The Global Language of Business

FIT Master Data Message Business Message Standard

Release 1.0, Ratified, May 2019

Document Summary

Document Item	Current Value
Document Name	FIT Master Data Message Business Message Standard
Document Date	May 2019
Document Version	1.0
Document Issue	
Document Status	Ratified
Document Description	

Log of Changes

Release	Date of Change	Changed By	Summary of Change
1.0	May 2019	Eric Kauz, Mark Van Eeghem, Maneesh Naganand	Original publication based on GSMP WRs 18-277 & 19-078

Disclaimer

GS1®, under its IP Policy, seeks to avoid uncertainty regarding intellectual property claims by requiring the participants in the Work Group that developed this **FIT Master Data Message Business Message Standard** to agree to grant to GS1 members a royalty-free licence or a RAND licence to Necessary Claims, as that term is defined in the GS1 IP Policy. Furthermore, attention is drawn to the possibility that an implementation of one or more features of this Specification may be the subject of a patent or other intellectual property right that does not involve a Necessary Claim. Any such patent or other intellectual property right is not subject to the licencing obligations of GS1. Moreover, the agreement to grant licences provided under the GS1 IP Policy does not include IP rights and any claims of third parties who were not participants in the Work Group.

Accordingly, GS1 recommends that any organisation developing an implementation designed to be in conformance with this Specification should determine whether there are any patents that may encompass a specific implementation that the organisation is developing in compliance with the Specification and whether a licence under a patent or other intellectual property right is needed. Such a determination of a need for licencing should be made in view of the details of the specific system designed by the organisation in consultation with their own patent counsel.

THIS DOCUMENT IS PROVIDED “AS IS” WITH NO WARRANTIES WHATSOEVER, INCLUDING ANY WARRANTY OF MERCHANTABILITY, NONINFRINGEMENT, FITNESS FOR PARTICULAR PURPOSE, OR ANY WARRANTY OTHER WISE ARISING OUT OF THIS SPECIFICATION. GS1 disclaims all liability for any damages arising from use or misuse of this document, whether special, indirect, consequential, or compensatory damages, and including liability for infringement of any intellectual property rights, relating to use of information in or reliance upon this document.

GS1 retains the right to make changes to this document at any time, without notice. GS1 makes no warranty for the use of this document and assumes no responsibility for any errors which may appear in the document, nor does it make a commitment to update the information contained herein.

GS1 and the GS1 logo are registered trademarks of GS1 AISBL.

Table of Contents

1 Introduction

1.1 Objective

The purpose of this document is to create standard messages used for the creation and management of identifier codes for Economic Operators, Facilities, Machines and Unit Level Items. In addition, there is messaging for registry responses and for message recall. It is assumed in this documentation that the messages for creating managing Economic Operators, Facilities and Machines will be APIs that will interact with a registry. The information sent to the registry and received from the registry will be in the JSON format.

The primary input to this document is Annex II (Technical Standards for the Establishment of a Traceability System for Tobacco Products) sections 1-3.

The document does not focus on the user interfaces to the standard APIs besides outlining the data requirements for each message type.

1.2 API Design Principles

This API is modelled as resource-oriented API with *collections* of individually addressable *resources* (the *nouns* of the API). Resources are referenced with their resource names and manipulated via a small set of methods (also known as *verbs* or *operations*). The API is modelled as a resource hierarchy, where each node is either a *simple resource* or a *collection resource*. For convenience, they are often called as a resource and a collection, respectively.

- A resource has some state and zero or more sub-resources. Each sub-resource can be either a simple resource or a collection resource.
- A collection contains a list of resources of the same type.
- Where the desired API functionality does not map to resources and methods, custom methods are used.
- This API currently addresses a simple resource `(e.g. EconomicOperatorRegistration)` and the corresponding collection resource `EconomicOperatorRegistration []`.
- Due to the high volume of data, collections of resources are often provided in paginated form and limits are placed on the maximum resources that can be returned per request (limits are to be determined by registry specification). Page size and page number can be refined via query parameters on the request URL.

The REST style for this API is characterised by:

- HTTP is used as the transport protocol.
- The request is constructed using the HTTP request method suited for that operation (GET, POST, DELETE, PATCH, PUT, etc.).
- The response payload is a JSON response document.
- HTTP response code 200 indicates success. HTTP response codes not beginning with “2” are used to indicate exceptions.

Client applications SHALL make a HTTP request to the service who SHALL respond using JSON formatted data [*JSON*]. Both the client and service SHALL conform to HTTP 1.1 [*HTTP*].

2 Business domain view

2.1 References

Reference name
[HTTP] IETF, “Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing” IETF RFC 7230, June 2014, http://www.ietf.org/rfc/rfc7230.txt

Reference name
[JSON] ECMA International, “The JSON Data Standard Interchange Format, 1st Edition” ECMA Standard ECMA-404, October 2013, http://www.ecma-international.org/publications/files/ECMA-ST/ECMA-404.pdf
Annex II of the Tobacco Products Directive (Annex II)

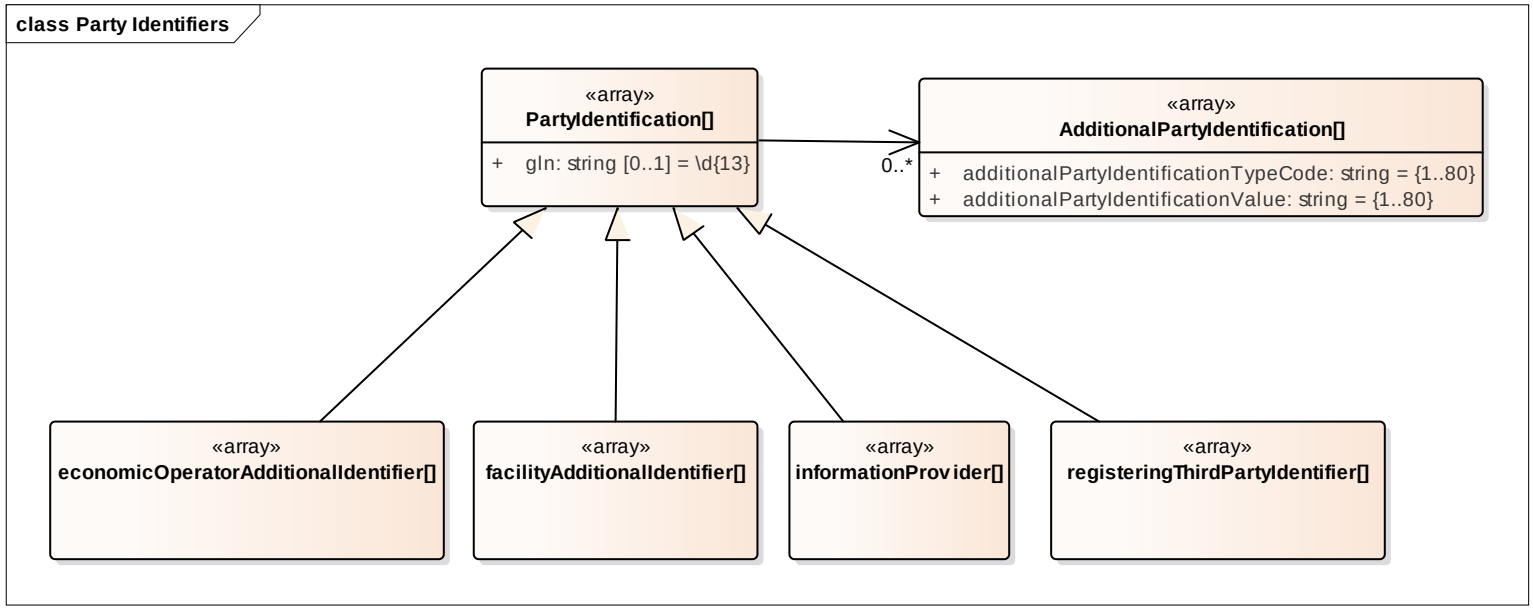
3 Data Definitions

3.1 Common Terms

Property Name	Value/Type	Description
Aggregated Level Unique Identifier (AUI)	string	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) serialisation 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))
GIAI	GIAI	The GIAI is the GS1 identification key used to identify an individual asset. The key is comprised of a GS1 Company Prefix and an individual asset reference.
GLN	string	The Global Location Number (GLN) is the GS1 Identification Key used to identify physical locations or parties. The key is comprised of a GS1 Company Prefix, Location Reference, and Check Digit.
GTIN	string	The GTIN represents a Global Trade Item Number (GTIN). The GTIN type is a 14-character numeric string which may contain a GTIN-8, GTIN-12, GTIN-13, or GTIN-14 as defined in the GS1 General Specifications. When the GTIN type holds a GTIN-8, GTIN-12, or GTIN-13, the GTIN value is padded on the left with zeros to make 14 characters total.
Economic Operator Identifier Code	string	Economic operator identifier code corresponding to the format established by the ID issuer coded with the invariant set of ISO646:1991. Coded as a UIC + GLN.
Exception	Object	An exception result as the result of processing a request.
Facility Identifier	string	The unique identification of a facility. For GS1, this is a UIC + GLN.
Machine Identifier	string	The unique identification of a production machine. For GS1, this is a UIC + GIAI.
Status Reason	Object	The result of processing a request.
Unit packet level unique identifier(L)	string	Unit packet level unique identifier coded with the invariant set of ISO646:1991 and composed of three blocks: (a) ID Issuer's prefix in accordance with ISO15459-2:2015, (b) middle block in the format established by the ID issuer and (c) timestamp following the Data Type: Time(s)
Unit packet level unique identifier(S)	string	Unit packet level unique identifier coded with the invariant set of ISO646:1991 and composed of two blocks: (a) ID issuer's prefix in accordance with ISO15459-2:2015 and (b) serialisation element in the format established by the ID issuer (i.e. UI made visible in the human readable format on the unit packets)

3.1.1 Party Identification

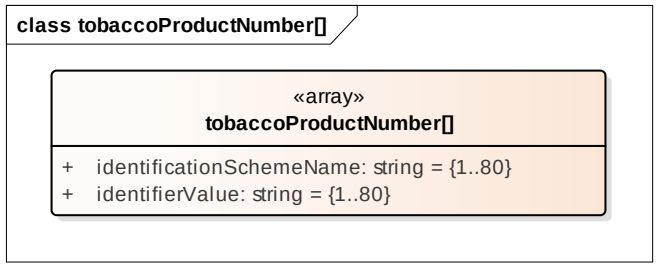
The following diagram and table specify the properties for GS1 Party Identifiers used in the FIT messages. PartyIdentification is a reusable object that allows for either a GLN or an additional identifier. It is required to include a type for the additional identifier value (e.g. EU_VAT_IDENTIFICATION_NUMBER). Party Identification is not directly used but is instead a template for other identifiers such as informationProvider.



Property Name	Value/Type	Multiplicity	Description
PartyIdentification[]	array		Information used to identify a party or location.
gln	string [\d{13}]	0..1	The Global Location Number (GLN) is the GS1 Identification Key used to identify physical locations or parties. The key is comprised of a GS1 Company Prefix, Location Reference, and Check Digit.
AdditionalPartyIdentification[]	array	0..*	Identification of a party by use of a code other than the Global Location Number.

Property Name	Value/Type	Multiplicity	Description
additionalPartyIdentificationValue	string{1..80}	1..1	The value of an identification of a party by use of a code in addition to the Global Location Number.
additionalPartyIdentificationTypeCode	string{1..80}	1..1	Identification of a party by use of a code in addition to the Global Location Number.

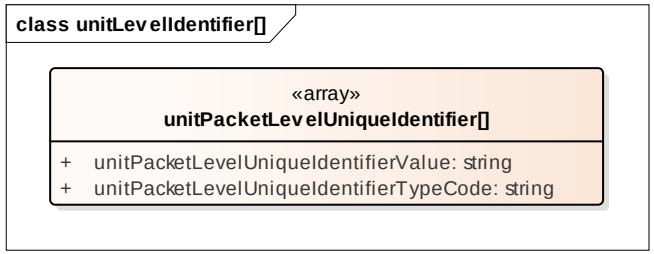
3.1.2 Tobacco Product Number



Property Name	Value/Type	Multiplicity	Description
tobaccoProductNumber[]			
identificationSchemeName	string	1..1	The name of the identification scheme.
identifierValue	string	1..1	The value of the tobacco product number used in the EU-CEG system to identify product presentations. Note: The gtin field should be used if a GTIN is used to identify the tobacco product. See the class "UnitLevelItem"

3.1.3 Unit Level Identifier

The following diagram and table specify the properties for unit level identifier. It is currently assumed that a Unit Level Identifier could be either a Unit packet level unique identifier(L) or Unit packet level unique identifier(S).



Property Name	Value/Type	Multiplicity	Description
unitPacketLevelUniqueIdentifier[]			
unitPacketLevelUniqueIdentifierValue	string	1..1	An identification value for either an aui, apUI(I) or apUI(s).
unitPacketLevelUniqueIdentifierTypeCode	string	1..1	The type of unit level identification for example aUI, apUI(I) and apUI(s).

3.2 Message Objects

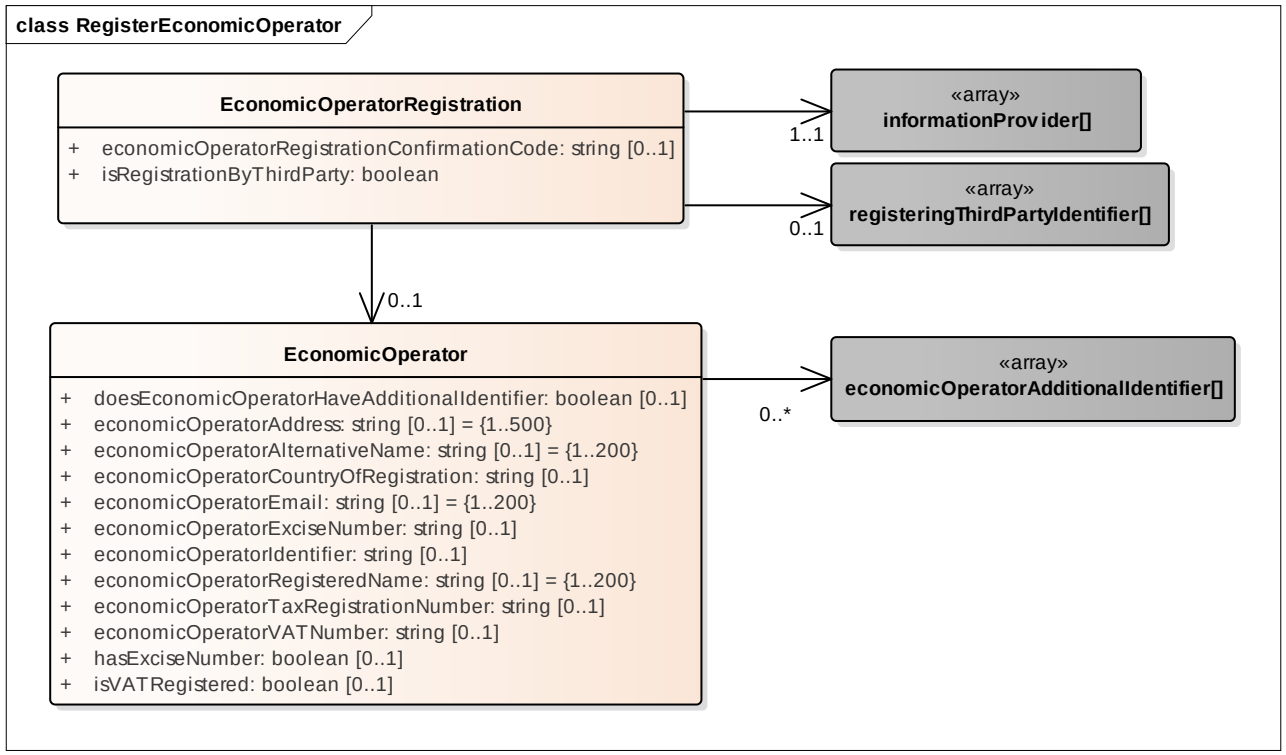
The following diagrams depict objects required to register, correct, deactivate identifiers in a registry.

Note:

- Attribute names used in Annex II were altered to reflect GS1 standard naming conventions (e.g. all attribute names for Booleans are phrased as questions.
- Many messages depicted in Annex II request a Message Type. This attribute was not viewed as necessary to the design since the object name and the HTTP request method is sufficient to identify the messages in all cases.

3.2.1 Economic Operator Registration

The following diagram and table specify the properties to manage an economic operator in a registry. This object can be used to register, correct or deregister an economic operator. Rules regarding what can be populated for each action (e.g. register, correct) can be found in Section 4.2.



Note: Grey boxes are common components and can be found in section 3.1.

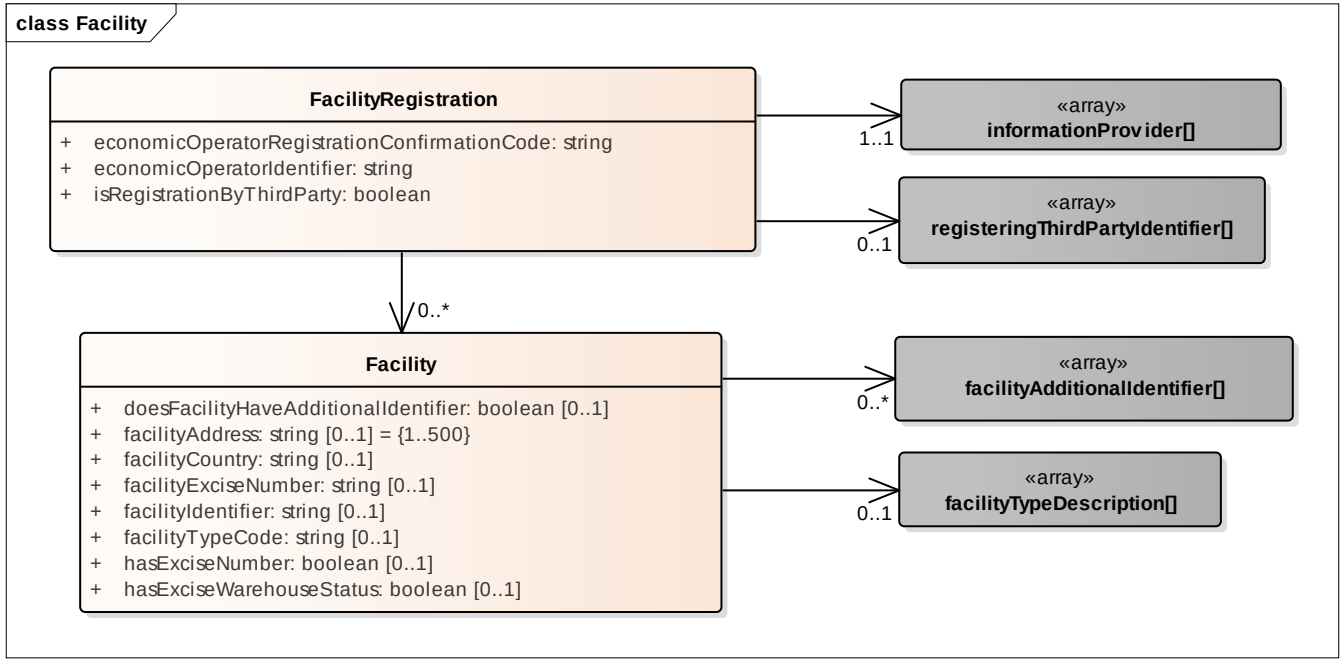
Property Name	Value/Type	Multiplicity	Description
doesEconomicOperatorHaveAdditionalIdentifier	boolean	0..1	Indication if the economic operator has been allocated an identifier by another ID issuer.
economicOperatorAddress	string	0..1	The economic operator's address including street, name, house number, postal code, city.

Property Name	Value/Type	Multiplicity	Description
economicOperatorAdditionalIdentifier[]	array	0..*	An identifier issued to the economic operator by another identifier issuer besides the required identifiers.
economicOperatorAlternativeName	string	0..1	An additional (e.g. abridged) name used by the economic operator.
economicOperatorCountryOfRegistration	string	0..1	The country that the economic operator is being registered in expressed as ISO 3166 Alpha-2.
economicOperatorEmail	string	0..1	The economic operator's email address used to inform about registration process, including subsequent changes and other required correspondence.
economicOperatorExciseNumber	string	0..1	Economic operators excise number issued by the competent authority for the purpose of identification of persons/premises. The excise number is composed as ISO 3166-1 code and eleven alphanumeric characters.
economicOperatorIdentifier	string	0..1	Economic operator identifier code corresponding to the format established by the ID issuer coded with the invariant set of ISO646:1991. Coded as a UIC + GLN.
economicOperatorRegisteredName	string	0..1	The full name used by the economic operator for the registration as an economic operator.
economicOperatorTaxRegistrationNumber	string	0..1	Economic operators tax registration number to identify a party for duty, fee or tax, issued by the competent authority.
economicOperatorVATNumber	string	0..1	Economic operators VAT registration number to identify a party for value added tax, issued by the competent authority.
hasExciseNumber	boolean	0..1	Determines whether the party has an excise number issued by a competent authority for the purpose of identification of persons/premises.
isVATRegistered	boolean	0..1	Determines whether the party is registered for Value Added Tax at the competent authority
informationProvider[]	array	1..1	The Global Location Number of the originator of the data. This could either be a data source or a data recipient.
economicOperatorRegistrationConfirmationCode	string	0..1	The economic operator's confirmation code provided in response to the registration of the economic operator.

Property Name	Value/Type	Multiplicity	Description
isRegistrationByThirdParty	boolean	1..1	Indicates whether the registration has been made on behalf of a retail outlet operator not otherwise involved in the tobacco trade.
registeringThirdPartyIdentifier[]	array	0..1	Identifier of the economic operator that acts on behalf of a retail outlet operator not involved otherwise in the tobacco trade.

3.3 Facility Registration

The following diagram and table specify the properties to manage a facility for an economic operator in a registry. This object can be used to register, correct or deregister a facility. Rules regarding what can be populated for each action (e.g. register, correct) can be found in Section 4.2.

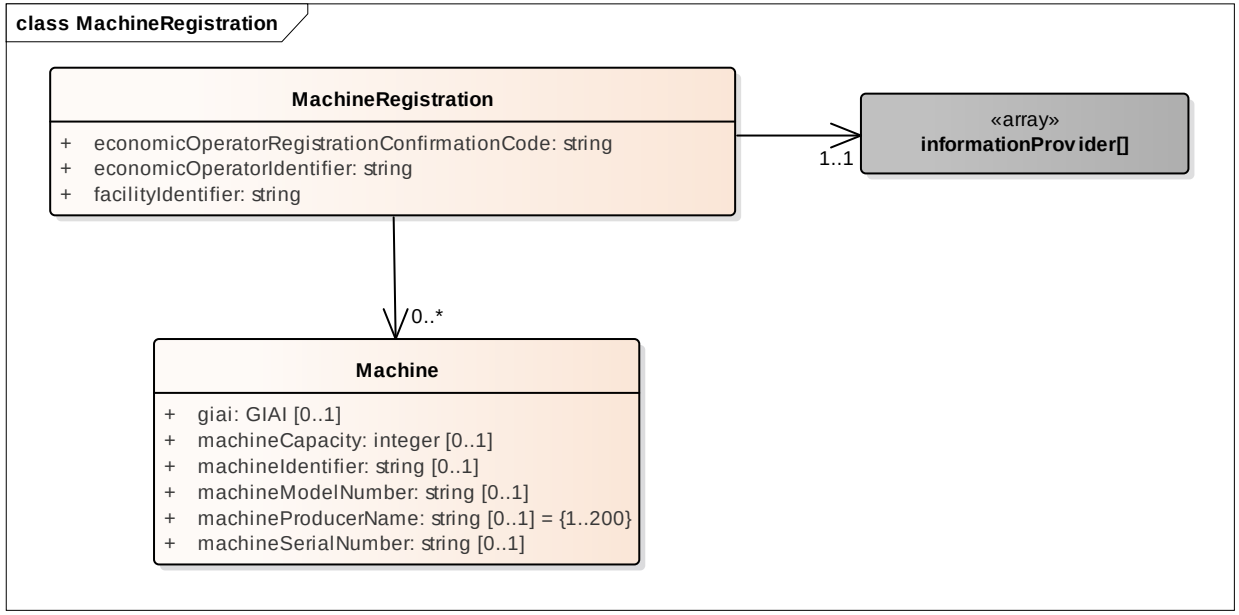


Property Name	Value/Type	Multiplicity	Description
doesFacilityHaveAdditionalIdentifier	boolean	0..1	Indication if the facility has been allocated an identifier by another ID issuer.
facilityAddress	string	0..1	Facility's address – street name, postal code and city
facilityAdditionalIdentifier[]	string	0..*	An identifier issued to the facility by another identifier issuer besides the required identifiers. The identifier must be coded with the invariant character set of ISO 646.

Property Name	Value/Type	Multiplicity	Description
facilityCountry	string	0..1	The country that the facility is located in.
facilityExciseNumber	string	0..1	An excise number issued by the competent authority to a facility for the purpose of identification of persons/premises.
facilityIdentifier	string	0..1	The unique identification of a facility. For GS1, this is a UIC + GLN.
facilityTypeCode	string	0..1	The type of facility expressed as a code for example manufacturing site with warehouse (1), standalone warehouse (2), retail outlet (3), other (4).
facilityTypeDescription[]	array	0..1	A description of the facility type if it has a facility type of other (4).
hasExciseWarehouseStatus	boolean	0..1	Indication if a part of the facility has a tax excise warehouse status.
economicOperatorIdentifier	string	1..1	Economic operator identifier code corresponding to the format established by the ID issuer coded with the invariant set of ISO646:1991. Coded as a UIC + GLN.
economicOperatorRegistrationConfirmationCode	string	1..1	The economic operator's confirmation code provided in response to the registration of the economic operator.
isRegistrationByThirdParty	boolean	1..1	Indicates whether the registration has been made on behalf of a retail outlet operator not otherwise involved in the tobacco trade.
informationProvider[]	array	1..1	The Global Location Number of the originator of the data. This could either be a data source or a data recipient.
registeringThirdPartyIdentifier[]	array	0..1	Identifier of the economic operator that acts on behalf of a retail outlet operator not involved otherwise in the tobacco trade.

3.4 Machine Registration

The following diagram and table specify the properties to manage a machine for a facility in a registry. This object can be used to register, correct or deregister a machine. Rules regarding what can be populated for each action (e.g. register, correct) can be found in Section 4.2.

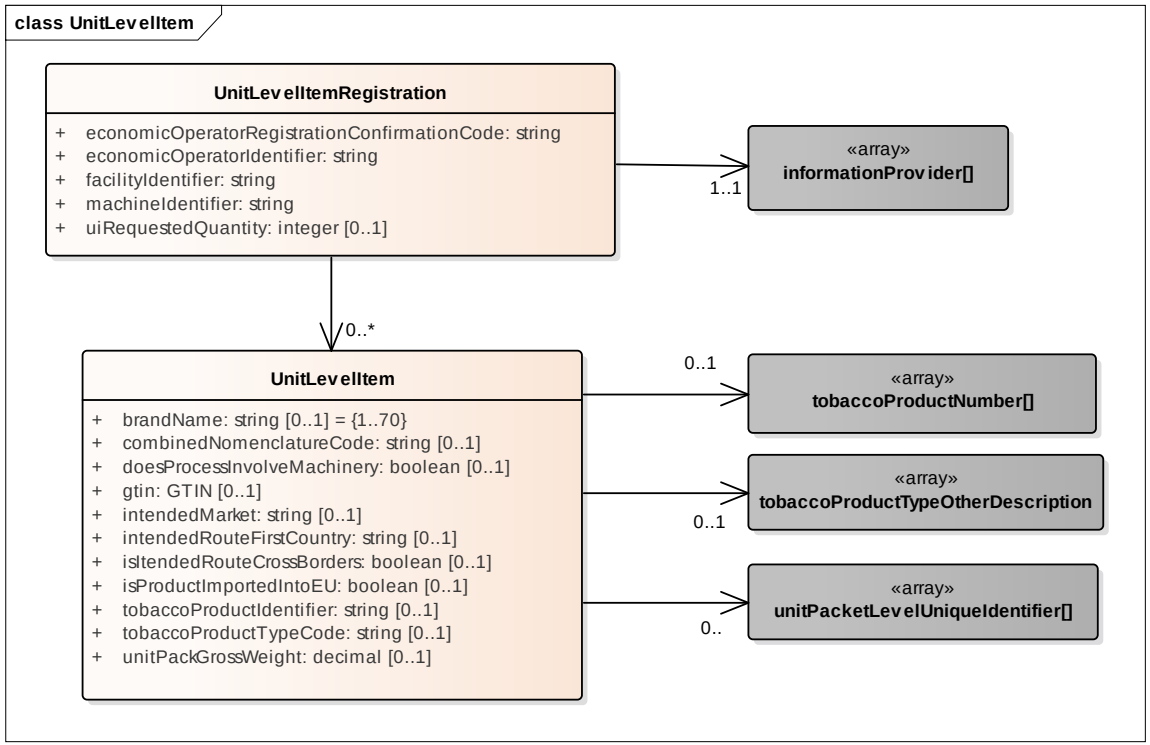


Property Name	Value/Type	Multiplicity	Description
giai	GIAI	0..1	The GIAI is the GS1 identification key used to identify an individual asset. The key is comprised of a GS1 Company Prefix and an individual asset reference.
machineCapacity	Integer	0..1	The maximum capacity over a 24-hour production cycle expressed in the number of packages.
machineIdentifier	string	0..1	Machine identifier code expressed as a GIAI.
machineModelNumber	string	0..1	The machines model number.

Property Name	Value/Type	Multiplicity	Description
machineProducerName	string	0..1	The manufacturer of the machine expressed as text.
machineSerialNumber	string	0..1	The serial number issued for the machine.
economicOperatorIdentifier	string	1..1	Economic operator identifier code corresponding to the format established by the ID issuer coded with the invariant character set of ISO646:1991. Coded as a UIC + GLN.
economicOperatorRegistrationConfirmationCode	string	1..1	The economic operator's confirmation code provided in response to the registration of the economic operator.
facilityIdentifier	string	1..1	The unique identification of a facility. For GS1, this is a UIC + GLN.

3.5 Unit Level Item Registration

The following diagram and table specify the properties to manage unit level identifiers. This object can be used to request/register a unit level identifier. Rules regarding what can be populated for each action can be found in Section 4.2.



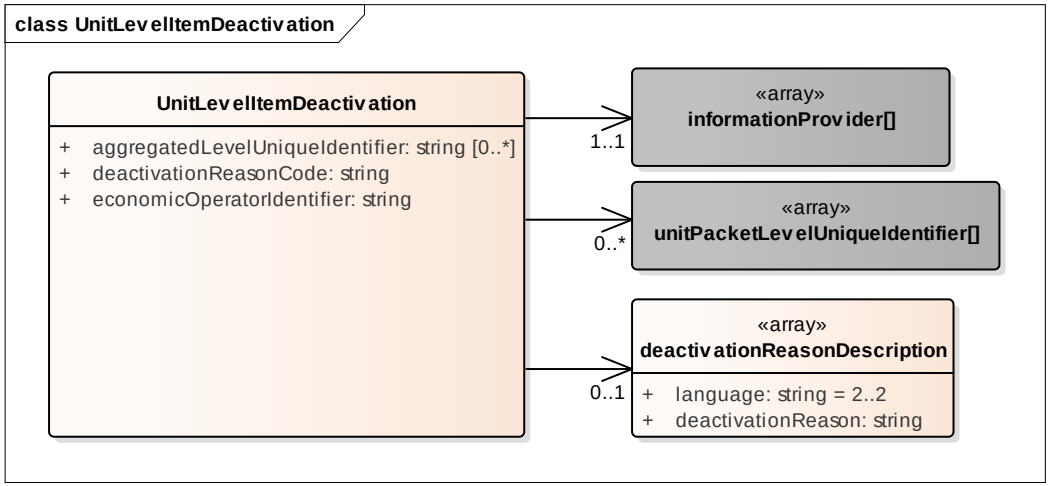
Property Name	Value/Type	Multiplicity	Description
economicOperatorIdentifier	string	1..1	Economic operator identifier code corresponding to the format established by the ID issuer coded with the invariant character set of ISO646:1991. Coded as a UIC + GLN.
economicOperatorRegistrationConfirmationCode	string	1..1	The economic operator's confirmation code provided in response to the registration of the economic operator.
facilityIdentifier	string	1..1	The unique identification of a facility. For GS1, this is a UIC + GLN.

Property Name	Value/Type	Multiplicity	Description
informationProvider[]	array	1..1	The Global Location Number of the originator of the data. This could either be a data source or a data recipient.
machineIdentifier	string	1..1	Machine identifier code expressed as a GIAI.
uiRequestedQuantity	integer	0..1	The requested number of unit packet level UIs.
brandName	string	0..1	The recognisable name used by a brand owner to uniquely identify a line of trade item or services. This is recognisable by the consumer.
combinedNomenclatureCode	string	0..1	An 8-digit product classification system used in export declarations and in statistical declarations for trade in the European Community.
doesProcessInvolveMachinery	boolean	0..1	Indicates if the production process involves machinery.
gtin	GTIN	1..1	Global Trade Item Number (GTIN), the GS1 key used for the identification of trade items. Note: this is mandatory when creating a unit packet level unique identifier.
intendedMarket	string	0..1	Intended country of retail sale represented as an ISO 3166-1 alpha 2 code.
intendedRouteFirstCountry	string	0..1	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. Represented as an ISO 3166-1 alpha 2 code.
isItendedRouteCrossBorders	boolean	0..1	Indication if the product is intended to be moved across country borders with terrestrial, water or air transport.
isProductImportedIntoEU	boolean	0..1	Indicates whether the product is imported into the European Union.
tobaccoProductIdentifier	string	0..1	The tobacco product identifier used in the EU-CEG system. tobaccoProductIdentifier is represented using the format NNNNN-NN-NNNN.
tobaccoProductNumber	string	0..1	The tobacco product number used in the EU-CEG system to identify product presentations. This could be a GTIN or could be an alternative identifier.

Property Name	Value/Type	Multiplicity	Description
tobaccoProductTypeCode	string	0..1	Type of tobacco product: 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 10 - Novel tobacco product 11 - Other (product placed on the market before 19 May 2014, not covered by categories 1-9)
tobaccoProductTypeOtherDescription[]	array	0..1	Description of other type of tobacco product.
unitPackGrossWeight	decimal	0..1	Average gross weight of unit packet including packaging measured in grams with a 0.1 gram accuracy.

3.6 Unit Level Item Deactivation

The following diagram and table specify the properties to deactivate one or many unit level identifiers.

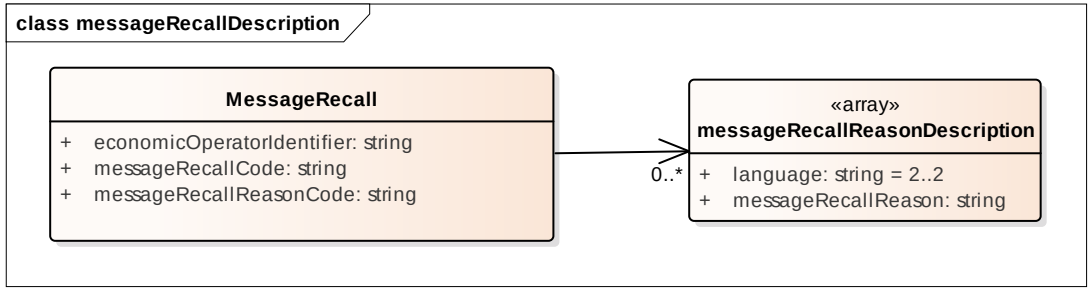


Property Name	Value/Type	Multiplicity	Description
aggregatedLevelUniqueIdentifier	String	0..*	Aggregated level unique identifier.
deactivationReasonCode	string	1..1	Indication of the reason for deactivation of the unit level. 1 – Product destroyed 2 – Product stolen 3 – UI destroyed 4 – UI stolen 5 – UI unused 6 – Other
deactivationReasonDescription[]	array	0..1	Description of the reason a unit level UI has been deactivated if the deactivation reason code equals other (6).

Property Name	Value/Type	Multiplicity	Description
economicOperatorIdentifier	string	1..1	Economic operator identifier code corresponding to the format established by the ID issuer coded with the invariant set of ISO646:1991. Coded as a UIC + GLN.
informationProvider[]	array	1..1	The Global Location Number of the originator of the data. This could either be a data source or a data recipient.
unitLevelIdentifier[]	array	1..*	A list of unit level identifiers to be deactivated.

3.7 Message Recall

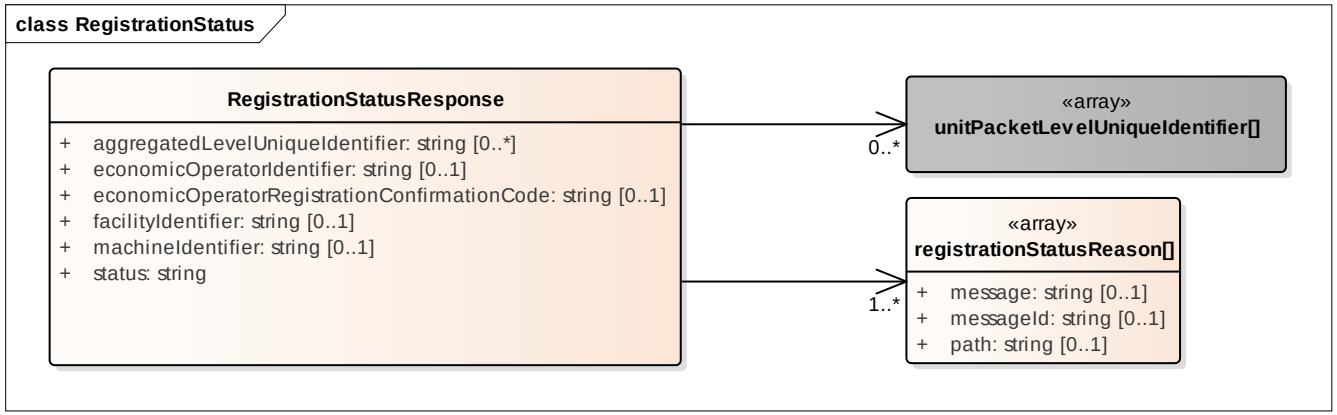
The following diagram and table specify the properties to recall a previously sent message. This object is used for message types 2-1, 2-2, 3-1 to 3.7 and 4.1 to 4.3 as described in Annex II. Other message formats (e.g. XML) may be required for messages 3-1 to 3.7 but the message recall payload should remain the same.



Property Name	Value/Type	Multiplicity	Description
economicOperatorIdentifier	string	1..1	Economic operator identifier code corresponding to the format established by the ID issuer coded with the invariant set of ISO646:1991. Coded as a UIC + GLN.
messageRecallCode	string	1..1	Message recall code provided to the message sender in the acknowledgement of the original message to be recalled.
messageRecallReasonCode	string	1..1	Reason for recalling the original message. 1 – reported event did not materialise (only for message types 3-3 and 3-5) 2 – message contained erroneous information 3 – other
messageRecallReasonDescription[]	array	0..*	Description of the reason for recalling the original message

3.8 Registration Status Response

The following diagram and table specify the properties used to respond to the management of an economic operator, facility, machine or unit level identifier in a registry. This object will provide responses for request/register, correct and deactivate actions.



Property Name	Value/Type	Multiplicity	Description
aggregatedLevelIdentifier	string	0..*	Aggregated level unique identifier.
economicOperatorIdentifier	string	0..1	Economic operator identifier code corresponding to the format established by the ID issuer coded with the invariant set of ISO646:1991. Coded as a UIC + GLN.
economicOperatorRegistrationConfirmationCode	string	0..1	The economic operator's confirmation code provided in response to the registration of the economic operator.
facilityIdentifier	string	0..1	The identifier of a facility within an economic operator.
machineIdentifierCode	string	0..1	Machine identifier code expressed as a GIAI.
unitLevelIdentifier[]	array	0..*	Unit level identifier.

Property Name	Value/Type	Multiplicity	Description
status	string	1..1	Code identifying the result of the operation 1: Product record created 2: Product record modified 3: Product record refreshed 4: Product record deleted 5: Operation failed 6: Not authorised to perform this operation
registrationStatus[]	array	0..*	The status of the registration of an economic operator, facility, machine or unit level.
path	string	0..1	Context path in the case of an error condition.
message	string	0..1	Human readable message providing additional information related to status
messageId	string	0..1	A code number providing additional information related to status.

4 API Reference/Examples

4.1 Method Implementation Details

The following sub-sections provides the implementation details for methods available for each requested FIT Message.

Method	API	HTTP Method	Action	Response payload
1.1. Request for an economic operator identifier code	api/v1/economicOperator/check?gln={gln}	POST		Response Status
1.1.1 Response to Request for an economic operator identifier code	/EconomicOperatorRegistration/??response??			
1.2. Correction of information concerning the economic operator identifier code	/api/v1/economicOperator/{eoid}/update	PUT		Response Status
1.3. De-registration of economic operator identifier code	/api/v1/economicOperator/{eoid}/deregister	PUT	Deregister	Response Status
1.4. Request for a facility identifier code	/api/v1/facility/check?gln=gln	POST		Response Status
1.4.1 Response request for a facility identifier code				
1.5. Correction of information concerning the facility identifier code	/api/v1/facility/{fid}/update	PUT		Response Status
1.6. De-registration of facility identifier code	/api/v1/facility/{fid}/deregister	PUT	Deregister	Response Status
1.7. Request for a machine identifier code	/api/v1/machine/check?giai=<giai>	POST		Response Status
1.7.1 Response to Request for a machine identifier code				
1.8. Correction of information concerning the machine identifier code	/api/v1/machine/{mid}/update	PUT		Response Status
1.9. De-registration of machine identifier code	/api/v1/machine/{mid}/deregister	PUT	Deregister	Response Status
2.1. Request for unit level UIs	/api/v1/product/check?gtin={GTIN}	POST		Response Status
2.1.1 Response to Request for unit level UIs				
2.2. Request for aggregated level UIs	/UnitLevelItemRegistration/	POST		Response Status
2.3. Request for deactivation of UIs	/UnitLevelItemRegistration/{unitLevelIdentifier}/deregister	PUT		Response Status
5. Recalls of requests, operational and transactional messages	/MessageRecall/	PUT		Response Status

4.2 Mandatory Attributes Per Message Type

Note: Grey colour means not applicable to message.

4.2.1 Economic Operator

Attribute	1.1. Request for an economic operator identifier code	1.1 Response	1.2. Correction of information concerning the economic operator identifier code	1.3. De-registration of economic operator identifier code
doesEconomicOperatorHaveAdditionalIdentifier	M		M	
economicOperatorAddress	M		M	
economicOperatorAdditionalIdentifier	C if doesEconomicOperatorHaveAdditionalIdentifier = True		C if doesEconomicOperatorHaveAdditionalIdentifier = True	
economicOperatorAlternativeName	O		O	
economicOperatorCountryOfRegistration	M		M	
economicOperatorExciseNumber	C if hasExciseNumber = True.		C if hasExciseNumber = True.	
economicOperatorEmail	M		M	
economicOperatorIdentifier		EOID	M	M
economicOperatorRegisteredName	M		M	
economicOperatorTaxRegistrationNumber	C if hasExciseNumber = False		C if hasExciseNumber = False	
economicOperatorVATNumber	M if hasExciseNumber = True.		M if hasExciseNumber = True.	
hasExciseNumber	M		M	
isVATRegistered	M		M	
informationProvider	M		M	
economicOperatorRegistrationConfirmationCode		EO CODE	M	M
isRegistrationByThirdParty	M		M	M
registeringThirdPartyIdentifier	M if isRegistrationByThirdParty = True		M if isRegistrationByThirdParty = True	M

4.2.2 Facility

Attribute	2.1. Request for Facility Identifier Code	2.1 Response	2.2. Correction of information concerning the Facility Identifier Code	2.3. De-registration of Facility identifier code
doesFacilityHaveAdditionalIdentifier	M		M	
facilityAddress	M		M	
facilityAdditionalIdentifier	O		O	
facilityCountry	M		M	
facilityExciseNumber	M		M	
facilityIdentifier		FID	M	
facilityType	M		M	
facilityTypeDescription	O		O	
hasExciseWarehouseStatus	M		M	
economicOperatorIdentifier	M		M	M
economicOperatorRegistrationConfirmationCode	M		M	
isRegistrationByThirdParty	M		M	
informationProvider	M		M	
registeringThirdPartyIdentifier	O		O	

4.2.3 Machine

Attribute	3.1. Request for a machine identifier code	3.1 Response	3,2 Correction of information concerning the machine identifier code	3.3. De-registration of machine identifier code
giai	O		O	
machineCapacity	M		M	
machineIdentifier[]		MID	M	M
machineModelNumber	M		M	
machineProducerName	M		M	
machineSerialNumber	M		M	
economicOperatorIdentifier	M		M	
economicOperatorRegistrationConfirmationCode	M		M	M
facilityIdentifier	M		M	
informationProvider	M		M	M

4.2.4 Unit Level Item Registration

Attribute	4.1. Request for unit level identifier code
economicOperatorIdentifier	M
economicOperatorRegistrationConfirmationCode	M
facilityIdentifier	M
informationProvider	M
machineIdentifier	M
uiRequestedQuantity	M
brandName	M
combinedNomenclatureCode	O
doesProcessInvolveMachinery	M
gtin	M
intendedMarket	M
intendedRouteFirstCountry	C if isItendedRouteCrossBorders = True.
isItendedRouteCrossBorders	M
isProductImportedIntoEU	M
tobaccoProductIdentifier	O
tobaccoProductNumber	O
tobaccoProductTypeCode	M
tobaccoProductTypeOtherDescription	O
unitPackGrossWeight	M
unitPacketLevelUniqueIdentifier	

4.3 Sample Messages

4.3.1 Register Economic Operator - Request

```
POST /api/v1/EconomicOperatorRegistration/
TTP/1.1

Host: 123.gs1.org
Content-Type: application/json
Accept: application/json
Authorization: Basic aHR0cHdhdGNoXcfft54jYtGG7h02TiQzOmY=
[
 {
      "economicOperatorRegistration": {
             "isRegistrationByThirdParty": "False",
                           "gln": "1234567890128"
             ,
             "economicOperator": {
                    "economicOperatorAddress": "123 Hudson Street",
                    "economicOperatorEmail": "abc@efg.org",
                    "economicOperatorExciseNumber": "00000987ABC",
                    "economicOperatorRegisteredName": "AAT Company",
                    "economicOperatorVATNumber": "1234567890"
             }
      }
}
]
```

4.3.2 Register Economic Operator - Response

```
POST /api/v1/EconomicOperatorRegistration/
HTTP/1.1

Host: 123.gs1.org
Content-Type: application/json
Accept: application/json
Authorization: Basic aHR0cHdhdGNoXcfft54jYtGG7h02TiQzOmY=
[
 {
      "economicOperatorRegistration": {
             "isRegistrationByThirdParty": "False",
             “EOID": "5vB_1234567890128",
             "economicOperator": {
                    "economicOperatorAddress": "123 Hudson Street",
                    "economicOperatorEmail": "abc@efg.org",
                    "economicOperatorExciseNumber": "00000987ABC",
                    "economicOperatorRegisteredName": "AAT Company",
                    "economicOperatorVATNumber": "1234567890"
             }
      }
}
]
```

4.3.3 Register Facility

```
POST /api/v1/FacilityRegistration/
HTTP/1.1

Host: 123.gs1.org
Content-Type: application/json
Accept: application/json
Authorization: Basic aHR0cHdhdGNoXcfft54jYtGG7h02TiQzOmY=

[
 {
      "economicOperatorRegistration": {
             "isRegistrationByThirdParty": "False",

             "gln": "123456780128",

             "economicOperator": {
                    "doesFacilityHaveAdditionalIdentifier":"False",
                    "facilityAddress": "Kerkstraat 90HS, 1000 GP Amsterdam, Netherlands",
                    "facilityCountry":"NL",
                    "facilityExciseNumber": "00000989ABC",
                    "facilityType": "2",
                    "hasExciseWarehouseStatus": "True",
                    "economicOperatorIdentifier":"5vB_1234567890128",
                    "economicOperatorRegistrationConfirmationCode":"1234"
             }
      }
}
]
```

4.3.4 Register Facility - Response

```
POST /api/v1/FacilityRegistration/
HTTP/1.1

Host: 123.gs1.org
Content-Type: application/json
Accept: application/json
Authorization: Basic aHR0cHdhdGNoXcfft54jYtGG7h02TiQzOmY=

[
 {
      "economicOperatorRegistration": {
             "isRegistrationByThirdParty": "False",
             "FID": "5vBH1234567890128",
             "economicOperator": {
                    "doesFacilityHaveAdditionalIdentifier":"False",
                    "facilityAddress": "Kerkstraat 90HS, 1000 GP Amsterdam, Netherlands",
                    "facilityCountry":"NL",
                    "facilityExciseNumber": "00000989ABC",
                    "facilityType": "2",
                    "hasExciseWarehouseStatus": "True",
                    "economicOperatorIdentifier":"5vBH1234567890128",
                    "economicOperatorRegistrationConfirmationCode":"1234"
             }
      }
}
```

4.3.5 Machine ID Registration

```
POST /api/v1/MachineRegistration/
HTTP/1.1

Host: 123.gs1.org
Content-Type: application/json
Accept: application/json
Authorization: Basic aHR0cHdhdGNoXcfft54jYtGG7h02TiQzOmY=

[
 {
      "machineRegistration": {
             "economicOperatorIdentifier":"5vB_1234567890128",
             "economicOperatorRegistrationConfirmationCode":"1234",
             "facilityIdentifier":"5vBH1234567890128",
             “giai”:”0123456df”,
             "machine": {
                    "machineCapacity":"200",
                    "machineModelNumber": "62524",
                    "machineProducerName": "Stripmaker",
                    "machineSerialNumber":"704012NL246172363681123"
               }
      }
}
]
```

4.3.6 Machine ID Registration - Response

```
POST /api/v1/MachineRegistration/
HTTP/1.1

Host: 123.gs1.org
Content-Type: application/json
Accept: application/json
Authorization: Basic aHR0cHdhdGNoXcfft54jYtGG7h02TiQzOmY=

[
 {
      "machineRegistration": {
             "economicOperatorIdentifier":"5vB_1234567890128",
             "economicOperatorRegistrationConfirmationCode":"1234",
             "facilityIdentifier":"5vBH1234567890128",
             “MID”:”5vBH0123456df”,
             "machine": {
```

```
                    "machineCapacity":"200",
                    "machineModelNumber": "62524",
                    "machineProducerName": "Stripmaker",
                    "machineSerialNumber":"704012NL246172363681123"
                }
        }
}
]
```

4.3.7 Unit Level Registration

```
POST /api/v1/UnitLevelRegistration/add HTTP/1.1

Host: 123.gs1.org
Content-Type: application/json
Accept: application/json
Authorization: Basic aHR0cHdhdGNoXcfft54jYtGG7h02TiQzOmY=

[
{
      "UnitLevelItemRegistration": {
              "economicOperatorIdentifier": "5vB_1234567890128",",
              "economicOperatorRegistrationConfirmationCode": "1234",
              "facilityIdentifier": "5vBH1234567890128",
              "machineIdentifier":"5vBH0123456df ",
              "uiRequestedQuantity": "1",
              "unitLevelItem": {
                      "brandName": "MM Brand",
                      "doesProcessInvolveMachinery": "True",
                      "gtin":"24635476567650",
                      "intendedMarket": "NL",
                      "intendedRouteFirstCountry":"NL",
                      "isIntendedRouteCrossBorders": "False",
                      "isProductImportedIntoEU": "False",
                      "tobaccoProductIdentifier":"02565-16-00230",
                      "tobaccoProductNumber":"00012345600012",
                      "tobaccoProductTypeCode": "1",
                      "unitPackGrossWeight": "1.1"
              }
      }
}
]
```

4.3.8 Unit Level Deactivation

```
POST /api/v1/UnitLevelRegistration/add HTTP/1.1

Host: 123.gs1.org
Content-Type: application/json
Accept: application/json
Authorization: Basic aHR0cHdhdGNoXcfft54jYtGG7h02TiQzOmY=

[
      {
              "UnitLevelItemRegistration": {
                      "economicOperatorIdentifier": "52J1723636811231",
                      "deactivationReasonCode": "1",
                      "unitPacketLevelUniqueIdentifier": "51 qY6IgY) < & Jp3 * j70123456789456",
                      "informationProvider": {"gln": "2463547656765"
                       }
              }
      }
]
```

4.3.9 Response

```
HTTP/1.1 200 OK
Content-Type: application/json

[
      {
        " economicOperatorIdentifier ":"52J1723636811231",
        "economicOperatorRegistrationConfirmationCode":"1234",
        "status":”1”,
        "message":"economicOperatorIdentifier successfully created.”
      }
]
```

4.3.10 Exception

```
HTTP/1.1 200 OK
Content-Type: application/json
[
{"exception":7,"message":"Not available at this time"}
]
```

5 Codes

5.1 Additional Party Identification

Codes used for this code list can be found at the following location: *Additional Party Identification*

5.2 Exception messages

The table below provides a list of standard messages that may be returned in `Exception.messageId`

Message ID	Description
E001	Integrity failed: The length of this GTIN is invalid.
E002	Integrity failed: Incorrect check digit.
E003	Integrity failed: String contains alphanumerical characters.
E004	Incorrect number. That GS1 prefix (3-digit country code) does not exist.
E005	Incorrect number based on GS1 Prefix reserved for special use.
E006	Incorrect number. That GS1 company prefix has not been assigned.
E007	Invalid economicOperatorIdentifier
E008	Invalid facilityIdentifier

Message ID	Description
E009	Invalid machineIdentifier
E010	facilityAdditionalIdentifier not populated.
E011	facilityExciseNumber not populated.
E012	economicOperatorRegistrationConfirmationCode not populated.
E013	informationProvider GLN is not populated.
E014	economicOperatorExciseNumber not populated.
E015	economicOperatorIdentifier not populated
E016	uiRequestedQuantity is not populated.
E017	brandName is not populated.
E018	doesProcessInvolveMachinery not populated.
E019	intendedMarket not populated.
E020	isProductImportedIntoEU is not populated.
E021	tobaccoProductTypeCode is not populated.
E022	unitPackGrossWeight is not populated.

5.3 Message Recall Reason Code

Code	Description
1	Reported event did not materialise (only for business transaction messages types 3-3 and 3-5)
2	Message contained erroneous information
3	Other

5.4 Status Code

Code	Description
1	Record created
2	Record modified
3	Record refreshed
4	Record deleted
5	Operation failed
6	Not authorised to perform this operation

5.5 Status Message

The table below provides a list of standard messages that may be returned in RegistrationStatusReason[].messageId. These messages provide more detail to Status Code.

Message ID	Description
S006	economicOperatorIdentifier not created.
S007	economicOperatorIdentifier, successfully created.
S008	facilityIdentifier not created.
S009	facilityIdentifier successfully created
S010	machineIdentifier not created.

Message ID	Description
S011	machineIdentifier successfully created
S012	unitLevelIdentifier not created.
S013	unitLevelIdentifier successfully created.

5.6 Tobacco Product Type Code

Message ID	Description
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
10	Novel tobacco product
11	Other (product placed on the market before 19 May 2014, not covered by categories 1-9)

5.7 Unit Packet Level Unique Identifier Code

Message ID	Description
UPI_L	Unit packet level unique identifier(L)
UPI_S	Unit packet level unique identifier(S)

6 Implementation considerations

Not Applicable

Appendix A: Contributors

First Name	Last Name	Company
Karen	Arkesteyn	GS1 Belgium & Luxembourg
Henri	Barthel	GS1 Global Office
Robert	Beideman	GS1 Global Office
Mike	Bennett	British American Tobacco (AIT) Limited
Elizabeth	Board	GS1 Global Office
Sofia	Bokvist	Swedish Match
Massimo	Bolchini	GS1 Italy
Ulrich	Bressem	REWE Group
Guido	Bündgen	Lekkerland eutschland GmbH & Co.KG
Jonas	Buskenfried	GS1 Sweden
Xiaoyun	Cao	GS1 China
Emanuela	Casalini	GS1 Italy
Karolin	Catela	GS1 Sweden
Madalina	Cernat	GS1 Romania
Anthony	Chan	GS1 Hong Kong
Patrick	Chanez	INEXTO SA
Shawn	Chen	GS1 Thailand
Claudia	Chmella	Reemtsma Cigarettenfabriken GmbH
Juan Antonio	Cornago Egido	Reemtsma Cigarettenfabriken GmbH
Luiz	Costa	GS1 Brasil
James	Cutforth	Domino Printing Sciences PLC
Stefanie	De Rocker	GS1 Belgium & Luxembourg
Raymond	Delnicki	GS1 US
Sean	Dennison	GS1 Ireland
Ferran	Domenech Fuste	GS1 Spain
Karina	Duvinger	GS1 Sweden
Oliver	Erlenkämper	Movilizer GmbH
Filipe	Esteves	GS1 Portugal
Dawn	Fiorentino Izzi	DoD Logistics AIT Standards Office
Richard	Fisher	DoD Logistics AIT Standards Office
Alan	Gormley	GS1 Ireland
Cátia	Gouveia	GS1 Portugal
Scott	Gray	GS1 Global Office
Rami	Habbal	GS1 UAE
Michaela	Hähn	GS1 Germany
Sandra	Hohenecker	GS1 Germany
Kurt	Hoppen	Bluhm Systeme GmbH
Martin	Hörberg	ICA Sverige AB
Marc	Inderbitzin	Migros-Genossenschafts-Bund

First Name	Last Name	Company
David	Inderfurth	REWE Group
Yoshihiko	Iwasaki	GS1 Japan
Coen	Janssen	GS1 Global Office
Yo Han	Jeon	GS1 Korea
Eric	Jones	British American Tobacco (AIT) Limited
Kenneth	Jørgensen	GS1 Denmark
Iliada	Karali	GS1 Association Greece
Eric	Kauz	GS1 Global Office
Kimmo	Keravuori	GS1 Finland
Sabine	Klaeser	GS1 Germany
Zoltan	Krazli	GS1 Hungary
Alexey	Krotkov	GS1 Russia
Jens	Kungl	METRO Group
Stephen	Lam	GS1 Hong Kong
Endre	Lazar	Movilizer GmbH
Brian	Lee	Japan Tobacco International SA
Zhimin	Li	GS1 China
Sean	Lockhead	Lockhead Consulting Group LLC
Marisa	Lu	GS1 Chinese Taipei
Yan	Luo	GS1 China
André	Machado	TrustaTAG
Ilka	Machemer	GS1 Germany
Valerie	Marchand	GS1 France
Bob	Martin	Booker Ltd
Roberto	Matsubayashi	GS1 Brasil
Riad	Mechtari	GS1 Algeria
Thomas	Meier	Transgourmet Schweiz AG
Lee	Metters	Domino Printing Sciences PLC
Mario	Mira	Movilizer GmbH
Maneesh	Naganand	GS1 Global Office
Mori	Naoko	GS1 Japan
Madeleine	Olsson	Axfood Sverige AB
Manos	Papadakis	GS1 Association Greece
James	Perng	GS1 Chinese Taipei
Neil	Piper	GS1 UK
Francesca	Poggiali	GS1 Global Office
Paul	Reid	GS1 UK
Marianna	Revallova	GS1 Slovakia
Rich	Richardson	GS1 US
Alexandre	Rieucau	GS1 France
Patrice	RIGOLLE	GS1 France

First Name	Last Name	Company
Steven	Robba	1WorldSync Holdings, Inc.
Jose	Rodrigo Gutierrez	Compañía de Distribución Integral Logista Holdings, S.A.
Greg	Rowe	GS1 Global Office
Sylvia	Rubio Alegren	ICA Sverige AB
Zbigniew	Rusinek	GS1 Poland
John	Ryu	GS1 Global Office
Ulrich	Schäfer	GS1 Global Office
Sue	Schmid	GS1 Australia
DAVID	SCHULLER DE SANTOS	Compañía de Distribución Integral Logista Holdings, S.A.
Theodore	Schultze	Symbology, Inc.
Eugen	Sehorz	GS1 Austria
Tamas	Sipos	Philip Morris International Management SA
Olga	Soboleva	GS1 Russia
Luigi	Stanchieri	Philip Morris International Management SA
Richard	Stanley	Brandbank
Earle	Stokes	Cognex Corp.
Roman	Strand	GS1 Germany
Torbjörn	Sundquist	SMD Logistics AB
Diane	Taillard	GS1 Global Office
Hiromitsu	Takai	GS1 Japan
Gary	Telfer	British American Tobacco (AIT) Limited
David	Thayer	Domino Printing Sciences PLC
Gysin	Tobias	Oettinger Davidoff AG
Jon	Verb	Bar Code Graphics, Inc.
Amber	Walls	GS1 US
Yi	Wang	GS1 China
Wilfried	Weigelt	REA Elektronik GmbH
Carrie	Wilkie	GS1 US
Alex	Winiarski	Winiarski Group
XinMin	WU	GS1 China
Jane	Wulff	GS1 Denmark
Ruoyun	Yan	GS1 China
Yang	Zilong	GS1 China
Sofia	Zvereva	British American Tobacco (AIT) Limited