PD CEN/TS 16405:2017

BSI Standards Publication

Intelligent transport systems — Ecall — Additional data concept specification for heavy goods vehicles

bsi.

National foreword

This Published Document is the UK implementation of CEN/TS 16405:2017. It supersedes PD CEN/TR 16405:2013 which is withdrawn.

The UK participation in its preparation was entrusted to Technical Committee EPL/278, Intelligent transport systems.

A list of organizations represented on this committee can be obtained on request to its secretary.

This publication does not purport to include all the necessary provisions of a contract. Users are responsible for its correct application.

Published by BSI Standards Limited 2017

ISBN 978 0 580 86428 5
ICS 03.220.20; 35.240.60

Compliance with a British Standard cannot confer immunity from legal obligations.

This Published Document was published under the authority of the Standards Policy and Strategy Committee on 28 February 2017.

Amendments/corrigenda issued since publication

Date	Text affected

TECHNICAL SPECIFICATION

SPÉCIFICATION TECHNIQUE

TECHNISCHE SPEZIFIKATION

CEN/TS 16405

January 2017

ICS 03.220.20; 35.240.60

Supersedes CEN/TR 16405:2013

English Version

Intelligent transport systems - Ecall - Additional data concept specification for heavy goods vehicles

Systèmes de transports intelligents - Sécurité - Spécification de conception de données additionnelles pour les poids lourds

Intelligente Verkehrssysteme - E-Sicherheit - Zusätzliche Datenkonzept-Spezifikation für Lastkraftwagen

This Technical Specification (CEN/TS) was approved by CEN on 13 October 2014 for provisional application.

The period of validity of this CEN/TS is limited initially to three years. After two years the members of CEN will be requested to submit their comments, particularly on the question whether the CEN/TS can be converted into a European Standard.

CEN members are required to announce the existence of this CEN/TS in the same way as for an EN and to make the CEN/TS available promptly at national level in an appropriate form. It is permissible to keep conflicting national standards in force (in parallel to the CEN/TS) until the final decision about the possible conversion of the CEN/TS into an EN is reached.

CEN members are the national standards bodies of Austria, Belgium, Bulgaria, Croatia, Cyprus, Czech Republic, Denmark, Estonia, Finland, Former Yugoslav Republic of Macedonia, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, Netherlands, Norway, Poland, Portugal, Romania, Serbia, Slovakia, Slovenia, Spain, Sweden, Switzerland, Turkey and United Kingdom.

EUROPEAN COMMITTEE FOR STANDARDIZATION
COMITÉ EUROPÉEN DE NORMALISATION
EUROPÄISCHES KOMITEE FÜR NORMUNG

CEN-CENELEC Management Centre: Avenue Marnix 17, B-1000 Brussels

Ref. No. CEN/TS 16405:2017 E

Contents

Page

Foreword

This document (CEN/TS 16405:2017) has been prepared by Technical Committee CEN/TC 278 “Intelligent transport systems”, the secretariat of which is held by NEN.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. CEN shall not be held responsible for identifying any or all such patent rights.

This document supersedes CEN/TR 16405:2013.

A Technical Report on this subject, proposing these specifications, was approved in 2012 (CEN/TR 16405), for field testing. The proposed specifications have subsequently been tested in the field (by EC Project HeERO and others). The semantic content of this Technical Specification remains unchanged. However the parent Standard EN 15722 (eCall Minimum Set of Data) has been revised and updated, and this Technical Specification is consistent with the layout and specifications of the revised EN 15722.

According to the CEN/CENELEC Internal Regulations, the national standards organisations of the following countries are bound to announce this Technical Specification: Austria, Belgium, Bulgaria, Croatia, Cyprus, Czech Republic, Denmark, Estonia, Finland, Former Yugoslav Republic of Macedonia, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, Netherlands, Norway, Poland, Portugal, Romania, Serbia, Slovakia, Slovenia, Spain, Sweden, Switzerland, Turkey and the United Kingdom.

Introduction

An *eCall* is an emergency call generated either automatically via activation of in-vehicle sensors or manually by the *vehicle occupants*; when activated, to provide notification and relevant location information to the most appropriate 'Public Safety Answering Points' (PSAP), by means of *mobile wireless communications networks* and carries a defined standardized 'Minimum Set of Data' (MSD), notifying that there has been an incident that requires response from the emergency services and establishes an audio channel between the occupants of the vehicle and the most appropriate PSAP.

The MSD (specified in EN 15722) contains static information regarding the vehicle, dynamic information regarding its location, direction of travel etc., at the time of the incident, and makes provision for additional data to be provided.

This Technical Specification provides specification for an optional additional data concept for commercial vehicles to provide dynamic data about the load that it is carrying at the time of the incident that triggered the *eCall*, with specific emphasis on identification of dangerous goods. Two variants are provided, one (schema A) for use where information about the goods (ADR classified or not) is known in the eCall device; the second variant (schema B) is for use where information about the load has to be fetched from other sources.

It is the intention that this Technical Specification is tested in demonstration projects (such as HeERO) with a view to becoming the basis for a future European or International Standard.

In order to claim conformance with this Technical Specification, communication is to be established using accepted wireless communication standards, and it is to be able to demonstrate that the MSD transferred together with any standardized optional data elements defined herein comply with the specifications of this Technical Specification, to the extent that such data are available from the vehicle.

1 Scope

This Technical Specification defines an additional data concept that may be transferred as an 'optional additional data concept' as defined in EN 15722 eCall MSD, that may be transferred from a goods vehicle to a PSAP in the event of a crash or emergency via an *eCall* communication session. Two variants are provided, one (schema A) for use where information about the goods (ADR classified or not) is known in the eCall device; the second variant (schema B) is for use where such information is to be fetched from elsewhere.

NOTE This Technical Specification is complementary and additional to EN 15722; and contains as little redundancy as possible.

The communications media protocols and methods for the transmission of the *eCall* message are not specified in this Technical Specification.

Additional data concepts may also be transferred, and any such data concepts should be registered using a data registry as defined in EN ISO 24978. See www.esafetydata.com for an example.

2 Normative References

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

EN 15722, *Intelligent transport systems - ESafety - ECall minimum set of data*

ISO/IEC 8825-2, *Information technology — ASN.1 encoding rules: Specification of Packed Encoding Rules (PER) — Part 2*

EN ISO 24978, *Intelligent transport systems - ITS Safety and emergency messages using any available wireless media - Data registry procedures (ISO 24978)*

3 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

3.1
112
single European emergency call number supporting Teleservice 12 [ETSI/TS 122 003]

3.2
ASN.1
abstract syntax notation one as specified in the various parts of ITU Recs 8824 and 8825 (ISO 8824 and ISO 8825 various parts)

3.3
commercial vehicle
mechanically propelled road vehicle (vehicle type N1, N2 or N3) that is of a construction primarily suited for the carriage of goods or burden of any kind (not including people) and travelling on a road laden

Note 1 to entry: This includes vehicles designed or adapted to have a maximum weight exceeding 3,500 tonnes, but explicitly excludes busses or other vehicles designed and constructed for the carriage of passengers (ie. vehicle types M1, M2 or M3)

3.4
dangerous goods
categories of goods carried by road defined by the 'European Agreement concerning the 'International Carriage of Dangerous Goods by Road' (ADR) as dangerous; these are characterised as articles or substances which are capable of posing a significant risk to health, safety or to property when transported

3.5
eCall
emergency call generated either automatically via activation of in-vehicle sensors or manually by the vehicle occupants; when activated it provides notification and relevant location information to the most appropriate 'Public Safety Answering Point', by means of mobile wireless communications networks, carries a defined standardized 'Minimum Set of Data' notifying that there has been an incident that requires response from the emergency services, and establishes an audio channel between the occupants of the vehicle and the most appropriate 'Public Safety Answering Point'

3.6
Kemler code
ADR Hazard Identification Number (HIN), carried on placards on tank cars and tank containers running by road under international ADR regulations

3.7
uniform resource identifier
URI
string of characters used to identify a name or a resource on the Internet

3.8
uniform resource locator
URL
URI that in addition to identifying a resource provides a means of locating the resource by describing its primary access mechanism

EXAMPLE Its network location

4 Symbols and abbreviations

ADR Accord européen relative au transport international des marchandises Dangereuses par Route

ETSI European Telecommunications Standards Institute

M Mandatory

MSD Minimum set of data

O Optional

PER Packed Encoding Rules (ASN.1)

PSAP Public Safety Answering Point

UPER Unaligned Packed Encoding Rules (ASN.1)

5 Requirements

5.1 General

This Technical Specification describes an addendum to the standard defined in EN 15722 for the coding of the MSD message. Any requirement from EN 15722 shall be met for the exchange of information about loads in the additional data block

5.2 Concepts and formats

5.2.1 MSD data concepts

The MSD as defined in EN 15722 is a direct, timely message to the PSAP operator receiving the emergency call.

The MSD has an optional additional data block that will be used to add information elements containing information about the load of the vehicle involved.

The information elements in the additional data block of the MSD have been selected on the basis of their relevance in an emergency rescue situation.

5.2.2 Representation of MSD data concepts

The MSD is represented in 'Abstract Syntax Notation' (ASN.1) using the 'Unaligned Packed Encoding Rules' (UPER) as defined in ISO/IEC 8825-2 using the ASN1 definitions defined in Annex A of EN 15722. The message shall be sent in the sequence defined in that same Annex.

The information about the load of the vehicle sending the MSD shall be represented in ASN.1 UPER as well, following the provision made in above named Annex.

5.2.3 Distribution of MSD data

The MSD shall be transmitted as described in EN 15722.

5.2.4 Commercial vehicles optional additional data concept 'Object Identifier'

The object identifier uniquely identifies the format and meaning of the data which follows in the optional additional data concept.

Both the syntax of the data structure and the semantic meaning of the content is referenced via this identifier so that it can be usefully applied.

The uniqueness of each specific relative identifier is ensured by a specific international standardizations body, and maintained in a data registry operated in accordance with EN ISO 24978. These identifiers are all relative to a specific root. And the root of all *eCall* relative OID's shall be the same.

eCall has been allocated the OID 1.0.14817.106.2.1. Within this, arc '.2' has been defined to contain 'Optional Additional Data concepts'. The OID for this deliverable shall be 1.0.14817.106.2.1.2.1.

This deliverable defines two schemes that each have their own unique OID:

Schema A: 1.0.14817.106.2.1.2.1.1

Schema B: 1.0.14817.106.2.1.2.1.2

The OID for 'Optional Additional Data concepts' (1.0.14817.106.2.1.2) is fixed and shall not be transmitted over the air as part of the optional additional data. The MSD data element 'oid' is defined as RELATIVE-OID and shall contain 1.1 if Schema A is used, or 1.2 if Scheme B is used.

For further detail regarding the use of OIDs in eCall, see EN 15722.

5.2.5 Commercial vehicle optional additional data concept 'data'

The objective of the commercial vehicle data concept is to provide the PSAP with data concerning the load of the affected vehicle transmitting the MSD.

Two variants are provided, one (schema A) for use where information about the goods (ADR classified or not) is known in the eCall device; the second variant (schema B) is for use where load information should be fetched from elsewhere.

Paramount priority is given to the transmission of data relating to dangerous/dangerous goods Provision is also made to transfer data concerning other (non ADR) cargos. While these cargoes may not be classified as dangerous/dangerous, in the event of an accident they may cause increased risk of accident or problems for the emergency services – for example livestock; small materials such as ball bearings, liquids, manure or other materials likely to affect the surface tension of the roadway surface or present obstacles on the roadway.

The data concept will take up slightly less than the amount of bytes available for the optional additional data, using the GSM/UMTS maximum message length limit as defined in EN 16062 (140 bytes). As such there is no risk of the complete MSD to exceed the maximum number of bytes allowed by using this data concept.

5.3 Contents of the 'Minimum Set of Data' (MSD)

The following subclauses provide the definition of the minimum set of data that shall be sent from the vehicle in case of an emergency call.

5.3.1 Basic contents of MSD

Table 1 provides a summary of the semantic contents of the MSD, for a full description please refer to EN 15722.

Table 1 — Contents/format of the MSD data concept

M Mandatory data field

O Optional data field

MSD					
msdVersion		INTEGER (1..255)	-	M	
msd					
msdStructure					
optionalAdditionalData				O	
	oid	RELATIVE-OID			
	data	OCTET STRING			

This document describes the contents of the optionalAdditionalData block.

5.3.2 Contents of the optionalAdditionalData for Schema A

Table 2 provides a summary of the semantic contents of the optionalAdditionalData part of the MSD for Schema A.

The sequence of data presentation shall be as specified in Table 2, represented as described in this clause and distributed as described in this clause.

For clarity the 'type' used in Table 2 is a semantic representation of the type used in the ASN.1 definition. The exact representation is found in Annex A.

The real position of the element in the data-stream is defined by the ASN.1 'unaligned packet encoding rules (uPER), following the definition in Annex A. Elements therefore do not necessarily start or end on a byte boundary.

Table 2 — Contents/format of Commercial vehicle additional data Schema A

M Mandatory data field (ie. mandatory if this encoding scheme is used)

O Optional data field

optionalAdditionalData					
oid		RELATIVE OID		M	Fixed value: 1.1
data					*encoded as OCTET STRING*
	commercialVehicleType	ENUM		M	The supported types are: - unknown - tanker, one compartment - tanker, more compartments - piece cargo
	consignorPhone	NumericalString		M	Consignor contact telephone number or telephone number displayed on goods container as contact number in case of emergency. NOTE: the number should be specified as international number, thus including the country- and area code (without zero)
	alarmInfo			O	*Information about sensors present is encoded. Each sensor is optional and should be left out if not present.* *If a sensor is generating an alarm its value should be set to true, if a sensor is available but not generating an alarm its value is false* *IMPORTANT NOTE: Emergency services need to be aware that the absence of an alarm indicates only that there was no alarm showing as activated at the time of compiling the data.* *Alarms raised post the population of/sending of the MSD will not be transmitted. These codes therefore only indicate status before or at the point of the incident, and cannot be taken as the current status post incident.*
		leakageAlarm	BOOLEAN	O	True if leakage has been detected
		fireAlarm	BOOLEAN	O	True if fire has been detected
		highTempAlarm	BOOLEAN	O	True if high temperature has been detected

	lowTempAlarm	BOOLEAN		O	True if low temperature has been detected
	shockAlarm	BOOLEAN		O	True if shock has been detected
	highPressureAlarm	BOOLEAN		O	True if high pressure has been detected
	lowPressureAlarm	BOOLEAN		O	True if low pressure has been detected
	orientationAlarm	BOOLEAN		O	True if abnormal orientation has been det.
	otherAlarm	BOOLEAN		O	True if any other alarm was raised
goodsADR				O	*Up to 7 goods (most dangerous based on response code, within same response code prioritised to most impact in fire or largest volume) can be fully defined.*
definedGoodsADR[1]				O	*Each defined good has its own container with:*
	cargoUNCode	INTEGER		M	UNCode (max. value: 9999)
	kemlerCode	KemlerCode[1]		M	Kemler Code of cargo, up to 3 digits
	packageGroup	INTEGER		M	Package group (1, 2 or 3)
	quantity	INTEGER		M	The quantity of the cargo. Possible values are: 0: empty but uncleaned, 1 – 98: the quantity as expressed 99: 99 tonnes / 99 m3 or more
	quantityInTonnes	BOOLEAN		M	True: quantity is given in tonnes False: quantity is given in m3 (rounded up)
	quantityIsGross	BOOLEAN		M	True: quantity is gross weight/volume
definedGoodsADR [2]				O	
	cargoUNCode kemlerCode packageGroup quantity quantityInTonnes quantityIsGross	See above			
definedGoodsADR [3] …				O	

[1] The Kemler Code is encoded in a defined type that takes the Kemler Code constraints into account

definedGoodsADR [7]					
numberOfUndefined … …GoodsADR	INTEGER			M	Number of ADR goods in the vehicle not fully defined in this section. Possible values: 0: no other ADR goods in vehicle, 1–9: specified number of other ADR goods in vehicle 10: 10 or more ADR goods in vehicle 15: unknown number of (other) ADR goods in vehicle
goodsNonADR				O	*Up to 6 materials of significant quantity (significant defined at the discretion of consignor) can be defined* *NOTE: definition should be in decreasing order of quantity.*
definedGoodsNonADR[1]				O	*Each defined good has its own container with:*
cargoSPSCode	INTEGER			M	The SPC code (can be obtained from www.unspsc.org)
containerType	ENUM			M	The container type code (according to ISO 6346)
definedGoodsNonADR[2]					
cargoSPSCode	See above				
containerType					
definedGoodsNonADR[3] … definedGoodsNonADR[6]				O	
numberOfUndefined … …GoodsNonADR	INTEGER			M	Number of non ADR goods in the vehicle not fully defined in this section. Possible values: 0: no other non ADR goods in vehicle, 1–9: specified number of other non ADR goods in vehicle 10: 10 or more non ADR goods in vehicle 15: unknown number of (other) non ADR goods in vehicle

5.3.3 Contents of the optionalAdditionalData for Schema B

Table 3 provides a summary of the semantic contents of the optionalAdditionalData part of the MSD for Schema B.

The sequence of data presentation shall be as specified in Table 3, represented as described in this clause and distributed as described in this clause.

For clarity the 'type' used in Table 3 is a semantic representation of the type used in the ASN.1 definition. The exact representation is found in Annex B.

The real position of the element in the data-stream is defined by the ASN.1 'unaligned packet encoding rules (uPER), following the definition in Annex A. Elements therefore do not necessarily start or end on a byte boundary.

Table 3 — Contents/format of Commercial vehicle additional data Schema B

M Mandatory data field (i.e. mandatory if this encoding scheme is used)

O Optional data field

optionalAdditionalData					
	oid	RELATIVE OID		M	Fixed value: 1,2
	data				*encoded as OCTET STRING*
	commercialVehicleType	ENUM		M	The supported types are: - unknown - tanker, one compartment - tanker, more compartments - truck, (*stukvracht*)
	consignorPhone	NumericalString		M	Consignor contact telephone number or telephone number displayd on goods container as contact number in case of emergency. NOTE: the number should be specified as international number, thus including the country- and areacode (without zero)
	alarmInfo			O	*Information about sensors present is encoded. Each sensor is optional and should be left out if not present.* *If a sensor is generating an alarm its value should be set to true, if a sensor is available but not generating an alarm its value is false* *IMPORTANT NOTE: Emergency services need to be aware that the absence of an alarm indicates only that there was no alarm showing as activated at the time of*

					compiling the data. *Alarms raised post the population of/sending of the MSD will not be transmitted. These codes therefore only indicate status before or at the point of the incident, and cannot be taken as the current status post incident.*
	leakageAlarm	BOOLEAN		O	True if leakage has been detected
	fireAlarm	BOOLEAN		O	True if fire has been detected
	highTempAlarm	BOOLEAN		O	True if high temperature has been detected
	lowTempAlarm	BOOLEAN		O	True if low temperature has been detected
	shockAlarm	BOOLEAN		O	True if shock has been detected
	highPressureAlarm	BOOLEAN		O	True if high pressure has been detected
	lowPressureAlarm	BOOLEAN		O	True if low pressure has been detected
	orientationAlarm	BOOLEAN		O	True if abnormal orientation has been det.
	otherAlarm	BOOLEAN		O	True if any other alarm was raised
	numberOfGoodsADR	INTEGER		M	Number of ADR goods in the vehicle not fully defined in this section. Possible values: 0–9: specified number of other ADR goods in vehicle 10: 10 or more ADR goods in vehicle 15: unknown number of (other) ADR goods in vehicle
	numberOfGoods… …NonADR	INTEGER		M	Number of non ADR goods in the vehicle not fully defined in this section. Possible values: 0: no other non ADR goods in vehicle, 1–9: specified number of other non ADR goods in vehicle 10: 10 or more non ADR goods in vehicle 15: unknown number of (other) non ADR goods in vehicle
	cargoInformationEnd point			O	
	cargoInformationURI	PrintableString		M	Information about the cargo is available through the given URI (max length: 80 char).

						NOTE 1: the URI should follow the format ‘ < scheme > :// < host > [: < port >]/path[?query-string], example: https://cargo.info.com/msdinfo?key=124 NOTE 2: the information end point should respond in a standardized way, as referenced by cargoInformationProtocol. That standardization is done to be set elsewhere and is outside the scope of this deliverable.
			cargoInformation … …Protocol	RELATIVE-OID		M	Relative object identifier designating the protocol to use to retrieve information through the above named URI.

Annex A
(normative)

ASN.1 definition of optional datablock

A.1 General

As soon as the OID has revealed the nature of the data as being Schema A or B (using the standard eCall MSD message definition, see EN 15722) the data from the optionalAdditionalData block can be decoded. Either by applying the definition of the datablock to that data (this Annex), or by applying a constituted complete eCall MSD message definition (Annex B).

A.2 Definition of contents of optionalAdditionalData.data Schema A

This section contains the ASN.1 definition of the extra data for Schema A.

A.2.1 ASN.1 definition

```
MSD_ADDITIONAL_CV_A_1

DEFINITIONS
AUTOMATIC TAGS::=
BEGIN

- Definition can be used to decode data in the data part
- of optionalAdditionalData in the MSD message.
-
- It can also be used in a constituted definition of an
- extended MSD definition like so:
-
- AdditionalData::= SEQUENCE {
-   oid  RELATIVE-OID,
-   data OCTET STRING (CONTAINING CVADSchemaA)
- }

CVADSchemaA::= SEQUENCE {
  commercialVehicleType          CVehicleType,
  consignorPhone         NumericString(SIZE(1..17)) OPTIONAL,
  alarmInfo              CVAlarmType       OPTIONAL,
  goodsADR               CVGoodsADRType    OPTIONAL,
  goodsNonADR            CVGoodsNonADRType OPTIONAL
}

CVehicleType::= ENUMERATED {
  unknownn(0),
  tankerSingleCompartment(1),
  tankerMultiCompartment(2),
  truckPieceCargo(3),
  ...
}

CVGoodsADRType::= SEQUENCE {
```

```
  definedGoodsADR              SEQUENCE SIZE(1..7) OF CVADRCoded
OPTIONAL,
  numberOfUndefinedGoodsADR    INTEGER (0..15)
}

CVGoodsNonADRType::= SEQUENCE {
  definedGoodsNonADR           SEQUENCE SIZE(1..6) OF CVSPCCoded
OPTIONAL,
  numberOfUndefinedGoodsNonADR INTEGER (0..15)
}

CVAlarmType::= SEQUENCE {
  leakageAlarm      BOOLEAN OPTIONAL,
  fireAlarm         BOOLEAN OPTIONAL,
  highTempAlarm     BOOLEAN OPTIONAL,
  lowTempAlarm      BOOLEAN OPTIONAL,
  shockAlarm        BOOLEAN OPTIONAL,
  highPressureAlarm BOOLEAN OPTIONAL,
  lowPressureAlarm  BOOLEAN OPTIONAL,
  orientationAlarm  BOOLEAN OPTIONAL,
  otherAlarm        BOOLEAN OPTIONAL,
  ...
}

CVADRCoded::= SEQUENCE {
  cargoUNCode       INTEGER (0..9999),
  kemlerCode   ADRKemlerCode,
  packageGroup      INTEGER (1..3),
  quantity          INTEGER (0..99),
  quantityUnit    HGVQuantityUnit
}

CVQuantityUnit::= ENUMERATED {
  qtyInTonnesNet(0),
  qtyInTonnesGross(1),
  qtyInCubicM(5),
  ...
}

ADRKemlerCode::= SEQUENCE {
  firstDigit   PrintableString (SIZE(1)) (FROM("2".."9"|"X")) DEFAULT
"2",
  secndDigit        PrintableString (SIZE(1)) (FROM("0"|"2".."9"|"X"))
OPTIONAL,
  thirdDigit        PrintableString (SIZE(1)) (FROM("0"|"2".."9"|"X"))
OPTIONAL
}

CVSPCCoded::= SEQUENCE {
  cargoSPSCode      NumericString (SIZE(2..8)),
  containerTypeCode CVisoContainerType OPTIONAL
}

CVisoContainerType::= ENUMERATED {
```

```
  containerTypeG0(0),  containerTypeG1(1),  containerTypeG2(2),
containerTypeG3(3),
  containerTypeV0(4),  containerTypeV2(5),  containerTypeV4(6),
  containerTypeR0(7),  containerTypeR1(8),  containerTypeR2(9),
containerTypeR3(10),
  containerTypeH0(11), containerTypeH1(12), containerTypeH2(13),
containerTypeH5(14),
   containerTypeH6(15),
  containerTypeU0(16), containerTypeU1(17), containerTypeU2(18),
containerTypeU3(19),
   containerTypeU4(20), containerTypeU5(21),
  containerTypeT0(22), containerTypeT1(23), containerTypeT2(24),
containerTypeT3(25),
   containerTypeT4(26), containerTypeT5(27), containerTypeT6(28),
containerTypeT7(29),
    containerTypeT8(30), containerTypeT9(31),
  containerTypeB0(32), containerTypeB1(33), containerTypeB3(34),
containerTypeB4(35),
   containerTypeB5(36), containerTypeB6(37),
  containerTypeP0(38), containerTypeP1(39), containerTypeP2(40),
containerTypeP3(41),
   containerTypeP4(42), containerTypeP5(43),
  containerTypeS0(44), containerTypeS1(45), containerTypeS2(46),
  ...
}

END
```

A.2.2 Syntax check of ASN.1 definition

```
ASN.1 Studio Version 6.0.4

Copyright (C) OSS Nokalva, Inc. 2011-2012.  All rights reserved.

This product is licensed for use by Cheiron IT bv, Leiden, Holland -
Standards Editor License 66410E related to eCall/European HeERO Project

ASN.1 syntax check result: C0043I: 0 error messages, 0 warning messages
and 0 informatory messages issued.

Compilation summary: The project HGV_Revision2013_SchemeA includes 1 PDUs
and 0 ASN.1 values.
```

A.2.3 Example

The example below is shown in ASN.1 value encoding (plain text):

```
value HGVSchemaA::=
{
  commercialVehicleType tankerSingleCompartment,
  consignorPhone "31207110123",
  alarmInfo
  {
    leakageAlarm FALSE,
    fireAlarm TRUE,
```

```
    highTempAlarm TRUE,
    highPressureAlarm TRUE,
    lowPressureAlarm FALSE
  },
  goodsADR
  {
    definedGoodsADR
    {
      {
        cargoUNCode 1203,
        kemlerCode "3",
        packageGroup 2,
        quantityAmount 44,
        quantityUnit qtyInCubicM
      }
    },
    numberOfUndefinedGoodsADR 0
  }
}
```

The same example encoded in UPER (hexadecimal representation, 38 bytes):

```
E2A42318 22123473 1D02598C C58800
```

A.3 Definition of contenst of optionalAdditionalData.data Schema B

A.3.1 General

This section contains the ASN.1 definition of the extra data for Schema B.

A.3.2 ASN.1 definition

```
MSD_ADDITIONAL_CV_B_1

DEFINITIONS
AUTOMATIC TAGS::=
BEGIN

- Definition can be used to decode data in the data part
- of optionalAdditionalData in the MSD message.
-
- It can also be used in a constituted definition of an
- extended MSD definition like so:
-
- AdditionalData::= SEQUENCE {
-   oid  RELATIVE-OID,
-   data OCTET STRING (CONTAINING HGVSchemaA)
- }

CVADSchemaB::= SEQUENCE {
  commercialVehicleType            CVehicleType,
  consignorPhone           NumericString(SIZE(1..17)),
  alarmInfo                CVAlarmType  OPTIONAL,
  numberOfGoodsADR         INTEGER (0..15),
  numberOfGoodsNonADR      INTEGER (0..15),
  cargoInformationEndpoint CVcargoInformationEndpoint OPTIONAL
}
```

```
CVcargoInformationEndpoint::= SEQUENCE {
  cargoInformationURI      PrintableString(SIZE(9..80)),
  cargoInformationProtocol RELATIVE-OID
}

CVehicleType::= ENUMERATED {
  unknownn(0),
  tankerSingleCompartment(1),
  tankerMultiCompartment(2),
  truckPieceCargo(3),
  ...
}

CVAlarmType::= SEQUENCE {
  leakageAlarm      BOOLEAN OPTIONAL,
  fireAlarm         BOOLEAN OPTIONAL,
  highTempAlarm     BOOLEAN OPTIONAL,
  lowTempAlarm      BOOLEAN OPTIONAL,
  shockAlarm        BOOLEAN OPTIONAL,
  highPressureAlarm BOOLEAN OPTIONAL,
  lowPressureAlarm  BOOLEAN OPTIONAL,
  orientationAlarm  BOOLEAN OPTIONAL,
  otherAlarm        BOOLEAN OPTIONAL,
  ...
}

END
```

A.3.3 Syntax check

```
ASN.1 Studio Version 6.0.4

Copyright (C) OSS Nokalva, Inc. 2011-2012.  All rights reserved.

This product is licensed for use by Cheiron IT bv, Leiden, Holland -
Standards Editor License 66410E related to eCall/European HeERO Project

ASN.1 syntax check result: C0043I: 0 error messages, 0 warning messages
and 0 informatory messages issued.

Compilation summary: The project HGV_Revision2013_SchemeB includes 1 PDUs
and 0 ASN.1 values.
```

A.3.4 Example

The example below is shown in ASN.1 value encoding (plain text):

```
value HGVSchemaB::=
{
  commerciaVehicleType truckPieceCargo,
  consignorPhone "31207110123",
  alarmInfo
  {
    otherAlarm TRUE
  },
  numberOfGoodsADR 3,
```

```
  numberOfGoodsNonADR 10,
  cargoInformationEndpoint
  {
    cargoInformationURI "https://cargo.infopoint.xx/msd?tag= ...",
    cargoInformationProtocol { 3 8 }
  }
}
```

The same example encoded in UPER (hexadecimal representation, 38 bytes):

```
DA908C60 8848D001 9D1FD1D3 A70E6E97 AFC78796 7DEBB4EE
CDBF86FD 3BBA2EF1 E17EDE79 1FF4C39D EB172E5C 31020308
```

Annex B
(informative)

ASN.1 definition of complete MSD message with HGV info

B.1 General

ASN.1 has the possibility to include coding rules. Therefor as soon as decoding with the definition from EN 15722 has revealed the OID belonging to Schema A or B, the complete message can be decoded using a constituted decoding scheme. This Annex shows the constituted rules for Schema A. It is informative because EN 15722 is normative for the basic part of the MSD message.

B.2 ASN.1 definition of complete extended MSD message, HGV Schema A

```
MSDASN1Module

DEFINITIONS
AUTOMATIC TAGS::=
BEGIN

- Version of this ASN.1 MSD specification
- (there is no real need for this element)
-
CurrentId::= INTEGER (1)

- ECallMessage is the top level information element
- The ECallMessage structure supports only one message type (msd)
- Extendibility at this level is not allowed, thus ensuring that the
- ID (message format version) can be extracted directly.
- Elements:
-   msdVersion:   MSD format version
-                 The format described in this document carries version 1
-   msd:          Minimum Set Of Data uplink from vehicle
-
ECallMessage::= SEQUENCE {
   msdVersion  INTEGER(0 .. 255),
   msd OCTET STRING (CONTAINING MSDMessage)
}

- The main uplink msd message from the vehicle (excluding ID)
- Elements:
-   msdStructure: The main MSD structure
-   optionalAdditionalData: Additional data
- Extendable in future versions at this level e.g. to add extra data
-
MSDMessage::= SEQUENCE {
   msdStructure           MSDStructure,
   optionalAdditionalData AdditionalData OPTIONAL,
  ...
}
```

```
- The main MSD structure, excluding additional data
- Elements:
-   messageIdentifier: Message identifier, starting with 1 for each
-                      new eCall session and to be incremented with
-                      every application layer MSD retransmission
-                      following a new 'Send MSD' request after the
-                      incident event
-   control: see ControlType
-   vehicleIdentificationNumber: see VIN
-   vehiclePropulsionStorageType: see VehiclePropulsionStorageType
-   timestamp: Timestamp of incident event
-              As seconds elapsed since midnight January 1st, 1970 UTC.
-              Failure value for time stamp set to "0"
-   vehicleLocation: see VehicleLocation
-   vehicleDirection: Direction of travel
-                     in 2°-degrees steps from magnetic north
-                     (0- 358, clockwise)
-                     If direction of travel is invalid or unknown,
-                     the value 0xFF shall be used
-                     Only values from 0 to 179 are valid.
-   recentVehicleLocationN1: location delta with respect to
-                            vehicleLocation
-                            see VehicleLocationDelta
-   recentVehicleLocationN2: location deltat with respect to
-                            recentVehicleLocationN1
-                            see VehicleLocationDelta
-   numberOfPassengers: Number of occupants in the vehicle according
-                       to available information.
-                       NOTE 1 This information is is indicative
-                       only as it may be not always be reliable
-                       in providing exact information about the
-                       number of passengers (e.g. because seatbelts
-                       may not be fastened by passengers or seatbelts
-                       may be fastened for other reasons).
-                       NOTE 2 If no information about the number of
-                       occupants is available, this parameter needs
-                       to be omitted or filled with the representation
-                       of value 255
-
MSDStructure ::= SEQUENCE {
   messageIdentifier             INTEGER(0 .. 255),
   control                       ControlType,
   vehicleIdentificationNumber   VIN,
   vehiclePropulsionStorageType  VehiclePropulsionStorageType,
   timestamp                     INTEGER(0 .. 4294967295),
   vehicleLocation               VehicleLocation,
   vehicleDirection              INTEGER(0 .. 255),
   recentVehicleLocationN1       VehicleLocationDelta OPTIONAL,
   recentVehicleLocationN2       VehicleLocationDelta OPTIONAL,
   numberOfPassengers            INTEGER(0 .. 255) OPTIONAL,
  ...
}

- The ControlType is a collection of the following elements:
- Elements:
```

```
-   automaticActivation:  true = Automatic activation,
-                         false = Manual activation
-   testCall:             true = Test call, false = Emergency
-   positionCanBeTrusted: true = Position can be trusted,
-                         false = low confidence in position
-                         NOTE "Low confidence in position"
-                         shall mean that there is less than 95%
-                         confidence that exact position is
-                         within the limits of a radius of ±150m
-                         of reported position
-   vehicleType:          see VehicleType
-
ControlType::= SEQUENCE {
  automaticActivation  BOOLEAN,
  testCall             BOOLEAN,
  positionCanBeTrusted BOOLEAN,
  vehicleType          VehicleType
}


- Definiton of the vehicle type reporting the incident.
- NOTE: Vehicle definitions class M, N according directive 2007/46/EC;
-       class L according directive 2002/24/EC
- Extendable in future versions for new vehicle types
-
VehicleType::= ENUMERATED{
   passengerVehicleClassM1 (1),
   busesAndCoachesClassM2 (2),
   busesAndCoachesClassM3 (3),
   lightCommercialVehiclesClassN1 (4),
   heavyDutyVehiclesClassN2 (5),
   heavyDutyVehiclesClassN3 (6),
   motorcyclesClassL1e (7),
   motorcyclesClassL2e (8),
   motorcyclesClassL3e (9),
   motorcyclesClassL4e (10),
   motorcyclesClassL5e (11),
   motorcyclesClassL6e (12),
   motorcyclesClassL7e (13),
  ...
}

- VIN (vehicle identification number) according ISO 3779
-   isowmi: World Manufacturer Index (WMI)
-   isovds: Vehicle Type Descriptor (VDS)
-   Vehicle Identifier Section (VIS) consisting of
-     isovisModelyear: Modelyear from Vehicle Identifier Section (VIS)
-     isovisSeqPlant:  Plant code + sequential number
-                      from Vehicle Identifier Section (VIS)
VIN::= SEQUENCE {
  isowmi          PrintableString (SIZE(3))
(FROM("A".."H"|"J".."N"|"P"|"R".."Z"|"0".."9")),
  isovds          PrintableString (SIZE(6))
(FROM("A".."H"|"J".."N"|"P"|"R".."Z"|"0".."9")),
```

```
  isovisModelyear PrintableString (SIZE(1))
(FROM("A".."H"|"J".."N"|"P"|"R".."Z"|"0".."9")),
  isovisSeqPlant  PrintableString (SIZE(7))
(FROM("A".."H"|"J".."N"|"P"|"R".."Z"|"0".."9"))
}


- VehiclePropulsionStorageType is a collection of elements
- that contain information about the presence of propulsion
- storage inside the vehicle sending the MSD.
-
- For each storage type the following coding applies:
-   false  = indicates a type of storage not present
-   true = indicates type of storage which is present
- The following storage types are supported:
-   Gasoline tank
-   Diesel tank
-   Compressed natural gas (CNG)
-   Liquid propane gas (LPG)
-   Electric energy storage (with more than 42v and 100Ah)
-   Hydrogen storage
-   other storage
- If the type of energy storage is unknown, then all elements
- shall be set to false.
- Extendible in future versions for new fuel storage types
VehiclePropulsionStorageType::= SEQUENCE {
  gasolineTankPresent   BOOLEAN DEFAULT FALSE,
  dieselTankPresent     BOOLEAN DEFAULT FALSE,
  compressedNaturalGas  BOOLEAN DEFAULT FALSE,
  liquidPropaneGas      BOOLEAN DEFAULT FALSE,
  electricEnergyStorage BOOLEAN DEFAULT FALSE,
  hydrogenStorage       BOOLEAN DEFAULT FALSE,
  otherStorage          BOOLEAN DEFAULT FALSE,
 ...
}

- VehicleLocation:
- The current location of the vehicle
- Elements:
-  Position latitude (WGS84) in milliarcsec
-   32 bits (4 octets) allocated to make signed value handling easier
-   Real latitude values in 1 milli-arc-second units
-   Valid value range (-324000000 to 324000000)
-    calculation example:
-     48.3003333 = 48°18′1.20" N
-                = 48*60*60.000" + 18*60.000" + 1.20"
-                = 173881.200"
-                = 173881200 milliarcsec
-
-    maximum value:
-     90°00′00.000"   = 324000000
-    minimum value:
-      -90°00′00.000" = -324000000
-
-  NOTE 1: if latitude is invalid or unknown,
```

```
-   the representation of value 2147483647 shall
-   be transmitted.
-  NOTE 2: if both latitude and longitude have
-   value 0 then the location shall also be
-   interpreted as invalid/unknown.
-  NOTE 3: if the transmitter deter¬mines either
-   latitude or longitude to be invalid/unknown,
-   then it is re-commended to transmit both
-   longitude and latitude as unknown.
-  NOTE 4: if the receiver determines either
-   latitude or longitude to be invalid/unknown,
-   then it is recom-mended to interpret both
-   longitude and latitude as invalid/unknown

-  Position longitude (WGS84)
-   32 bits (4 octets) allocated to make signed value handling easier
-   Real longitude values in 1 milli-arc-second units
-   Valid value range (-648000000 to 648000000)
-
-   see 'Position latitude'
-
VehicleLocation ::= SEQUENCE {
  positionLatitude  INTEGER(-2147483648..2147483647),
  positionLongitude INTEGER(-2147483648..2147483647)
}


- VehicleLocationDelta:
- Description of (the delta of) a recent vehicle locatation
- before the incident
-  Latitude Delta (+ for North and - for South)
-  with respect to vehicleLocation.
-   1 Unit = 100 miliarcseconds, which is approximately 3m
-   Coded value range (-512..511)
-   representing -51200 to +51100 miliarcseconds,
-   or from 51,2"S to 51,1"N from the reference position

-  Longitude Delta (+ for East and - for West)
-  with respect to vehicleLocation.
-   1 Unit = 100 miliarcseconds, which is approximately 3m
-   Coded value range (-512..511)
-   representing -51200 to +51100 miliarcseconds,
-   or from 51,2"W to 51,1"E from the reference position
-
VehicleLocationDelta::= SEQUENCE {
  latitudeDelta  INTEGER (-512..511),
   longitudeDelta INTEGER (-512..511)
}


- AdditionalData:
- Further additional bytes of data encoded as in a
- seperate ASN.1 definition
- NOTE: The framework format of this field is defined here,
-       which includes a method to uniquely identify the exact
```

```
-       format of the data.
- Elements:
-   oid: Object identifier which uniquely identifies the format
-        and meaning of the data which follows.

-   data:Transparent optional additional data,
-        according to the format referenced by the oid
-        The user must ensure that the size of this element
-        is restricted to ensure that the total ECallMessage is
-        small enough for the relevant transmission medium.
AdditionalData::= SEQUENCE {
  oid  RELATIVE-OID,
   data OCTET STRING (CONTAINING CVSchemaA)
}

CVADSchemaA::= SEQUENCE {
  commercialVehicleType          CVehicleType,
  consignorPhone         NumericString(SIZE(1..17)) OPTIONAL,
  alarmInfo              CVAlarmType       OPTIONAL,
  goodsADR               CVGoodsADRType    OPTIONAL,
  goodsNonADR            CVGoodsNonADRType OPTIONAL
}

CVehicleType::= ENUMERATED {
  unknownn(0),
  tankerSingleCompartment(1),
  tankerMultiCompartment(2),
  truckPieceCargo(3),
  ...
}

CVGoodsADRType::= SEQUENCE {
  definedGoodsADR              SEQUENCE SIZE(1..7) OF CVADRCoded
OPTIONAL,
  numberOfUndefinedGoodsADR    INTEGER (0..15)
}

CVGoodsNonADRType::= SEQUENCE {
  definedGoodsNonADR           SEQUENCE SIZE(1..7) OF CVSPCCoded
OPTIONAL,
  numberOfUndefinedGoodsNonADR INTEGER (0..15)
}

CVAlarmType::= SEQUENCE {
  leakageAlarm      BOOLEAN OPTIONAL,
  fireAlarm         BOOLEAN OPTIONAL,
  highTempAlarm     BOOLEAN OPTIONAL,
  lowTempAlarm      BOOLEAN OPTIONAL,
  shockAlarm        BOOLEAN OPTIONAL,
  highPressureAlarm BOOLEAN OPTIONAL,
  lowPressureAlarm  BOOLEAN OPTIONAL,
  orientationAlarm  BOOLEAN OPTIONAL,
  otherAlarm        BOOLEAN OPTIONAL,
  ...
}
```

```
CVADRCoded::= SEQUENCE {
  cargoUNCode       INTEGER (0..9999),
  kemlerCode   ADRKemlerCode,
  packageGroup      INTEGER (1..3),
  quantity          INTEGER (0..99),
  quantityUnit    HGVQuantityUnit
}

CVQuantityUnit::= ENUMERATED {
  qtyInTonnesNet(0),
  qtyInTonnesGross(1),
  qtyInCubicM(5),
  ...
}

ADRKemlerCode::= SEQUENCE {
  firstDigit   PrintableString (SIZE(1)) (FROM("2".."9"|"X")) DEFAULT
"2",
  secndDigit        PrintableString (SIZE(1)) (FROM("0"|"2".."9"|"X"))
OPTIONAL,
  thirdDigit        PrintableString (SIZE(1)) (FROM("0"|"2".."9"|"X"))
OPTIONAL
}

CVSPCCoded::= SEQUENCE {
  cargoSPSCode      NumericString (SIZE(2..8)),
  containerTypeCode CVisoContainerType OPTIONAL
}

CVisoContainerType::= ENUMERATED {
  containerTypeG0(0),  containerTypeG1(1),  containerTypeG2(2),
containerTypeG3(3),
  containerTypeV0(4),  containerTypeV2(5),  containerTypeV4(6),
  containerTypeR0(7),  containerTypeR1(8),  containerTypeR2(9),
containerTypeR3(10),
  containerTypeH0(11), containerTypeH1(12), containerTypeH2(13),
containerTypeH5(14),
   containerTypeH6(15),
  containerTypeU0(16), containerTypeU1(17), containerTypeU2(18),
containerTypeU3(19),
   containerTypeU4(20), containerTypeU5(21),
  containerTypeT0(22), containerTypeT1(23), containerTypeT2(24),
containerTypeT3(25),
   containerTypeT4(26), containerTypeT5(27), containerTypeT6(28),
containerTypeT7(29),
    containerTypeT8(30), containerTypeT9(31),
  containerTypeB0(32), containerTypeB1(33), containerTypeB3(34),
containerTypeB4(35),
   containerTypeB5(36), containerTypeB6(37),
  containerTypeP0(38), containerTypeP1(39), containerTypeP2(40),
containerTypeP3(41),
   containerTypeP4(42), containerTypeP5(43),
  containerTypeS0(44), containerTypeS1(45), containerTypeS2(46),
  ...
```

```
}

END
```

B.3 Example

Given the following UPER encoded message:

```
02385C06 80E30A51 439E2955 D4380080 0437F80A 31056690 23F8A711 66932185
B0041500 43C04040 2021FC54 84630442 468E63A0 4B319AB1 0000
```

Decoding with the normal (not extended) rules gives the following result

```
value ECallMessage::=
{
  msdVersion 2,
  msd
    CONTAINING
    {
      msdStructure
      {
        messageIdentifier 1,
        control
        {
          automaticActivation TRUE,
          testCall FALSE,
          positionCanBeTrusted TRUE,
          vehicleType passengerVehicleClassM1
        },
        vehicleIdentificationNumber
        {
          isowmi "ECA",
          isovds "LLEXAM",
          isovisModelyear "P",
          isovisSeqPlant "LE02013"
        },
        vehiclePropulsionStorageType
        {
          gasolineTankPresent TRUE,
          dieselTankPresent FALSE,
          compressedNaturalGas FALSE,
          liquidPropaneGas FALSE,
          electricEnergyStorage FALSE,
          hydrogenStorage FALSE,
          otherStorage FALSE
        },
        timestamp 1367878452,
        vehicleLocation
        {
          positionLatitude 18859320,
          positionLongitude 187996428
        },
        vehicleDirection 45,
        recentVehicleLocationN1
        {
          latitudeDelta 0,
```

```
          longitudeDelta 10
        },
        recentVehicleLocationN2
        {
          latitudeDelta 0,
          longitudeDelta 30
        },
        numberOfPassengers 2
      },
      optionalAdditionalData
      {
        oid { 1 1 },
        data 'E2A42318221234731D02598CD58800'H
      }
    }
}
```

Decoding the same message with the extended ruleset gives:

```
value1 ECallMessage::=
{
  msdVersion 2,
  msd
    CONTAINING
    {
      msdStructure
      {
        messageIdentifier 1,
        control
        {
          automaticActivation TRUE,
          testCall FALSE,
          positionCanBeTrusted TRUE,
          vehicleType passengerVehicleClassM1
        },
        vehicleIdentificationNumber
        {
          isowmi "ECA",
          isovds "LLEXAM",
          isovisModelyear "P",
          isovisSeqPlant "LE02013"
        },
        vehiclePropulsionStorageType
        {
          gasolineTankPresent TRUE,
          dieselTankPresent FALSE,
          compressedNaturalGas FALSE,
          liquidPropaneGas FALSE,
          electricEnergyStorage FALSE,
          hydrogenStorage FALSE,
          otherStorage FALSE
        },
        timestamp 1367878452,
        vehicleLocation
```

```
        {
          positionLatitude 18859320,
          positionLongitude 187996428
        },
        vehicleDirection 45,
        recentVehicleLocationN1
        {
          latitudeDelta 0,
          longitudeDelta 10
        },
        recentVehicleLocationN2
        {
          latitudeDelta 0,
          longitudeDelta 30
        },
        numberOfPassengers 2
      },
      optionalAdditionalData
      {
        oid { 1 1 },
        data
          CONTAINING
          {
            commercialVehicleType tankerSingleCompartment,
            consignorPhone "31207110123",
            alarmInfo
            {
              leakageAlarm FALSE,
              fireAlarm TRUE,
              highTempAlarm TRUE,
              highPressureAlarm TRUE,
              lowPressureAlarm FALSE
            },
            goodsADR
            {
              definedGoodsADR
              {
                {
                  cargoUNCode 1203,
                  kemlerCode "3",
                  packageGroup 2,
                  quantityAmount 44,
                  quantityUnit qtyInCubicM
                }
              },
              numberOfUndefinedGoodsADR 0
            }
          }
      }
    }
}
```

Bibliography

[1] ISO/IEC/ITU IS/Rec 8824-1/ X.680/8824-1:2002 *Information technology - Abstract Syntax Notation One (ASN.1): Specification of basic notation Cor 1 2006*

[2] ISO/IEC/ITU IS/Rec 8824-2 /X.681:2002 *Information technology - Abstract Syntax Notation One (ASN.1): Information object specification Amd1 2004*

[3] ISO/IEC/ITU IS/Rec 8825-1/X.690:2002 *Information technology - ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER).Amd 1 2004 JTC1/SC6*

[4] ISO/IEC/ITU IS/Rec 8825-2/X.691:2002 *Information technology - ASN.1 encoding rules: Specification of Packed Encoding Rules (PER) Amd 1 2004. Cor 1 2006. Amd 2 2007. Cor 2 2006*

[5] UN Dangerous-Goods-Emergency-Action-Code-List-2011

[6] UNSPSC® United Nations Standard Products and Services Code®

This page deliberately left blank

British Standards Institution (BSI)

BSI is the national body responsible for preparing British Standards and other standards-related publications, information and services.

BSI is incorporated by Royal Charter. British Standards and other standardization products are published by BSI Standards Limited.

About us

We bring together business, industry, government, consumers, innovators and others to shape their combined experience and expertise into standards -based solutions.

The knowledge embodied in our standards has been carefully assembled in a dependable format and refined through our open consultation process. Organizations of all sizes and across all sectors choose standards to help them achieve their goals.

Information on standards

We can provide you with the knowledge that your organization needs to succeed. Find out more about British Standards by visiting our website at bsigroup.com/standards or contacting our Customer Services team or Knowledge Centre.

Buying standards

You can buy and download PDF versions of BSI publications, including British and adopted European and international standards, through our website at bsigroup.com/shop, where hard copies can also be purchased.

If you need international and foreign standards from other Standards Development Organizations, hard copies can be ordered from our Customer Services team.

Copyright in BSI publications

All the content in BSI publications, including British Standards, is the property of and copyrighted by BSI or some person or entity that owns copyright in the information used (such as the international standardization bodies) and has formally licensed such information to BSI for commercial publication and use.

Save for the provisions below, you may not transfer, share or disseminate any portion of the standard to any other person. You may not adapt, distribute, commercially exploit, or publicly display the standard or any portion thereof in any manner whatsoever without BSI's prior written consent.

Storing and using standards

Standards purchased in soft copy format:

- A British Standard purchased in soft copy format is licensed to a sole named user for personal or internal company use only.
- The standard may be stored on more than 1 device provided that it is accessible by the sole named user only and that only 1 copy is accessed at any one time.
- A single paper copy may be printed for personal or internal company use only.

Standards purchased in hard copy format:

- A British Standard purchased in hard copy format is for personal or internal company use only.
- It may not be further reproduced – in any format – to create an additional copy. This includes scanning of the document.

If you need more than 1 copy of the document, or if you wish to share the document on an internal network, you can save money by choosing a subscription product (see 'Subscriptions').

Reproducing extracts

For permission to reproduce content from BSI publications contact the BSI Copyright & Licensing team.

Subscriptions

Our range of subscription services are designed to make using standards easier for you. For further information on our subscription products go to bsigroup.com/subscriptions.

With **British Standards Online (BSOL)** you'll have instant access to over 55,000 British and adopted European and international standards from your desktop. It's available 24/7 and is refreshed daily so you'll always be up to date.

You can keep in touch with standards developments and receive substantial discounts on the purchase price of standards, both in single copy and subscription format, by becoming a **BSI Subscribing Member**.

PLUS is an updating service exclusive to BSI Subscribing Members. You will automatically receive the latest hard copy of your standards when they're revised or replaced.

To find out more about becoming a BSI Subscribing Member and the benefits of membership, please visit bsigroup.com/shop.

With a **Multi-User Network Licence (MUNL)** you are able to host standards publications on your intranet. Licences can cover as few or as many users as you wish. With updates supplied as soon as they're available, you can be sure your documentation is current. For further information, email subscriptions@bsigroup.com.

Revisions

Our British Standards and other publications are updated by amendment or revision.

We continually improve the quality of our products and services to benefit your business. If you find an inaccuracy or ambiguity within a British Standard or other BSI publication please inform the Knowledge Centre.

Useful Contacts

Customer Services
Tel: +44 345 086 9001
Email (orders): orders@bsigroup.com
Email (enquiries): cservices@bsigroup.com

Subscriptions
Tel: +44 345 086 9001
Email: subscriptions@bsigroup.com

Knowledge Centre
Tel: +44 20 8996 7004
Email: knowledgecentre@bsigroup.com

Copyright & Licensing
Tel: +44 20 8996 7070
Email: copyright@bsigroup.com

BSI Group Headquarters

389 Chiswick High Road London W4 4AL UK

bsi.