (note the "?" that makes it optional).
Ex: ^(?<IID>\w{5})(?<FID>\w{3})(?<MID>\w{3})(?<TPID>\w{8}).{10}\d{8}?$

4.2.2 Flat File type II format

4.2.2.1 Flat File type II Algorithm overview

Each ID Issuer should provide software that can convert any upUI that it generated into the set of values for index variables that can be looked up in the following lookup tables

- ProductLookup.csv
- MachineLookup.csv
- FacilityLookup.csv
- ImporterLookup.csv
- TargetMarketLookup.csv
- RouteLookup.csv

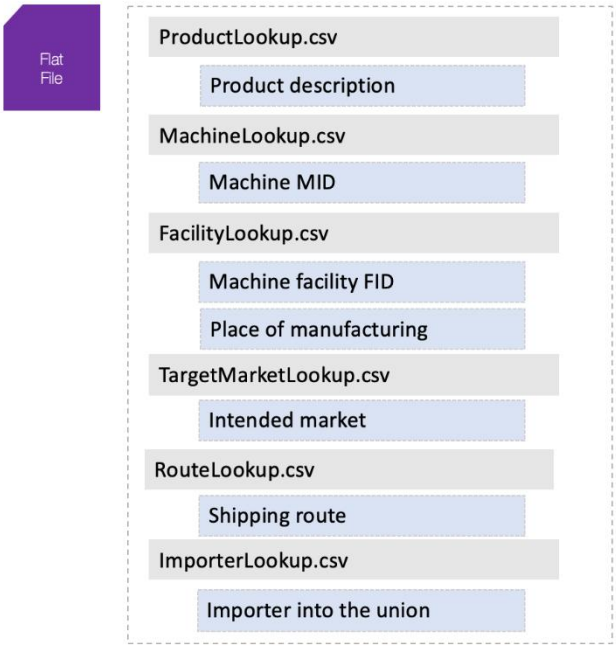


Figure 13 Granular Flat File Structure

4.2.2.2 ProductLookup.csv

Field	Description	Data Type	Priority	Comments
ProductLookupId	The product lookup ID	Text(20)		
TP_ID	Tobacco product identifier used in the EU-CEG system	TPID	M, if Intended_Market is an EU country	
P_Type	Type of tobacco product	int		
P_OtherType	Description of other type of tobacco product	Text(200)		
P_CN	Combined Nomenclature (CN) code	Text(200)		
TP_PN	Tobacco product number used in the EU-CEG system	PN		

P_Brand	Brand of tobacco product	Text(200)		
P_OtherID	Optional Product ID	Text(20)		
P_weight	Average gross weight of unit packet, including packaging, in grams with 0,1 gram accuracy	Decimal		
P_SubType_Exist	Indicates if the product "subtype name" exists. Subtype name provides further product identification beyond a product's brand name.	Boolean		
P_SubType_Name	The product "subtype name" (if any) as marketed on its intended market	Text(200)	M, if P_SubType_Exist = 1	
P_units	The number of individual units in the unit packet (number of sticks in the package).	Integer	M, if P_Type = 1 or 2 or 3	
Active	If the lookup entry is active	Boolean		

4.2.2.3 MachineLookup.csv

Field	Description	Data Type	Priority	Comments
MachineLookupId	The machine lookup ID	Text(20)		
MID	MID of manufacturing machine – for use with Machines.csv registry file	MID		This field might be empty.
FID	FID of manufacturing facility – for use with Machines.csv registry file	MID		
Active	If the lookup entry is active	Boolean		

4.2.2.4 FacilityLookup.csv

Field	Description	Data Type	Priority	Comments
FacilityLookupId	The facility lookup ID	Text(20)		
FID	FID of manufacturing facility – for use with Machines.csv registry file	FID		
Active	If the lookup entry is active	Boolean		

4.2.2.5 ImporterLookup.csv

Field	Description	Data Type	Priority	Comments
ImporterLookupId	The imported lookup ID	Text(20)		
ImporterIntoEU	Indication if the product is imported into the EU	Boolean		
Importer_EOID	EOID of the imported	EOID		
Importer_Index		Text(2)		
Active	If the lookup entry is active	Boolean		

4.2.2.6 TargetMarketLookup.csv

Field	Description	Data Type	Priority	Comments
ImporterLookupId	The imported lookup ID	Text(20)		
Intended_Market	Intended country of retail sale	Country		
Active	If the lookup entry is active	Boolean		

4.2.2.7 RouteLookup.csv

Field	Description	Data Type	Priority	Comments
ImporterLookupId	The imported lookup ID	Text(20)		
IntendedShipmentRoute	The first country of terrestrial/water/air transport after the product leaves the Member State of manufacturing or the Member State of importation established on the basis of a check point on the land border, next seaport or next airport respectively	Country		
Active	If the lookup entry is active	Boolean		

4.3 Offline Flat File Data Exchange

Offline flat files are the output of all the flat files sent by the ID Issuers.

4.3.1 audit.csv

Field	Description	Data Type	Comments
Key	key	Text(50)	
Value	Value	Text(255)	

4.3.2 IdIssuers.csv

Field	Description	Data Type	Comments
Issuer_Prefix	ID issuer's prefix in accordance with ISO15459-2:2015	IIID	
Issuer_Name	The name of the ID issuer	Text(255)	
Issuer_Country	Country for which the id issuer operates for.	Country	
Issuer_FlatfileType	The type of the flat file used by the ID Issuer	Text(1)	1.- flatfile type 1 2.- flatfile type 2

4.3.3 countries.csv

See section 2.6.1

4.3.4 facilitytype.csv

See section 2.6.5

4.3.5 tobaccoproducttype.csv

See section 2.6.11

4.3.6 transportmode.csv
See section 2.6.12

4.3.7 EconomicIdentifiers.csv
See section 4.1.1

4.3.8 Facilities.csv
See section 4.1.2

4.3.9 Machines.csv
See section 4.1.3

4.3.10 Flat File type I

4.3.10.1 ProductLookup.csv
See section 4.2.1.2

4.3.10.2 MachineLookup.csv
See section 4.2.1.3

4.3.10.3 RegularExpression.csv
See section 4.2.1.4

4.3.11 Flat File type II

4.3.11.1 ProductLookup.csv
See section 4.2.2.2

4.3.11.2 MachineLookup.csv
See section 4.2.2.3

4.3.11.3 FacilityLookup.csv
See section 4.2.2.4

4.3.11.4 ImporterLookup.csv
See section 4.2.2.5

4.3.11.5 TargetMarketLookup.csv
See section 4.2.2.6

4.3.11.6 RouteLookup.csv
See section 4.2.2.7

4.3.12 Filename
YYYYMMDD_OFFLINE.zip

5 List of Error Codes

5.1 Security errors

HTTP status	Error Code	Text Description
401	INVALID_OR_EXPIRED_TOKEN Related control: VAL_SEC_TOKEN	Error Descr: The incoming token is not valid or expired Comment: The security token has expired and should be renewed.

5.2 Processing errors

HTTP status	Error Code	Text Description
400	FAILED_VALIDATION Related control: VAL_FIE_REF VAL_MSG_XML	Error Descr: The field <XXX> should contain a valid <YYY> Error Descr for Circular Reference issue: The message contains UI values that form a circular reference E.g.: The value for the field Aggregation_Type (**XXX**) is not in the defined set of values for AggregationType (**YYY**) (1 2 or 3) Comments: The values must match the values included in the set defined in the Data Dictionary. Concerning circular reference error: the UI mentioned in the event message is a parent of another UI present in the same message.
400	REQUIRED_FIELD_FAILED_VALIDATION Related controls: VAL_FIE_MAN VAL_MSG_JSON	Error Descr: The field <XXX> is required. VAL_FIE_MAN: Data missing in Mandatory field. E.g.: field = “” VAL_MSG_JSON: Missing mandatory field. E.g.: field = null or not present in the JSON
400	INVALID_MESSAGE_TYPE Related control: VAL_MSG_TYPE	Error Descr: Message type is unknown Comment: The type of Message you are using is not present in the Data Dictionary.
400	INVALID_SIGNATURE Related control: VAL_SEC_HASH	Error Descr: Hash information not matching the message signature Comment: The validation of the HASH of the body of the message doesn’t match the transmitted HASH information in the header X-OriginalHash.
400	MAX_LENGTH_FAILED_VALIDATION Related control: VAL_MSG_JSON	Error Descr: The field <XXX> should be a value with maximum length of <Y> Comment: The message doesn’t follow the specifications defined in the Data Dictionary. The number of characters must remain under the max length.

400	MIN_LENGTH_FAILED_VALIDATION Related control: VAL_MSG_JSON	Error Descr: The field <XXX> should be a value with minimum length of <Y> Comment: The message doesn't follow the specifications defined in the Data Dictionary. The number of characters must remain above the min length.
400	ENTRY_LENGTH_FAILED_VALIDATION Related control: VAL_MSG_JSON	Error Descr: The field <XXX> should be a 2-dimensional array where each row contains <Y> elements Comment: The message doesn't follow the specifications defined in the Data Dictionary.
400	INVALID_INPUT_FORMAT Related controls: VAL_MSG_JSON VAL_FIE_FORMAT	Error Descr: see examples below Example for 3.3 message (dispatch - EDP): n/a is a permitted value for the field 'Transport_vehicle' only if Transport_mode = 0 Example for recall messages: Please note that a recall can not be performed on <XXX> messages. Comments: The message doesn't follow the specifications defined in the Data Dictionary. The body of the message contains at least one field in wrong format or does not correspond to a valid JSON message. Recall messages in particular cannot be performed on 2.1 messages (IRU), 2.2 messages (IRA) and 2.3 messages (IDA)
400	PAYLOAD_NOT_UNIQUE Related control: VAL_MSG_DUPLICATE	Error Descr: The message should contain a payload which was not previously used Comment: The system already processed the same payload delivered in an earlier message. You cannot resend the same payload.
400	EXCISE_NUMBER_NOT_VALID Related control: VAL_MSG_JSON	Error Descr: The field 'EO_ExciseNumber2' should contain a valid excise number Comment: The format of the field EO_ExciseNumber2 doesn't match the Data Dictionary.
400	NON_COMPATIBLE_UIS Related control: VAL_MSG_JSON	Error Descr: the field 'upUI_2' should be compatible with 'upUI_1' Comment: Activation failed as ordered list of UIs with timestamp, did not match short UIs.
400	NOT_THE_SAME_NUMBER_OF_ITEMS Related control: VAL_MSG_JSON	Error Descr: For 3.1 message (activation - EUA): The field 'upUI_1' should contain the same number of items as 'upUIs_2' For 4.1 message (invoice - EIV): The field 'Product_Items_2' should contain the same number of items as 'ProductIdentifiers' And The field 'Product_Price' should contain the same number of items as 'ProductIdentifiers' For IRU message: The field "upUI_i" and "upUI_s" must match in length. Comments: 3.1 message (activation - EUA): Activation failed as number of UI with timestamp, did not same number as short UIs.

		4.1 message (invoice - EIV): message failed as the items contained int the fields 'Product_Items_2' and/or 'Product_Price' are/is not the same as the number of items in 'ProductIdentifiers'.
400	MULTIPLE_UID Related control: VAL_UI_MULT_MSG	Error Descr: The field 'upUI'/'aUI' contains duplicate values Comment: Multiple duplicate UI present in the message lists. Message must contain only one occurrence of the same UI.
500	SYSTEM_ERROR	Error descr: Null Comment: The internal error ID should be provided to Dentsu support if required.
400	TIME_2019 Related Control: VAL_TIME_2019	Error Descr: The field 'Event Time' or 'Message Time Long' is earlier than May 2019. No reported is allowed before that date.
400	TIME_72 Related Control: VAL_TIME_72	Error Descr: The field 'Event Time' or 'Message Time long' is reported in the future.

5.3 Validation Warning

HTTP status	Error Code	Text Description
299	OPERATION_WITHIN_24_HOURS Related control: VAL_EVT_24H	Warning Descr: Reporting events should be performed within 24 hours of the occurrence of the event (except dispatch and trans-loading events) Comment: You received this warning because this message has been reported late, i.e. more than 24 hours after the event time. Please note that the reporting time frame will be reduced to 3 hours starting from May 2028.
299	SHIPMENT_WITHIN_24_HOURS Related control: VAL_EVT_TIME	Warning Descr: The date/Time provided in the field 'Event_Time' should not be more than 24 hours ahead of the actual reporting time Comment: Dispatch and transloading events have to be reported within a time frame of 24 hours prior to the occurrence of the movement. Control is based on the "actual date – Event_Time" time difference.
299	UI_SEQUENCE_WARNING Related control: VAL_UI_ORD_SEQUENCE_WARNING	Warning Descr: <action> is not expected nor allowed when state is generated/deactivated/implicitly disaggregated Error Data: list of UI Comment: This error happens when you are trying to generate upUIs which are already existing in the Secondary, or to deactivate upUIs/aUIs already deactivated, or to deactivate aUIs already implicitly disaggregated.
299	UI_NOT_EXIST Related controls: VAL_UI_EXIST_UPUI VAL_UI_EXIST_AUI	Warning Descr: The field 'upUIs/aUIs' must contain elements that are already recorded and in one of the following states: Activated, Generated Error Data: list of UI Comment: This error is implemented for logistic actions, excluding UI activation.

		The most common reasons causing this message to occur are: a) the UI in question is a pre TPD/legacy UI never reported to the Secondary, b) the UI in question is a new UI not yet reported to the Secondary, c) The UI in question has not been encoded or decoded in line with the existing formatting instructions VAL_UI_EXIST_AUI Logistic action is not expected nor allowed if a UI does not exist (has not been part of an EPA message as a parent). VAL_UI_EXIST_UPUI Logistic action is not expected nor allowed if a UI does not exist (has not been part of any IRU message).

5.4 Validation errors

HTTP status	Error Code	Text Description
400	CANNOT_ROUTE Related control: CANNOT_ROUTE	Error Descr: This message cannot be routed Comment: The IRU message cannot be routed to the corresponding primary repository. The ID Issuer should contact the primary provider and ensure that the EOID is correctly configured and pointing to the corresponding primary repository.
400	UI_NOT_VALID Related controls: VAL_UI_EXIST_UPUI_SEQ VAL_UI_EXIST_AUI_SEQ	Error Descr: <action> is not expected nor allowed when the pack has not been part of an application or an aggregation Error Data: list of UI Comment: Action on upUI is not expected nor allowed when the upUI has not been applied. Action on aUI is not expected nor allowed when the aUI has not been aggregated
400	UIS_APPLICATION_ERROR Related controls: VAL_UI_EXIST_APP VAL_UI_DUPLICATE_APP	Error Descr for 3.1 message (activation - EUA): Unique Identifier application on unit pack is not expected nor allowed when pack does not exist or has been reported to be already applied Error descr for 2.3 message (deactivation - IDA): Deactivation of upUI/aUI is not expected nor allowed when upUI/aUI does not exist Error Data: list of UI Comment: this error is generated when trying to activate/deactivate UIs which are not recorded in the Secondary or which have already received an application event.
400	UI_DEACTIVATED Related controls: VAL_UI_ORD_REACTIVATION VAL_UI_ORD_DEACTIVATED	Error Descr for 3.1 messages (activation - EUA): Unique identifier application on unit pack is not expected nor allowed once the unique identifier has been deactivated Error Descr for other logistic actions: <Action> is not expected nor allowed once the unique identifier has been deactivated Error Data: list of UI

		Comment: The action you want to perform is not expected nor allowed after the UI has been deactivated with the 2.3 message.
400	MULTIPLE_AGGREGATION Related control: VAL_UI_ORD_AGG_MULT	Error Descr: Aggregation is not expected nor allowed when the pack is considered as aggregated or implicitly disaggregated Error Data: list of UI Comment: Multiple aggregation identified for an aUI (as a parent) without having an explicit disaggregation of this aUI.
400	UI_ALREADY_DISAGGREGATED Related controls: VAL_UI_ORD_DISAGG VAL_UI_ORD_IMPLDISAGG	Error Descr: <action> is not expected nor allowed once the pack has been made explicitly available for aggregation after disaggregation Error Data: list of UI Comment: an aUI that has been disaggregated (explicitly or implicitly) cannot be part of any product movement prior of being aggregated.
400	LOCATION_MISMATCH Related controls: VAL_UI_ORD_AGG_FID VAL_UI_ORD_DISPATCH	Error Descr: the FID must match with the location state of the reported unique identifiers Error Data: list of UI Comment: VAL_UI_ORD_AGG_FID All the goods must have been produced or reported to be in stock in the location where they are aggregated / disaggregated. VAL_UI_ORD_DISPATCH All the goods must have been produced or reported to be in stock in the location from which they are dispatched.
400	FID_MISMATCH Related Control: VAL_UI_FID_APP	Error Descr: the FID must match the FID specified in the meta data of unique identifiers Comment: UI application in this location is not expected nor allowed as this location is not the one of the 2.1 message. Error Data: list of UI
400	ARRIVAL_NOTALLOWED Related Control: VAL_UI_ORD_ARRIVAL VAL_UI_ORD_ARRIVAL_RETURN	Error Descr: <action> not expected nor allowed when the pack is considered as in stock within EU facility following an application, arrival or return Error Data: list of UI Comment: arrival in EU facility is not expected nor allowed when the UI is considered as 'in stock' in a non-retail facility , meaning that a UI must have been part of a prior reported dispatch or transloading event before having an arrival It is not allowed to arrive multiple times the same UIs. Exceptions: Imported products are allowed for arrival in EU facility without having any prior dispatch or transloading
400	UI_SEQUENCE_ERROR Related control: VAL_UI_ORD_SEQUENCE	Error descr: <action> is not expected nor allowed when/once <state> Error Data: list of UI Comment: this is a generic sequence validation error caused by a message sent out of the permitted sequence.

		In particular, an implicit disaggregation in transit is not allowed, meaning that the reporting of an arrival must be done with reference to the same UI(s) as reported for the purpose of preceding dispatch/transloading. Example for a 3.3 message (dispatch): *Dispatch from EU for delivery to retail destination is not expected nor allowed when the pack has been reported as dispatched for delivery to retail destination (DISPATCHED_EU_FIXED_QT_RETAIL/EDP_EU_FIXED_QT_RETAIL)* In this case, the dispatch is failing because it is including UIs which are in state dispatched Example for a 3.7 message (report of delivery through Vending Van) *Report of EU delivery with a vending van to retail outlet destination is not expected nor allowed when the pack has not been previously reported as dispatched for VV delivery (DISPATCHED_EU_FIXED_QT_RETAIL/EVR)* In this case, the delivery through Vending Van is not allowed because the previous event is a dispatch to a facility with delivery of fixed quantity. It should have been a dispatch with delivery with Vending Van.
400	UI_EXPIRED Related control: VAL_UI_EXPIRY	Error Descr: Some or all unique identifiers listed in the message have expired Error Data: list of UI Comment: Validation if the application or the aggregation date doesn't exceed the 6 months period after the generation of unique identifiers by the ID issuers.
400	EOID_NOT_EXIST_OR_ACTIVE Related controls: VAL_ENT_EXIST_EOID VAL_ENT_ACTIVE_EOID	Error Descr: EOID mentioned in the field 'EO_ID' is not marked as active in the repository Error Data: EOID Comment: The field 'EO_ID' must contain elements that are already recorded and active.
400	FID_NOT_EXIST_OR_ACTIVE Related controls: VAL_ENT_EXIST_FID VAL_ENT_ACTIVE_FID	Error Descr: FID mentioned in the field 'F_ID' is not marked as active in the repository Error Data: FID Comment: The field 'F_ID' must contain elements that are already recorded and active.
400	MID_NOT_EXIST_OR_ACTIVE Related controls: VAL_ENT_EXIST_MID VAL_ENT_ACTIVE_MID	Error Descr: MID mentioned in the field 'M_ID' is not marked as active in the repository Error Data: MID Comment: The field 'M_ID' must contain elements that are already recorded and active.
400	FID_NOT_RELATED_TO_EOID Related control: VAL_ENT_REL_EOID_FID	Error Descr: There is no existing relation recorded between FID mentioned in the field 'F_ID' and a EOID Comment: There is no existing relation recorded between FID mentioned in the field "F_ID" and an EOID.
400	MID_NOT_RELATED_TO_FID Related control: VAL_ENT_REL_ FID_MID	Error Descr: There is no existing relation recorded between MID mentioned in the field 'M_ID' and a FID Comment: there is no existing relation recorded between MID mentioned in the field "M_ID" and a FID.
400	CODE_NOT_UNIQUE Related controls: VAL_MSG_CODE_DUPLICATE VAL_RECALL_EXIST	Error Descr: The field 'Recall_Code' must contain a value which was not previously used

		Comment: The recall code provided has been used before. For recall messages, you cannot recall twice on the same recall code if the previous recall message succeeded.
400	CODE_NOT_EXIST Related control: VAL_RECALL_EXIST	Error Descr: The field 'Recall_Code' must contain elements that are already recorded Comment: The recall code provided has not been found into the secondary.
400	RECALL_NOT_LAST_EVENT Related control: VAL_RECALL_LAST	Error Descr: Please note that a recall can only be performed on valid messages that referred to UIs which were not later used in other messages. Error Data: list of pair (UI @ Previous RecallCode) Comment: recalls can only be performed on last event.