INTERNATIONAL STANDARD

ISO 24534-3

Second edition
2016-04-01

Intelligent transport systems — Automatic vehicle and equipment identification — Electronic registration identification (ERI) for vehicles —

Part 3: **Vehicle data**

Systèmes de transport intelligents — Identification automatique des véhicules et des équipements — Identification d'enregistrement électronique (ERI) pour les véhicules —

Partie 3: Données du véhicule

Reference number
ISO 24534-3:2016(E)

ISO copyright office
Ch. de Blandonnet 8 • CP 401
CH-1214 Vernier, Geneva, Switzerland
Tel. +41 22 749 01 11
Fax +41 22 749 09 47
copyright@iso.org
www.iso.org

Contents

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular the different approval criteria needed for the different types of ISO documents should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation on the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the WTO principles in the Technical Barriers to Trade (TBT) see the following URL: Foreword - Supplementary information

The committee responsible for this document is ISO/TC 204, *Intelligent transport systems*.

This second edition cancels and replaces the first edition (ISO 2453-3:2010), which has been technically revised.

ISO 24534 consists of the following parts, under the general title *Automatic vehicle and equipment identification — Electronic registration identification (ERI) for vehicles*:

— *Part 1: Architecture*

— *Part 2: Operational requirements*

— *Part 3: Vehicle data*

— *Part 4: Secure communications using asymmetrical techniques*

— *Part 5: Secure communications using symmetrical techniques.*

Introduction

A quickly emerging need has been identified within administrations to improve the unique identification of vehicles for a variety of services. Situations are already occurring where manufacturers intend to fit lifetime tags to vehicles. Various governments are considering the needs and benefits of ERI, such as legal proof of vehicle identity with potential mandatory usages. There is a commercial and economic justification both in respect of tags and infrastructure that a standard enables an interoperable solution.

Electronic registration identification (ERI) is a means of uniquely identifying road vehicles. The application of ERI will offer significant benefits over existing techniques for vehicle identification. It will be an enabling technology for the future management and administration of traffic and transport including applications in free-flow, multi-lane, traffic conditions with the capability to support mobile transactions. ERI addresses the need of authorities, and other users for a trusted electronic identification, including roaming vehicles.

This part of ISO 24534 specifies the vehicle-related data that can be exchanged between an onboard electronic registration tag (ERT) and an ERI reader/writer inside or outside the vehicle. The vehicle-related data consist of the vehicle identifier and may also include additional vehicle data as typically included in a vehicle registration certificate.

This part of ISO 24534 does not provide any accurate definitions for additional vehicle data items. This is left to the local registration authorities and/or local legislation. This part of ISO 24534 only provides the means for an unambiguous exchange of vehicle parameters registered by local registration authorities.

This part of ISO 24534 makes use of the basic automatic vehicle identification (AVI) definitions in ISO 14816.

Intelligent transport systems — Automatic vehicle and equipment identification — Electronic registration identification (ERI) for vehicles —

Part 3: Vehicle data

1 Scope

This part of ISO 24534 provides the requirements for an electronic registration identification (ERI) that is based on an identifier assigned to a vehicle (e.g. for recognition by national authorities) suitable to be used for the following:

- electronic identification of local and foreign vehicles by national authorities;
- vehicle manufacturing, in-life-maintenance, and end-of-life identification (vehicle life cycle management);
- adaptation of vehicle data, e.g. in case of international re-sales;
- safety-related purposes;
- crime reduction;
- commercial services;
- adhering to privacy and data protection regulations.

This part of ISO 24534 defines the vehicle identification data. This data is called the ERI data and includes the following:

- the vehicle identifier;
- possible additional vehicle-related information (as typically included in a vehicle registration certificate).

All additional vehicle data elements are defined as optional. It is left to local legislation and/or the discretion of a registration authority to use or not to use a particular data element. If used, the value is assumed to be the one registered by the registration authority in accordance with local legislation. This part of ISO 24534 only provides the syntax for all these data elements.

NOTE The secure application layer interfaces for the exchange of ERI data with an ERI reader or writer are specified in ISO 24534-4 and in ISO 24534-5.

2 Normative references

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

ISO 612:1978, *Road vehicles — Dimensions of motor vehicles and towed vehicles — Terms and definitions*

ISO 1176:1990, *Road vehicles — Masses — Vocabulary and codes*

ISO 3779, *Road vehicles — Vehicle identification number (VIN) — Content and structure*

ISO 3780, *Road vehicles — World manufacturer identifier (WMI) code*

ISO 3833, *Road vehicles — Types — Terms and definitions*

ISO/IEC 8824 (all parts), *Information technology — Abstract Syntax Notation One (ASN.1)*

ISO/IEC 8825-2:2008, *Information technology — ASN.1 encoding rules: Specification of Packed Encoding Rules (PER)*

ISO 14816, *Road transport and traffic telematics — Automatic vehicle and equipment identification — Numbering and data structure*

3 Terms and definitions

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

3.1
additional vehicle data
ERI data (3.5) in addition to the vehicle identifier

3.2
attribute
type (3.12) with an associated identifier

[SOURCE: ISO/IEC 8824-1:2008, Annex G.2.15.1, modified]

3.3
distinguishing identifier
information which unambiguously distinguishes an entity

[SOURCE: ISO/IEC 9798-1, 3.12, modified]

3.4
electronic registration identification
ERI
action or act of identifying a vehicle with electronic means for purposes as mentioned in the scope of this part of ISO 24534

3.5
ERI data
vehicle identifying data which can be obtained from an *ERT* (3.6)

Note 1 to entry: ERI data consist of the vehicle identifier and possible *additional vehicle data* (3.1).

3.6
electronic registration tag
ERT
onboard ERI device that contains the *ERI data* (3.5) including relevant security provisions and one or more interfaces to access that data

Note 1 to entry: In case of high security, the ERT is a secure application module (SAM).

Note 2 to entry: The ERT may be a separate device or may be integrated into an onboard device that also provides other capabilities (e.g. DSRC communications).

3.7
periodic motor vehicle test
compulsory periodic (e.g. annual) test of the roadworthiness of a motor vehicle of above a specified age or a certificate of passing such a test

EXAMPLE The MOT test in the United Kingdom is an example.

3.8
privacy
right of individuals to control or influence what information related to them may be collected and stored and by whom and to whom that information may be disclosed

[SOURCE: ISO 7498-2, 3.3.43]

Note 1 to entry: Since this term relates to the right of individuals, it cannot be very precise and its use should be avoided except as a motivation for requiring security.

3.9
registration authority
<for vehicles> authority responsible for the registration and maintenance of vehicle records

Note 1 to entry: The authority may provide vehicle records to accredited organizations.

3.10
registration authority
<for ERI data> organization responsible for writing *ERI data* (3.5) and security data according to local legislation

Note 1 to entry: The registration authority for ERI data may be the same as the *registration authority* (3.9) for vehicles. This part of ISO 24534, however, does not require this.

3.11
registration certificate
vehicle registration document (paper or smart card) issued by the *registration authority* (3.9) for vehicles in which the vehicle and its owner or lessee are registered

3.12
type
named set of values

[SOURCE: ISO/IEC 8824-1, 3.8.86]

4 Abbreviations

AEI	Automatic Equipment Identification
ASN.1	Abstract Syntax Notation One [as defined in ISO 8824 (all parts)]
AVI	Automatic Vehicle Identification
EEA	European Economic Area
EFC	Electronic Fee Collection
EN	Europäische Norm (German), English: European Standard
ENV	Europäische Norm Vorausgabe (German), English: European Pre-Standard
ERI	Electronic Registration Identification
ERT	Electronic Registration Tag
EU	European Union
IEC	International Electrotechnical Commission
ISO	International Organization for Standardization
VIN	Vehicle Identification Number

5 Requirements

5.1 Vehicle identification data

This subclause is informative only.

The secure onboard environment in which the vehicle identification data is stored is called the electronic registration tag (ERT).

Clause 5 provides an abstract definition of the ERI data to be exchanged between the ERT and an ERI reader or writer. The abstract definitions are defined using abstract syntax notation one (ASN.1) as defined in ISO 8824 (all parts).

The identifier used to identify a vehicle is called the vehicle identifier or vehicleId. The preferred vehicle identifier is the VIN that is assigned to the vehicle by its manufacturer in accordance with ISO 3779.

However, in order to make this part of ISO 24534 also applicable in countries where the VIN is not used, an alternative is also supported (see 5.2). The fundamental requirement is that the combination of a registration authority and a vehicle identifier should be globally distinguishing.

NOTE 1 As two vehicles built 30 years after each other may have the same VIN, the VIN is not 100 % unique.

NOTE 2 Empirical data has shown that a database of a registration authority may contain duplicate VIN numbers.

NOTE 3 In this part of ISO 24534, the combination of the almost unique vehicleId and a unique ERT number may be used as the unambiguous distinguishing identifier. The ERT number is a unique read-only identifier that is written into the ERT during ERT manufacturing time (see EN ISO 24534-4 and ISO 24534-5 for details).

Apart from the vehicle identifier, this part of ISO 24534 also supports the use of additional vehicle data as typically included in a vehicle registration certificate. This additional vehicle data may, for example, be used as

— additional identification information to improve the trust in a vehicle identifier, and

— certified vehicle information for other applications (e.g. for tolling to determine a tariff).

5.2 Vehicle identifier

The VehicleId type shall be used for the vehicle identifier according to local legislation and is defined as follows.

```
VehicleId ::= CHOICE {
          vin                        VIN,               -- preferred choice
          raSpecificVehicleId       RaSpecificVehicleId,
          …
          }
VIN ::= CS5
```

NOTE 1 The '...' at the end of the definition designates that the type VehicleId may be extended with additional components at the end of the type definition in new versions of this part of ISO 24534, e.g. to cope with a new VIN standard.

The vehicle identifier should be a globally distinguishing identifier.

NOTE 2 When identifying a vehicle, the ERT always delivers the VehicleId in combination with the identifier of the registration authority and the ERT number. The identifier of the registration authority may be used to obtain additional information about the vehicle. The ERT number is an extra unique identifier from another source that may be used to resolve potential disputes about the VIN of a vehicle.

NOTE 3 The choice of which alternative is used is outside the scope of this part of ISO 24534. It may, for example, depend on local legislation.

The VIN alternative, if used, shall be of type VIN and is the preferred vehicle identifier. The type VIN is identical to the type CS5 as defined in ISO 14816. The value of the VIN alternative shall be the value of the VIN as assigned conforming to ISO 3779 by a manufacturer or a registration authority.

The RaSpecificVehicleId alternative, if used, shall contain a globally distinguishing identifier for the vehicle and shall be of type raSpecificVehicleId as defined below.

```
RaSpecificVehicleId ::= SEQUENCE {
         wmi                          UTF8String (SIZE(3)),
         nonIsoStandardId             UTF8String (SIZE (1..20))
         }
```

The wmi component shall contain the world manufacturer identifier (WMI) code of the organization that assigned the nonIsoStandardId value and the WMI code shall be assigned to this organization according to ISO 3780.

The nonIsoStandardId component shall be of type UTF8String with a maximum length of 20 characters.

NOTE Any additional meaning conveyed in the value of a nonIsoStandardId component is outside the scope of this part of ISO 24534.

5.3 ERI data type

The EriData type shall be used for the ERI data and is defined as follows:

```
EriData ::= SEQUENCE {
        vehicleId             VehicleId,
        additionalEriData     AdditionalEriData OPTIONAL
        }
```

The vehicleId component shall contain the vehicle's identifier as defined in 5.2.

The additionalEriData component, if present, shall contain the additional ERI data.

5.4 Additional ERI data type

The type AdditionalEriData type is used for the additional ERI data and is defined as follows:

```
AdditionalEriData ::= CHOICE {
          additionalEriRegistrationData  AdditionalEriRegistrationData, -          -
preferred choice
          …,
          raSpecificAdditionalEriData   OCTET STRING (SIZE (0..1024))
                                         -- only to be used if
AdditionalEriRegistrationData is not supported
          }
```

The additionalEriRegistrationData alternative is the preferred alternative and shall be chosen whenever a value of the type AdditionalEriRegistrationData can be used.

The raSpecificAdditionalEriData alternative is of type OCTET STRING with a maximum length of 1 024 octets and shall only be used if a value of additionalEriRegistrationData cannot be used.

NOTE The '…' in the definition designates that the type AdditionalEriData may be extended with additional alternatives at the end of the type definition in new versions of this part of ISO 24534, e.g. to cope with a new version of the alternative ERI registration data.

5.5 Additional ERI registration data

5.5.1 Additional ERI registration data type

5.5.1.1 Definition of the additional ERI registration data type

The AdditionalEriRegistrationData type contains the vehicle related data typically found in a vehicle registration certificate and is defined as follows.

```
AdditionalEriRegistrationData ::= SEQUENCE {
          -- Administrative data
          registrationAuthority         RegistrationAuthority OPTIONAL,
          vehicleIdStatus               VehicleIdStatus OPTIONAL,
          dateOfFirstRegistration       DateOfFirstRegistration OPTIONAL,
          dateOfRegistration            DateOfRegistration OPTIONAL,
          validThru                     ValidThru OPTIONAL,
          chassisNumber                 ChassisNumber (SIZE (1..23))
                                        OPTIONAL,
          registrationNumber            RegistrationNumber OPTIONAL,


          -- Vehicle type
          vehicleMake                   VehicleMake OPTIONAL,
          vehicleType                   VehicleType OPTIONAL,
          vehicleTypeStatus             VehicleTypeStatus OPTIONAL,
          commercialDescription         CommercialDescription OPTIONAL,
          typeApprovalNumber            TypeApprovalNumber OPTIONAL,
          vehicleCategory               VehicleCategory OPTIONAL,
          vehicleTaxCategory            VehicleTaxCategory OPTIONAL,
          euVehicleCategoryCode         EuVehicleCategoryCode OPTIONAL,
          raSpecificVehicleClass1      RaSpecificVehicleClass1 OPTIONAL,
          raSpecificVehicleClass2      RaSpecificVehicleClass2 OPTIONAL,
          raSpecificVehicleClass3      RaSpecificVehicleClass3 OPTIONAL,
          vehicleUse                    VehicleUse OPTIONAL,
          privateUse                    PrivateUse OPTIONAL,
          colour                        VehicleColour OPTIONAL,


          -- Vehicle shape
          length                        VehicleLength OPTIONAL,
          width                         VehicleWidth OPTIONAL,
          height                        VehicleHeight OPTIONAL,
          wheelbase                     Wheelbase OPTIONAL,
          bodyShape                     VehicleBodyShape OPTIONAL,
          euBodyWorkType                EuBodyWorkType OPTIONAL,
          iso3833VehicleType            Iso3833VehicleType OPTIONAL,


          -- Vehicle number of passengers, axles, and mass
          numberOfSeats                 NumberOfSeats OPTIONAL, --
                                        including the driver seat
          numberOfStandingPlaces        NumberOfStandingPlaces OPTIONAL,
          maxNumberOfPassengers         MaxNumberOfPassengers OPTIONAL,
                                        -- including the driver
          unladenWeight                 UnladenWeight OPTIONAL,
          maxDesignLadenMass            MaxDesignLadenMass OPTIONAL,
          maxAuthorizedLadenMass        MaxAuthorizedLadenMass OPTIONAL,
          maxAuthorizedTrainMass        MaxAuthorizedTrainMass OPTIONAL,
          maxAuthorizedPayload          MaxAuthorizedPayload OPTIONAL,
          numberOfAxles                 NumberOfAxles OPTIONAL,
          authorizedAxleLadenMass       AuthorizedAxleLadenMass OPTIONAL,
                                        -- from front to rear axle
          maxTowableMassBrakedTrailer   MaxTowableMassBrakedTrailer
OPTIONAL,
          maxTowableMassUnbrakedTrailer MaxTowableMassUnbrakedTrailer
OPTIONAL,


          -- Vehicle engine and power source
          engineId                      EngineId (SIZE (1..60)) OPTIONAL,
          primeEngineType               PrimeEngineType OPTIONAL,
          enginePowerSources            EnginePowerSources OPTIONAL,
          primePowerSource              PrimePowerSource OPTIONAL,
          engineMaxNetPower             EngineMaxNetPower OPTIONAL,
          engineDisplacement            EngineDisplacement OPTIONAL,
          ratedEngineSpeed              RatedEngineSpeed OPTIONAL,
          powerWeightRatio              PowerWeightRatio OPTIONAL,
          maxSpeed                      MaxSpeed OPTIONAL,
          fuelTanksCapacity             FuelTanksCapacity OPTIONAL,


          -- Environmental characteristics
          stationarySoundLevel          StationarySoundLevel OPTIONAL,
          engineSpeed                   EngineSpeed OPTIONAL,
          driveBySoundLevel             DriveBySoundLevel OPTIONAL,
          emissionCO                    EmissionCO OPTIONAL,
          emissionHC                    EmissionHC OPTIONAL,
          emissionNOx                   EmissionNOx OPTIONAL,
          emissionHCandNOx              EmissionHCandNOx OPTIONAL,
          particulatesForDiesel         ParticulatesForDiesel OPTIONAL,
          correctedAbsorptionCoefficient CorrectedAbsorptionCoefficient
OPTIONAL,
          emissionCO2                   EmissionCO2 OPTIONAL,
          combinedFuelConsumption       CombinedFuelConsumption OPTIONAL,
          environmentalCategory         EnvironmentalCategory OPTIONAL,
          euroType                      EuroType OPTIONAL,


          -- Others
          lastOfficialTestData         OfficialVehicleTestData OPTIONAL,
          …,
          raSpecificData               RaSpecificData OPTIONAL
          axlesPerAxleGroup             AxlesPerAxleGroup OPTIONAL,
                                        -- from front to rear axle group
          authorizedAxleGroupLadenMass  AuthorizedAxleGroupsLadenMass
OPTIONAL,
                                        -- from front to rear axle group


          -- ERI data used for EFC (types imported from
ElectronicRegistrationIdentificationEfcVehicleData
          efcVehicleDimensions          VehicleDimensions,
          efcPassengerCapacity          PassengerCapacity,
          efcVehicleWeightLimits        VehicleWeightLimits,
          efcAxleWeightLimits           AxleWeightLimits,
          efcVehicleSpecificCharacteristics
VehicleSpecificCharacteristics,
          efcTrailerCharacteristics     TrailerCharacteristics,
          efcEngine                     Engine,
          efcSoundLevel                 SoundLevel,
          efcCO2EmissionValue           CO2EmissionValue,
          efcExhaustEmissionValues      ExhaustEmissionValues,
          efcDieselEmissionValues       DieselEmissionValue,
          ...
          }
```

The type of the components of the `AdditionalEriRegistrationData` type is defined in 5.5 below.

NOTE 1 The '...' at the end of the `AdditionalEriRegistrationData` definition designates that the `AdditionalEriRegistrationData` type may be still extended with additional components at the end of the type definition in new versions of this part of ISO 24534.

All components are optional. Whether or not an optional component is present or absent depends on local legislation and/or the discretion of the vehicle's registration authority.

The precise meaning of a value of a component, if present, shall be determined by local legislation and/or the vehicle's registration authority and shall always take precedence over a definition in this part of ISO 24534.

NOTE 2 This part of ISO 24534 only facilitates the exchange of ERI registration data values for the purpose of vehicle identification. Both the precise definition of terms and the assignment of values in a particular state or country are outside the scope of this part of ISO 24534.

In order to maintain consistency with the vehicle's registration certificate, the value of a component, if present and applicable, should be equal to or at least as precise as the value of the corresponding data item on the vehicle's registration certificate.

5.5.1.2 Administrative data components

The `registrationAuthority` component, if present, shall identify the registration authority that registered the vehicle.

The `vehicleIdStatus` component, if present, shall specify the status of the vehicle identifier.

The `vehicleIdStatus` component shall not be present for a VIN-based vehicle identifier.

The `dateOfFirstRegistration` component, if present, shall specify the date of the first registration of the vehicle with its current registration authority.

The `dateOfRegistration` component, if present, shall specify the date of the registration to which this ERI data refers.

The `validThru` component, if present, shall specify the last day the ERI data is valid. If not present, the validity period may be assumed to be unlimited.

The `chassisNumber` component, if present, shall specify the chassis number of the vehicle.

The `registrationNumber` component, if present, shall specify the vehicle's registration number as assigned by the registration authority.

5.5.1.3 Vehicle type components

The `vehicleMake` component, if present, shall specify the make of the vehicle as assigned by the vehicle manufacturer.

The `vehicleType` component, if present, shall specify the variant, if applicable, and/or the version, if applicable, of the vehicle as assigned by the vehicle manufacturer.

The `vehicleTypeStatus` component, if present, shall specify the status of the vehicle type.

The `commercialDescription` component, if present, shall contain the commercial description(s) of the vehicle.

The `typeApprovalNumber` component, if present, shall specify the type-approval number.

The `vehicleCategory` component, if present, shall specify the vehicle category according to local legislation.

The `vehicleTaxCategory` component, if present, shall specify the vehicle tax category according to local legislation.

The `euVehicleCategoryCode` component, if present, shall specify the vehicle category according to the EU directives EU 2001/116, EU 2002/24, and UNECE 1999.

The `raSpecificVehicleClass1` component, if present, shall contain a registration authority specific vehicle class, category, or code.

The `raSpecificVehicleClass2` component, if present, shall contain a registration authority specific vehicle class, category, or code.

The `raSpecificVehicleClass3` component, if present, shall contain a registration authority specific vehicle class, category, or code.

The `vehicleUse` component, if present, shall specify the use of the vehicle.

The `privateUse` component, if present, shall specify whether the vehicle is for private or for commercial use.

The colour component, if present, shall specify the colour of the vehicle.

5.5.1.4 Vehicle shape components

The `length` component, if present, shall specify the length of the vehicle.

The `width` component, if present, shall specify the width of the vehicle.

The `height` component, if present, shall specify the height of the vehicle.

The `wheelbase` component, if present, shall specify the wheelbase of the vehicle.

The `bodyShape` component, if present, shall specify the shape of the body of the vehicle.

The `euBodyWorkType` component, if present, shall specify the type of body work of the vehicle according to EU 2001/116.

The `iso3833VehicleType` component, if present, shall specify the type of the vehicle according to ISO 3833.

5.5.1.5 Vehicle number of passengers, axles, and mass components

The `numberOfSeats` component, if present, shall specify the number of seats including the driver seat.

The `numberOfStandingPlaces` component, if present, shall specify the number of standing places.

The `maxNumberOfPassenger` component, if present, shall specify the maximum permissible number of passengers (including the driver) that may use the vehicle.

The `unladenWeight` component, if present, shall specify the nominal unladen mass of the vehicle with bodywork.

The `maxDesignLadenMass` component, if present, shall specify the maximum technically permissible total mass of the vehicle including payload (but excluding the weight of trailers).

The `maxAuthorizedLadenMass` component, if present, shall specify the maximum permissible total mass of the vehicle including payload (but excluding the mass of trailers) when in service in the jurisdiction of the registration authority.

The `maxAuthorizedTrainMass` component, if present, shall specify the maximum permissible total mass in kilograms of the whole vehicle including payload (and including trailers and the payload of trailers) when in service in the jurisdiction of the registration authority.

The `maxAuthorizedPayload` component, if present, shall specify the maximum permissible payload of the vehicle (but excluding the payload of trailers).

The `numberOfAxles` component, if present, shall specify the number of axles of the vehicle (including lifted axles).

The `axlesPerAxleGroup` component, if present, shall specify the number of axles of each axle group of the vehicle (including lifted axles).

The `authorizedAxleLadenMass` component, if present, shall specify the maximum laden mass on each axle in kilograms.

The `authorizedAxleGroupLadenMass` component, if present, shall specify the maximum laden mass on each axle group in kilograms

The `maxTowableMassBrakedTrailer` component, if present, shall specify the technically permissible maximum braked towable mass of the trailer.

The `maxTowableMassUnbrakedTrailer` component, if present, shall specify the technically permissible maximum unbraked towable mass of the trailer.

5.5.1.6 Vehicle engine and power source components

The `engineId` component, if present, shall specify the engine identification number.

The `primeEngineType` component, if present, shall specify type of the prime engine of the vehicle as assigned by the manufacturer.

The `enginePowerSource` component, if present, shall specify the power source(s) of the vehicle.

The primePowerSource component, if present, shall specify the power source of the prime engine.

The engineMaxNetPower component, if present, shall specify the nominal (maximum net) power of the engine.

The engineDisplacement component, if present, shall specify the displacement of the engine.

The ratedEngineSpeed component, if present, shall specify the rated speed.

The powerWeightRatio component, if present, shall specify the power/weight ratio of the vehicle.

The maxSpeed component, if present, shall specify the maximum speed of the vehicle.

The fuelTankCapacity component, if present, shall specify the capacity of the fuel tank(s).

5.5.1.7 Environmental characteristics components

The stationarySoundLevel component, if present, shall specify the stationary sound level of the vehicle.

The engineSpeed component, if present, shall specify the engine speed.

The driveBySoundLevel component, if present, shall specify the drive-by sound level of the vehicle.

The emissionCO component, if present, shall specify the CO exhaust emission.

The emissionHC component, if present, shall specify the HC exhaust emission.

The emissionNOx component, if present, shall specify the NOx exhaust emission.

The emissionHCandNOx component, if present, shall specify the HC plus COx exhaust emission.

The particulatesForDiesel component, if present, shall specify the exhaust emission of particulates for diesel.

The correctedAbsorptionCoefficient component, if present, shall specify the corrected absorption coefficient for diesel.

The emissionCO2 component, if present, shall specify the CO2 exhaust emission.

The combinedFuelConsumption component, if present, shall specify the combined fuel consumption of the vehicle.

The environmentalCategory component, if present, shall be of type EnvironmentalCategory and shall specify the environmental category of the vehicle.

The euroType component, if present, shall reference for light duty vehicles to the environmental category Euro 1...6 according to EU directive 70/220/EEC and EU Regulation 715/2007 and for heavy duty vehicles to the category Euro I...VI according to EU Directive 1988/77, EU Directive 2005/55, and EU Regulation 595/2009.

5.5.1.8 Components used for EFC

The efcVehicleDimensions component, if present, shall specify the vehicle's overall length, height and width.

The efcPassengerCapacity component, if present, shall specify both the number of seats and the number standing places.

The efcVehicleWeightLimits component, if present, shall specify the maximum weight for the laden vehicle, the vehicle train, and the unladen vehicle weight.

The `efcAxleWeightLimits` component, if present, shall specify the maximum laden weight on the vehicle's axles.

The `efcVehicleSpecificCharacteristics` component, if present, shall specify the vehicle's environmental characteristics, engine characteristics, descriptive characteristics, and future characteristics.

The `efcTrailerCharacteristics` component, if present, shall specify for a trailer its type, the number of axles, the maximum permissible total weight, and the unladen weight.

The `efcEngine` component, if present, shall specify the engine's capacity and power.

The `efcSoundLevel` component, if present, shall specify the vehicle's stationary and drive-by soundlevels.

The `efcCO2EmissionValue` component, if present, shall specify the vehicle's emission of CO2.

The `efcExhaustEmissionValues` component, if present, shall specify the vehicle's emission of CO, HC, NOX and HC plus NOX.

The `efcDieselEmissionValues` component, if present, shall specify the emission of diesel particles and the corrected absorption coefficient.

5.5.1.9 Other components

The `lastOfficialTestData` component, if present, shall specify the data observed during the last official vehicle test, e.g. a periodic motor vehicle test or an intermediate motor vehicle test.

The `lastOfficialTestData` component shall only be present if at least one of its components is present.

NOTE The value of the `EriRegistrationData` has to be signed and therefore, needs a canonical representation.

The `raSpecificVehicleData` component, if present, may contain any additional registration authority-specific data.

5.5.2 Administrative types

```
RegistrationAuthority          ::= EntityId
VehicleIdStatus                ::= Text60
DateOfFirstRegistration        ::= DATE
DateOfRegistration             ::= DATE
ValidThru                      ::= DATE
ChassisNumber                  ::= UTF8String (SIZE (1..23)) -- incl. 3 WMI
                               characters
RegistrationNumber             ::= CS4
```

Unless specified otherwise, the definition and assignment of the values for the types defined in this subclause are at the discretion of the registration authority and/or local legislation (see 5.5.1).

The `RegistrationAuthority` type shall be used to identify a registration authority and shall be of type EntityId.

The `VehicleIdStatus` type shall be used to specify the status of a vehicle identifier.

The `VehicleIdStatus` type shall not be used for a VIN-based vehicle identifier.

EXAMPLE Possible values are, e.g. ordinary, authority assigned, the identifier of the authority that assigned the `vehicleId`, etc.

The `DateOfFirstRegistration` type shall be used to specify the date of the first registration of a vehicle with its current registration authority.

The value of a DateOfFirstRegistration type shall at least contain the year and optionally the month and/or day.

The DateOfRegistration type shall be used to specify the date of the registration to which this ERI data refers.

The ValidThru type shall be used to specify the last day the ERI data is valid. If not present, the validity period may be assumed to be unlimited.

The ChassisNumber type shall be used to specify the chassis number of a vehicle.

The RegistrationNumber type shall be used to specify the vehicle's registration number as assigned by the registration authority.

5.5.3 EU vehicle category code type

The EuVehicleCategoryCode type is defined as follows.

```
EuVehicleCategoryCode ::= CHOICE {
         euVehicleCategoryL            EuVehicleCategoryL,
-- conforms to EU 2002/24 and UNECE 1999
         euVehicleCategoryM            EuVehicleCategoryM,
-- conforms to EU 2001/116 and UNECE 1999
         euVehicleCategoryN            EuVehicleCategoryN,
-- conforms to EU 2001/116 and UNECE 1999
         euVehicleCategoryO            EuVehicleCategoryO,
-- conforms to EU 2001/116 and UNECE 1999
         euVehilcleCategoryT           NULL,    -- conforms to UNECE 1999
         euVehilcleCategoryG           NULL     -- conforms to EU 2001/116
and UNECE 1999
         }
EuVehicleCategoryL ::= ENUMERATED {l1, l2, l3, l4, l5, l6, l7}
EuVehicleCategoryM ::= ENUMERATED {m1, m2, m3}
EuVehicleCategoryN ::= ENUMERATED {n1, n2, n3}
EuVehicleCategoryO ::= ENUMERATED {o1, o2, o3, o4}
```

An EuVehicleCategoryCode value may be an euvehicleCategoryL value, an euvehicleCategoryM value, an euvehicleCategoryN value, an euvehicleCategoryO value, an euvehicleCategoryT value, or an euvehicleCategoryG value.

The euVehicleCategoryL alternative shall be chosen for a vehicle in the L category according to directive EU2002/24 or UNECE 1999.

The euVehicleCategoryL alternative, if chosen, shall have one of the values l1, l2, l3, l4, l5, l6, or l7 for, respectively, the vehicle categories L1, L2, L3, L4, L5, L6, or L7 defined in directive EU2002/24 or UNECE 1999.

NOTE 1 EU Vehicle category L is used for vehicles with less than three wheels.

The euVehicleCategoryM alternative shall be chosen for a vehicle in the M category according to directive EU2001/116 or UNECE 1999.

The euVehicleCategoryM alternative, if chosen, shall have one of the values m1, m2, or m3 for, respectively, the vehicle categories M1, M2, or M3 defined in directive EU2001/116 or UNECE 1999.

NOTE 2 EU Vehicle category M is used for passenger vehicles with M1 for vehicles with no more than eight seats in addition to the driver seat, M2 for vehicles with more than eight seats, but less than five tons, and M3 for vehicles with more than eight seats and more than five tons.

NOTE 3 The category M1 special purpose vehicles from UNECE 1999 can be described as EU2001/116 body types (see the euBodyWorkType).

The euVehicleCategoryN alternative shall be chosen for a vehicle in the N category according to directive EU2001/116 or UNECE 1999.

The euVehicleCategoryN alternative, if chosen, shall have one of the values n1, n2, or n3 for, respectively, the vehicle categories N1, N2, or N3 defined in directive EU 2001/116 or UNECE 1999.

NOTE 4 EU Vehicle category N is used for vehicles carrying goods with N1 for vehicles not exceeding 3,5 tons maximum mass, N2 for vehicles exceeding 3,5 tons maximum mass, but not exceeding 12 tons, and N3 for vehicles exceeding 12 tons maximum mass.

The euVehicleCategoryO alternative shall be chosen for a vehicle in the O category according to directive EU2001/116 or UNECE 1999.

The euVehicleCategoryO alternative, if chosen, shall have one of the values o1, o2, o3, or o4 for, respectively, the vehicle categories O1, O2, O3, or O4 defined in directive EU 2001/116 or UNECE 1999.

NOTE 5 EU Vehicle category O is used for (semi-) trailers with O1 for trailers not exceeding 0,75 tons maximum mass, O2 for trailers exceeding 0,75 tons maximum mass, but not exceeding 3,5 tons, O3 for trailers exceeding 3,5 tons maximum mass, but not exceeding 10 tons, and O4 for trailers exceeding 10 tons maximum mass.

The euVehicleCategoryT alternative shall be chosen for a vehicle in the T category according to UNECE 1999.

NOTE 6 EU Vehicle category T is used for agricultural and forestry tractors.

The euVehicleCategoryG alternative shall be chosen for a vehicle in the G category according to directive EU2001/116 or UNECE 1999.

NOTE 7 EU Vehicle category G is used for off-road vehicles.

5.5.4 Other Vehicle type types

```
VehicleMake                    ::= Text60
VehicleType                    ::= Text60
VehicleTypeStatus              ::= Text60
CommercialDescription          ::= Text60
TypeApprovalNumber             ::= UTF8String (SIZE (1..60))
VehicleCategory                ::= Text60
VehicleTaxCategory             ::= Text60
RaSpecificVehicleClass1       ::= Text60
RaSpecificVehicleClass2       ::= Text60
RaSpecificVehicleClass3       ::= Text60
VehicleUse                     ::= Text60
PrivateUse                     ::= BOOLEAN      -- False = commercial use
VehicleColour                  ::= Text60
```

Unless specified otherwise, the definition and assignment of the values for the types defined in this subclause are at the discretion of the registration authority and/or local legislation (see 5.5.1).

The VehicleMake type shall be used to specify the make of a vehicle as assigned by the vehicle manufacturer.

The VehicleType type shall be used to specify the variant, if applicable, and/or the version, if applicable, of a vehicle as assigned by the vehicle manufacturer.

The VehicleTypeStatus type shall be used to specify the status of the vehicle type.

EXAMPLE 1 Possible values are, e.g. prototype, prefabricated type, modified type, etc.

The CommercialDescription type shall be used for the commercial description(s) of a vehicle.

The TypeApprovalNumber type shall be used to specify the type-approval number, if available.

The VehicleCategory type shall be used to specify the vehicle category according to local legislation.

The VehicleTaxCategory type shall be used to specify the vehicle tax category according to local legislation.

The RaSpecificVehicleClass1 type shall be used for a registration authority specific vehicle class, category, or code.

The RaSpecificVehicleClass2 type shall be used for a registration authority specific vehicle class, category, or code.

The RaSpecificVehicleClass3 type shall be used for a registration authority specific vehicle class, category, or code.

The VehicleUse type shall be used to specify the use of a vehicle.

EXAMPLE 2 A registration authority may want to use the VehicleUse type to specify the usage of a particular category of a vehicle with values such as, e.g. private or commercial, “school bus” for a vehicle in the bus category, or “rental car” for a private car.

The PrivateUse type shall be used to specify whether a vehicle is for private or for commercial use. If for private use, the value shall be TRUE. If for commercial use, the value shall be FALSE.

The VehicleColour type shall be used to specify the colour of a vehicle.

5.5.5 ISO 3833 vehicle type

The Iso3833VehicleType type is defined as follows.

```
Iso3833VehicleType ::= INTEGER {
        passengerCar                 (0),     -- term No 3.1.1
        saloon                       (1),     -- term No 3.1.1.1 (sedan)
        convertibleSaloon            (2),     -- term No 3.1.1.2
        pullmanSaloon                (3),     -- term No 3.1.1.3
        stationWagon                 (4),     -- term No 3.1.1.4
        truckStationWagon            (5),     -- term No 3.1.1.4.1
        coupe                        (6),     -- term No 3.1.1.5 (coupé)
        convertible                  (7),     -- term No 3.1.1.6 (open
tourer, roadstar, spider)
        multipurposePassengerCar     (8),     -- term No 3.1.1.7
        forwardControlPassengerCar   (9),     -- term No 3.1.1.8
        specialPassengerCar          (10),    -- term No 3.1.1.9
        bus                          (11),    -- term No 3.1.2
        minibus                      (12),    -- term No 3.1.2.1
        urbanBus                     (13),    -- term No 3.1.2.2
        interurbanCoach              (14),    -- term No 3.1.2.3
        longDistanceCoach            (15),    -- term No 3.1.2.4
        articulatedBus               (16),    -- term No 3.1.2.5
        trolleyBus                   (17),    -- term No 3.1.2.6
        specialBus                   (18),    -- term No 3.1.2.7
        commercialVehicle            (19),    -- term No 3.1.3
        specialCommercialVehicle     (20),    -- term No 3.1.3.1
        specialVehicle               (21),    -- term No 3.1.4
        trailingTowingVehicle        (22),    -- term No 3.1.5 (draw-bar
tractor)
        semiTrailerTowingVehicle     (23),    -- term No 3.1.6 (fifth
wheel tractor)
        trailer                      (24),    -- term No 3.2.1
        busTrailer                   (25),    -- term No 3.2.1.1
        generalPurposeTrailer        (26),    -- term No 3.2.1.2
        caravan                      (27),    -- term No 3.2.1.3
        specialTrailer               (28),    -- term No 3.2.1.4
        semiTrailer                  (29),    -- term No 3.2.2
        busSemiTrailer               (30),    -- term No 3.2.2.1
        generalPurposeSemiTrailer    (31),    -- term No 3.2.2.2
        specialSemiTrailer           (32),    -- term No 3.2.2.3
        roadTrain                    (33),    -- term No 3.3.1
        passengerRoadTrain           (34),    -- term No 3.3.2
        articulatedRoadTrain         (35),    -- term No 3.3.3
        doubleRoadTrain              (36),    -- term No 3.3.4
        compositeRoadTrain           (37),    -- term No 3.3.5
        specialRoadTrain             (38),    -- term No 3.3.6
        moped                        (39),    -- term No 3.4
        motorCycle                      (40)      -- term No 3.5
        } (0..255)
```

A “term No” in a comment parts refers to the number of the corresponding term and its definition in ISO 3833.

A value of `Iso3833VehicleType` type shall conform the corresponding definition in ISO 3833.

5.5.6 Other vehicle shape types

```
VehicleLength                 ::= Millimetre
VehicleWidth                  ::= Millimetre
VehicleHeight                 ::= Millimetre
Wheelbase                     ::= Millimetre
VehicleBodyShape              ::= Text60
EuBodyWorkType                ::= UTF8String (SIZE (2))
                              -- conforms to EU 2001/116
```

Unless specified otherwise, the definition and assignment of the values for the types defined in this subclause are at the discretion of the registration authority and/or local legislation (see 5.5.1).

The `VehicleLength` type shall be used to specify the length of a vehicle in millimetres.

Unless local legislation specifies otherwise, ISO 612:1978, 6.1 applies and the length includes draw gear.

The `vehicleWidth` type shall be used to specify the width of a vehicle in millimetres.

Unless local legislation specifies otherwise, ISO 612:1978, 6.2 applies.

The `VehicleHeight` type shall be used to specify the height of a vehicle in millimetres.

Unless local legislation specifies otherwise, ISO 612:1978, 6.3 applies.

The `Wheelbase` type shall be used to specify the wheelbase of a vehicle in millimetres.

Unless local legislation specifies otherwise, ISO 612:1978, 6.4 (wheel space) applies and for vehicles with three or more axles (or semi-trailers with two or more axles) the distance is taken from the foremost wheel (the fifth wheel kingpin for semi-trailers) to the rearmost wheel.

The `VehicleBodyShape` type shall be used to specify the shape of the body of a vehicle.

The `EuBodyWorkType` type shall be used to specify the type of body work of a vehicle according to EU 2001/116.

NOTE 1 The EU bodywork type (two printable characters) is a refinement of ISO 3833.

NOTE 2 The EU bodywork type is also used for the designation of the special purpose vehicles as defined in UNECE 1999 (see EU2001/116).

5.5.7 Number of passenger, axles, and mass types

```
NumberOfSeats                ::= INTEGER      -- including the driver seat
NumberOfStandingPlaces       ::= INTEGER
```

```
MaxNumberOfPassengers        ::= INTEGER      -- including the driver
UnladenWeight                ::= Kilogram
MaxDesignLadenMass           ::= Kilogram
MaxAuthorizedLadenMass       ::= Kilogram
MaxAuthorizedTrainMass       ::= Kilogram
MaxAuthorizedPayload         ::= Kilogram
```

```
NumberOfAxles                ::= INTEGER
AxlesPerAxleGroup            ::= SEQUENCE OF Int1     -- from front to rear
axle group
AuthorizedAxleLadenMass      ::= SEQUENCE OF Kilogram     -- from front to
rear axle
AuthorizedAxleGroupsLadenMass   ::= SEQUENCE OF Kilogram  -- from front to
rear axle group
MaxTowableMassBrakedTrailer   ::= Kilogram
MaxTowableMassUnbrakedTrailer ::= Kilogram
```

Unless specified otherwise, the definition and assignment of the values for the types defined in this subclause are at the discretion of the registration authority and/or local legislation (see 5.5.1).

The `NumberOfSeats` type shall be used to specify the number of seats, including the driver seat.

The `NumberOfStandingPlaces` type shall be used to specify the number of standing places.

The `MaxNumberOfPassenger` type shall be used to specify the maximum permissible number of passengers (including the driver) that may use a vehicle.

The `UnladenWeight` type shall be used to specify the nominal unladen mass of a vehicle with bodywork in kilograms.

Unless local legislation specifies otherwise, ISO 1176:1990, 4.6 (complete vehicle kerb mass) applies.

The `MaxDesignLadenMass` type shall be used to specify the maximum technically permissible total mass of a vehicle including payload (but excluding the weight of trailers) in kilograms.

Unless local legislation specifies otherwise, ISO 1176:1990, 4.7 (maximum design total mass) applies.

The `MaxAuthorizedLadenMass` type shall be used to specify the maximum permissible total mass of a vehicle including payload (but excluding the mass of trailers) in kilograms when in service in the jurisdiction of the registration authority.

Unless local legislation specifies otherwise, ISO 1176:1990, 4.8 (maximum authorized total mass) applies.

The `MaxAuthorizedTrainMass` type shall be used to specify the maximum permissible total mass in kilograms of a whole vehicle including payload (and including trailers and the payload of trailers) when in service in the jurisdiction of the registration authority.

Unless local legislation specifies otherwise, ISO 1176:1990, 4.19 (maximum authorized mass of vehicle combination) applies.

The `MaxAuthorizedPayload` type shall be used to specify the maximum permissible payload of a vehicle (but excluding the payload of trailers) in kilograms.

Unless local legislation specifies otherwise, ISO 1176:1990, 4.10 (maximum authorized pay mass) applies.

The `NumberOfAxles` type shall be used to specify the number of axles of a vehicle (including lifted axles).

The `AxlesPerAxleGroup` type shall be used to specify the number of axles of each axle group of a vehicle (including lifted axles).

The `AuthorizedAxleLadenMass` type shall be used to specify the maximum laden mass on each axle in kilograms.

The first value in the list shall specify the maximum laden mass on the front axle, the second on the axle next to the front axle (including lifted axles), and so on.

Unless local legislation specifies otherwise, ISO 1176:1990, 4.11 (maximum design axle load) applies.

The `AuthorizedAxleGroupsLadenMass` type shall be used to specify the maximum laden mass on each axle group in kilograms.

The MaxTowableMassBrakedTrailer type shall be used to specify the technically permissible maximum braked towable mass of a trailer in kilograms.

Unless local legislation specifies otherwise, ISO 1176:1990, 4.18 (maximum design towed mass) applies.

The MaxTowableMassUnbrakedTrailer type shall be used to specify the technically permissible maximum unbraked towable mass of a trailer in kilograms.

Unless local legislation specifies otherwise, ISO 1176:1990, 4.18 (maximum design towed mass) applies.

5.5.8 Engine and power source types

```
EngineId                        ::= UTF8String (SIZE (1..60))
PrimeEngineType                 ::= Text60
EnginePowerSources              ::= SEQUENCE OF PowerSource     -- primary
source first
PrimePowerSource                ::= PowerSource
EngineMaxNetPower               ::= Kilowatt
EngineDisplacement              ::= Millilitre
RatedEngineSpeed                ::= PerMinute
PowerWeightRatio                ::= KilowattPerKilogram
MaxSpeed                        ::= KilometrePerHour
FuelTanksCapacity               ::= Litre


PowerSource ::= INTEGER {
        notPowered              (0),
        humanPowered            (1),
        animalPowered           (2),
        unleadedPetrol          (3),
        leadedPetrol            (4),
        diesel                  (5),
        bioDiesel               (6),
        alcohol                 (7),
        otherFuel               (8),
        lpg                     (9),
        hydrogen                (10),
        externalElectricPower   (11),
        battery                 (12),
        sun                     (13),
        other                   (14)
        } (0..255)
```

Unless specified otherwise, the definition and assignment of the values for the types defined in this subclause are at the discretion of the registration authority and/or local legislation (see 5.5.1).

The EngineId type shall be used to specify the engine identification number.

The PrimeEngineType type shall be used to specify type of the prime engine of a vehicle as assigned by the manufacturer.

The EnginePowerSources type shall be used to specify the power source(s) of a vehicle (see below). In case of a hybrid vehicle, the primary power source is mentioned first.

The PrimePowerSource type shall be used to specify the power source of the prime engine.

The EngineMaxNetPower type shall be used to specify the nominal (maximum net) power of the engine in kW.

The EngineDisplacement type shall be used to specify the displacement of the engine in millilitres (cm−3).

The RatedEngineSpeed type shall be used to specify the rated speed in min−1.

The RowerWeightRatio type shall be used to specify the power/weight ratio of a vehicle in kW/kg.

The MaxSpeed type shall be used to specify the maximum speed of a vehicle in km/h.

The FuelTankCapacity type shall be used to specify the capacity of the fuel tank(s) in litres.

The PowerSource type shall be used to specify a power source and shall have one of the values: notPowered, humanPowered, animalPowered, unleadedPetrol, leadedPetrol, diesel, bioDiesel, alcohol, otherFuel, lpg, hydrogen, externalElectricPower, battery, sun, or other.

5.5.9 Environmental types

```
StationarySoundLevel              ::= DB
EngineSpeed                       ::= PerMinute
DriveBySoundLevel                 ::= DB
EmissionCO                        ::= MgpkmOrMgpkwh
EmissionHC                        ::= MgpkmOrMgpkwh
EmissionNOx                       ::= MgpkmOrMgpkw
EmissionHCandNOx                  ::= MilligramPerKilometre
ParticulatesForDiesel             ::= MicrogrampkmOrMicrogrampkwh
CorrectedAbsorptionCoefficient  ::= PerKmMetre
EmissionCO2                       ::= GramPerKilometre
CombinedFuelConsumption           ::= MillilitrePerKilometre
EnvironmentalCategory             ::= Text60


EuroType ::= INTEGER{
         non-euro                 (0),
         euro-1                   (1),
         euro-2                   (2),
         euro-3                   (3),
         euro-4                   (4),
         euro-5                   (5),
         euro-6                   (6)
          (0..255)}
```

Unless specified otherwise, the definition and assignment of the values for the types defined in this subclause are at the discretion of the registration authority and/or local legislation (see 5.5.1).

The StationarySoundLevel type shall be used to specify the stationary sound level of the vehicle in dB(A).

The EngineSpeed type shall be used to specify the engine speed in min–1.

The DriveBySoundLevel type shall be used to specify the drive-by sound level of the vehicle in dB(A).

The EmissionCO type shall be used to specify the CO exhaust emission in mg/km or mg/kWh.

The EmissionHC type shall be used to specify the HC exhaust emission in mg/km or mg/kWh.

The EmissionNOx type shall be used to specify the NOx exhaust emission in mg/km or mg/kWh.

The EmissionHCandNOx type shall be used to specify the HC plus COx exhaust emission in mg/km.

The ParticulatesForDiesel type shall be used to specify the exhaust emission of particulates for diesel in µg/km or µg/kWh.

The CorrectedAbsorptionCoefficient type shall be used to specify the corrected absorption coefficient for diesel in km-1.

The EmissionCO2 type shall be used to specify the CO2 exhaust emission in g/km.

The CombinedFuelConsumption type shall be used to specify the combined fuel consumption of a vehicle in millilitre/km (=litre/1 000km).

The EnvironmentalCategory type shall be used to be of type EnvironmentalCategory and shall specify the environmental category of a vehicle.

The EuroType type shall be used to specify for light duty vehicles the environmental category Euro 1…6 according to EU directive 70/220/EEC (repealed with effect from 2 January 20013) and EU Regulation 715/2007 and for heavy duty vehicles the category Euro I…VI according to EU Directive 2005/55 (repealed with effect from 31 December 2013) and EU Regulation 595/2009.

5.5.10 Official test data type

The type OfficalVehicleTestData shall be used for the results of an official roadworthiness test of the vehicle, e.g. a periodic motor vehicle test or an intermediate test and is defined as follows.

```
OfficialVehicleTestData ::= SEQUENCE {
          date                          DATE OPTIONAL,
          location                      NameAndAddress OPTIONAL,
          odometerValue                 Kilometre OPTIONAL,
          emissionCO                    MgpkmOrMgpkwh OPTIONAL,
          emissionHC                    MgpkmOrMgpkwh OPTIONAL,
          remarks                       Text60 OPTIONAL,
          …
          }
```

The date component, if present, shall specify the date of a vehicle test.

The location component, if present, shall be of type NameAndAddress and shall specify the name and address of the vehicle test facility.

The odometerValue component, if present, shall specify the reading of the odometer in kilometres.

The emissionCOx component, if present, shall specify the CO$_X$ exhaust emission in mg/km or mg/kWh.

The emissionHC component, if present, shall specify the HC exhaust emission in mg/km or mg/kWh.

The remarks component, if present, shall specify any additional remarks.

5.5.11 Types used for EFC

5.5.11.1 General

Vehicle data is exchanged also for electronic fee collection (EFC) purposes.

5.5.11 contains a number of additional vehicle data types used for EFC. These types may be used for other applications or may be supported by a registration authority as well.

NOTE 1 A reason for use may be a more compact encoding and/or the compatibility with ISO 14906:2011.

The units of data to be exchanged are the values of the EFC attributes listed in ISO 14906:2011, Table 36. The types defined in 5.5.11 are equivalent to the types listed in this table.

NOTE 2 Not all EFC vehicle data attributes can be used for ERI. Some EFC attributed contain dynamic date, i.e. data that is not suitable for a vehicle registration certificate. Some EFC attributes contain values for two registered vehicles: a towing one and a trailer. Therefore, it requires both vehicle registration certificates and, consequently, two EriData types to represent these two values. Some attributes contain data that is specific for a third party called ”service provider” (for details, see Annex C).

NOTE 3 The ASN.1 specification of enumerated types is simplified by removing the numbers from named numbers. With PER encoding, these numbers are only used to order the enumeration before assigning an index number for each enumerated item. Then, the index number is used for the encoding (see ISO/IEC 8825-2:2008, Clause 14 for details).

5.5.11.2 Vehicle dimensions type

The VehicleDimensions type shall be used to specify the vehicle’s overall length, height, and width in decimetres and rounded to the next decimetre.

The VehicleDimensions type is defined as follows.

```
VehicleDimensions ::= SEQUENCE {
         vehicleLengthOverall        Int1,
         vehicleHeigthOverall        Int1,
         vehicleWidthOverall         Int1
         }
```

The vehicleLengthOverall component shall specify the nominal maximum overall length of the vehicle in accordance with ISO 612.

The vehicleHeigthOverall component shall specify the nominal overall unladen height in accordance with ISO 612.

The vehicleWidthOverall component shall specify the nominal overall width in accordance with ISO 612.

NOTE The VehicleDimensions type corresponds to the combined use of the VehicleLength, the VehicleHeight, and the VehicleWidth types with values rounded to the next decimetre.

5.5.11.3 Passenger capacity type

The VehicleCapacity type shall be used to specify both the number of seats and the number standing places.

The VehicleCapacity type is defined as follows.

```
PassengerCapacity ::= SEQUENCE{
        numberOfSeats                Int1,
        numberOfStandingPlaces       Int1
        }
```

The numberOfSeats component shall specify the number of seats of the vehicle including the driver’s seat.

The numberOfStandingPlaces component shall specify the number of standing places of the vehicle.

NOTE The VehicleDimensions type corresponds to the combined use of the NumberOfSeats type and the NumberOfStandingPlaces type with the maximum values restricted to 255.

5.5.11.4 Vehicle weight limits type

The VehicleWeightLimits type shall be used to specify the maximum weight for the laden vehicle, the vehicle train, and the unladen vehicle weight.

The VehicleWeightLimits type is defined as follows.

```
VehicleWeightLimits ::= SEQUENCE {
        vehicleMaxLadenWeight          Int2,
        vehicleTrainMaximumWeight      Int2,
        vehicleWeightUnladen           Int2
        }
```

The vehicleMaxLadenWeight component shall specify the maximum permissible total weight including payload in accordance with ISO 1176.

The vehicleTrainMaximumWeight component shall specify the maximum authorized mass of vehicle combination in accordance with ISO 1176 Code ISO-M19.

The vehicleWeightUnladen component shall specify the nominal unladen weight in accordance with ISO 1176.

All weight values are expressed in units of 10 kg and rounded down to the next 10 kg unit.

NOTE The type of the vehicleMaxLadenWeight, vehicleTrainMaximumWeight, and vehicleWeightUnladen corresponds the combined use of the MaxAuthorizedLadenMass, MaxAuthorizedTrainMass, and UnladenWeight types rounded down to 10 kilogram units and with values limited to 655 350 kilogram.

5.5.11.5 Axle weight limits type

The AxleWeightLimits type shall be used to specify the maximum laden weight on the vehicle's axles.

The AxleWeightLimits type is defined as follows.

```
AxleWeightLimits ::= SEQUENCE{
        maxLadenweightOnAxle1      Int2,
        maxLadenweightOnAxle2      Int2,
        maxLadenweightOnAxle3      Int2,
        maxLadenweightOnAxle4      Int2,
        maxLadenweightOnAxle5      Int2
        }
```

The maxLadenweightOnAxle1, maxLadenweightOnAxle2, maxLadenweightOnAxle3, maxLadenweightOnAxle4, and maxLadenweightOnAxle5 components shall be used to specify the technically permissible maximum laden weight on repectively the first, second, third, fourth, and fifth axle in 10 kg units and rounded down to the next 10 kg unit.

NOTE The AxleWeightLimits type differs from the AuthorizedAxleLadenMass. First, the authorized maximum may be less than the technically permissible maximum. Second, the value is rounded down to units of 10 kg and third, the number axle is limited to five.

5.5.11.6 Vehicle specific characteristics type

5.5.11.6.1 Vehicle specific characteristics type

The VehicleSpecificCharacteristics type shall be used to specify the vehicle's environmental characteristics, engine characteristics, descriptive characteristics, and future characteristics.

The VehicleSpecificCharacteristics type is defined as follows.

```
VehicleSpecificCharacteristics ::= SEQUENCE {
         environmentalCharacteristics    EnvironmentalCharacteristics,
         engineCharacteristics           EngineCharacteristics,
         descriptiveCharacteristics      DescriptiveCharacteristics,
         futureCharacteristics           FutureCharacteristics
                  }
```

The environmentalCharacteristics component shall specify the environmental characteristics.

The engineCharacteristics component shall specify the engine characteristics

The descriptiveCharacteristics component shall specify the descriptive characteristics

The futureCharacteristics component shall specify the vehicle's future characteristics.

5.5.11.6.2 Environmental characteristics type

The EnvironmentalCharacteristics type shall be used to specify the vehicles environmental characteristics.

The EnvironmentalCharacteristics type is defined as follows.

```
EnvironmentalCharacteristics::= SEQUENCE {
        euroValue                       EuroValue,
        copValue                        CopValue
        }


EuroValue::= INTEGER{
        noEntry                         (0),
        euro-1                          (1),
        euro-2                          (2),
        euro-3                          (3),
        euro-4                          (4),
        euro-5                          (5),
        euro-6                          (6),
        } (0..15) -- 4 bits, EURO-Clases as defined in EC directive
88/77/EEC, annex 1
        -- and in 91/542/EEC, 96/1/EC, 1999/96/EC, 2001/27/EC


CopValue ::= INTEGER                    {
         noEntry (0),
         co2class1 (1),                 -- below 101 g/km
         co2class2 (2),                 -- 101 to 120 g/km
         co2class3 (3),                 -- 121 to 140 g/km
         co2class4 (4),                 -- 141 to 160 g/km
         co2class5 (5),                 -- 161 to 200 g/km
         co2class6 (6),                 -- 201 to 250 g/km
         co2class (7)7                  -- above 250 g/km
         } (0..15) -- 4 bits, reserved for carbon dioxide pollution
values as defined in
             -- EC directive 2003/127/EC
```

The euroValue component shall specify the vehicle’s environmental category in accordance with EU directives 88/77/EEC, Annex 1, 91/542/EEC, 96/1/EC, 1999/96/EC, and 2001/27/EC.

NOTE 1 This specification is in accordance with ISO 14906:2011. However, see NOTES 2 and 3.

NOTE 2 All these directives are repealed in Annex IX part A of the newer EU directive 2005/55/EC which, in turn, is repealed with effect from 31 December 2013 (see EU Regulation 715/2007). EU Regulation 715/2007 stipulates also that references made to the repealed Directives shall be construed as being made to this regulation

NOTE 3 For light duty vehicles, EU Regulation 715/2007 applies.

The copValue component shall specify vehicle’s emission of CO_2 using the values of the CopValue type.

NOTE 4 The EnvironmentalCharacteristics type corresponds to the combined use of the EuroType and EmissionCO2 types with the EmissionCO2 values expressed in g/km and divided into seven intervals/classes.

5.5.11.6.3 Engine characteristics type

The EngineCharacteristics type shall be used specify the vehicle’s power source.

The EngineCharacteristics type is defined as follows.

```
EngineCharacteristics::= INTEGER {
        noEntry                         (0),
        noEngine                        (1),
        petrolUnleaded                  (2),
        petrolLeaded                    (3),
        diesel                          (4),
        lPG                             (5),
        battery                         (6),
        solar                           (7)
        -- (8-255) are reserved for future CEN use
        } (0..255)
```

NOTE The EngineCharacteristics type corresponds with the PrimePowerSource type, but has a more limited set of values.

5.5.11.6.4 Descriptive characteristics type

The DescriptiveCharacteristics type shall be used to specify the vehicle’s descriptive characteristics.

The DescriptiveCharacteristics type is defined as follows.

```
DescriptiveCharacteristics ::= INTEGER {
          noEntry                         (0),
          vehicleShape1                   (1),
          vehicleShape2                   (2),
          vehicleShape3                   (3),
          vehicleShape4                   (4),
          vehicleShape5                     (5),
          vehicleShape6                     (6),
          vehicleShape7                     (7),
          vehicleShape8                     (8),
          vehicleShape9                     (9),
          vehicleShape10                    (10),
          vehicleShape11                    (11),
          vehicleShape12                    (12),
          vehicleShape13                    (13),
          vehicleShape14                    (14),
          vehicleShape15                    (15),
          vehicleShape16                    (16),
          vehicleShape17                    (17),
          vehicleShape18                    (18),
          vehicleShape19                    (19),
          vehicleShape20                    (20),
          vehicleShape21                    (21),
          vehicleShape22                    (22),
          vehicleShape23                    (23),
          vehicleShape24                    (24),
          vehicleShape25                    (25),
          vehicleShape26                    (26),
          vehicleShape27                    (27),
          vehicleShape28                    (28),
          vehicleShape29                    (29),
          vehicleShape30                    (30),
          vehicleShape31                    (31),
          vehicleShape32                    (32),
          vehicleShape33                    (33),
          vehicleShape34                    (34),
          vehicleShape35                    (35),
          vehicleShape36                    (36),
          vehicleShape37                    (37),
          vehicleShape38                    (38),
          vehicleShape39                    (39),
          vehicleShape40                    (40),
          vehicleShape41                    (41),
          vehicleShape42                    (42),
          vehicleShape43                    (43),
          vehicleShape44                    (44),
          vehicleShape45                    (45),
          vehicleShape46                    (46),
          vehicleShape47                    (47),
          vehicleShape48                    (48),
          vehicleShape49                    (49),
          vehicleShape50                    (50),
       -- (1..50) are reserved for future CEN ISO use
       -- (51..255) are reserved for private use
       } (0..255)
```

NOTE The DescriptiveCharacteristics type does not correspond to any other ERI type.

5.5.11.6.5 Future characteristics type

The FutureCharacteristics type shall be used to specify the vehicle's suspension, if applicable.

```
FutureCharacteristics ::= INTEGER {
        noEntry                           (0),
        airSuspension                     (1)
        -- (2..255) are reserved for future CEN use
        } (0..255)
```

The airSuspension value shall be used to specify the presence of air suspension.

The noEntry value shall be used in case the vehicle is not equipped with air suspension or in case no suspension data is to be provided.

5.5.11.7 Trailer characteristics type

The TrailerCharacteristics type shall be used to specify for a trailer its type, the number of axles, the maximum permissible total weight, and the unladen weight.

The TrailerCharacteristics type is defined as follows.

```
TrailerCharacteristics ::= SEQUENCE {
         trailerDetails                    TrailerDetails,
         trailerMaxLadenWeight             Int2,
         trailerWeightUnladen              Int2
         }


TrailerDetails::= SEQUENCE {
         trailerType                       TrailerType,
         trailerAxles                      TrailerAxles
         }


TrailerType ::= INTEGER{
         notPresent (0),                   -- trailer not attached or only
one trailer attached, see
                                           -- VehicleAxlesNumber for more
information
         trailer (1),                      -- also known as pull-bar
trailer
         semitrailer (2)                   -- also known as articulate trailer
         -- (3..31) reserved for future CEN/ISO use
         } (0..31)
TrailerAxles ::= INTEGER (0..7)                             -- number of axles of the
trailer when available
```

The trailerDetails component shall specify the trailer type and its number of axles.

The trailerType component shall specify the type of the trailer. Defined values are notPresent, trailer (also known as pull-bar trailer), and semi-trailer (a trailer without a front axle).

When used as part of the ERI data (i.e. as part of a trailer registration certificate data), the value notPresent shall not be used.

The trailerAxles component shall specify the number of axles of the trailer.

The trailerMaxLadenWeight component shall specify the maximum permissible total weight of the trailer including payload in accordance with ISO 1176, in 10 kg units, and rounded down to the next 10 kg unit.

The trailerWeightUnladen component shall specify the nominal unladen weight of the trailer which shall be according to ISO 1176, in 10 kg units, and rounded down to the next 10 kg unit.

NOTE The TrailerCharacteristics type corresponds to the combined use of the following types for a trailer: NumberOfAxles, MaxAuthorizedLadenMass, and UnladenWeight.

5.5.11.8 Engine type

The Engine type shall be used to specify the engine's capacity and power.

The Engine type is defined as follows.

```
Engine ::= SEQUENCE{
         engineCapacity              Int2,
         enginePower                 Int2
         }
```

The engineCapacity component shall specify the capacity of the vehicle's engine in millilitres.

The engineCapacity component shall specify the maximum net power of the vehicle's engine, in kilowatt.

NOTE The Engine type corresponds to the combined use of the EngineDisplacement and EngineMaxNetPower types with their values limited to 65535 cm^3 and 65535 kW respectively.

5.5.11.9 Sound level type

The SoundLevel type shall be used to specify the vehicle's stationary and drive-by soundlevels.

The SoundLevel type is defined as follows.

```
SoundLevel ::= SEQUENCE{
             soundstationary               Int1,
             sounddriveby                  Int1
             }
```

The soundstationary component shall specify the stationary sound level of the vehicle according to its registration certificate in dB(A).

The sounddriveby component shall specify the drive-by sound of the vehicle according to its registration certificate in dB(A).

NOTE The SoundLevel type corresponds to the combined use of the StationarySoundLevel and DriveBySoundLevel types with their values limited to 255 dB.

5.5.11.10 CO$_2$ emission value type

The CO2EmissionValue type shall be used to specify the vehicle's emission of CO$_2$.

The CO2EmissionValue type is defined as follows.

```
CO2EmissionValue ::= Int2
```

The value of the CO2EmissionValue type shall be in g/km and according to the vehicle registration certificate.

NOTE The CO2EmissionValue type corresponds to the EmissionCO2 type with its values limited to 65 535 g/km.

5.5.11.11 Exhaust emission values type

The ExhaustEmissionValues type shall be used to specify the vehicle's emission of CO, HC, NO$_X$, and HC plus NO$_X$.

The ExhaustEmissionValues type is defined as follows.

```
ExhaustEmissionValues ::= SEQUENCE {
        unitType                       UnitType,
        emissionCO                     INTEGER (0..32766),
        emissionHC                     Int2,
        emissionNOX                    Int2,
        emissionHCNOX                  Int2
        }


UnitType ::= ENUMERATED {
        mg-km,
        mg-kWh
        }
```

The unitType component shall specify the unit for the emissionCO, emissionHC, emissionNOX, emissionHCNOX components. The defined values are mg/km and mg/kWh.

The emissionCO component shall specify the CO exhaust emission in the units specified by the unitType component and according to the vehicle registration certificate.

The emissionHC component shall specify the HC exhaust emission in the units specified by the unitType component and according to the vehicle registration certificate.

The emissionNOX component shall specify the NO_X exhaust emission in the units specified by the unitType component and according to the vehicle registration certificate.

The emissionHCNOX component shall specify the HC plus NO_X exhaust emission in the units specified by the unitType component and according to the vehicle registration certificate.

The value of the emissionHCNOX component should be the sum of the emissionHC and emissionHCNOX components values.

NOTE The ExhaustEmissionValues corresponds to the combined use of the EmissionCO, EmissionHC, EmissionNOx, and EmissionHCandNOx types with their values limited to 32 766 (for the CO emission) or to 65 535 (for the other exhaust emissions).

5.5.11.12 Diesel emission values type

The DieselEmissionValues type shall be used to specify the emission of diesel particles and the absorption coefficient.

```
DieselEmissionValues ::= SEQUENCE {
        unitType                        UnitType,
        particulate                     INTEGER (0.. 32767),
        absorptionCoeff                 Int2
        }


UnitType ::= ENUMERATED {
        mg-km,
        mg-kWh
        }
```

The unitType component shall specify the unit for the particulate component. The defined values are mg/km and mg/kWh.

The particulate component shall specify the particulates for diesel in the units specified by the unitType component and according to vehicle registration certificate.

The absorptionCoeff component shall specify the Corrected absorption coefficient for diesel, according to vehicle registration documents, in 10^{-3} m^{-1} (= per km).

NOTE The DieselEmissionValues type corresponds to the combined use of the ParticulatesForDiesel and CorrectedAbsorptionCoefficient types with the unit of the particles changed from µg/km or µg/kWh to mg/km or mg/kWh, with value of the particulates limited to 32 766, with the unit of the corrected absorption coefficient changed from m^{-1} to km^{-1} and its values limited to 65 535.

5.5.12 Other types

5.5.12.1 Registration authority specific data

```
RaSpecificData ::= Text60
```

The raSpecificVehicleData component, if present, may contain any additional registration authority-specific data.

5.5.12.2 Measurement units

Measurement unit types to describe a vehicle are defined as follows.

```
MgpkmOrMgpkwh ::= CHOICE {
       milligramPerKilometre        MilligramPerKilometre,
       milligramPerKilowattHour     MilligramPerKilowattHour
       }
```

```
MicrogrampkmOrMicrogrampkwh ::= CHOICE {
       microgramPerKilometre        MicrogramPerKilometre,
       microgramPerKilowattHour     MicrogramPerKilowatthour
       }
Kilogram ::=                        INTEGER (0..MAX)
Kilometre ::=                       INTEGER (0..MAX)
Millimetre ::=                      INTEGER (0..MAX)
PerKm ::=                           INTEGER (0..MAX)
Litre ::=                           INTEGER (0..MAX)
Millilitre ::=                      INTEGER (0..MAX)
Kilowatt ::=                        INTEGER (0..MAX)
KilometrePerHour ::=                INTEGER (0..MAX)
PerMinute ::=                       INTEGER (0..MAX)
DB ::=                              INTEGER (0..MAX)
GramPerKilometre ::=                INTEGER (0..MAX)
MilligramPerKilometre ::=           INTEGER (0..MAX)
MicrogramPerKilometre ::=           INTEGER (0..MAX)
MilligramPerKilowatthour ::=        INTEGER (0..MAX)
MicrogramPerKilowatthour ::=        INTEGER (0..MAX)
KilowattPerKilogram ::=             INTEGER (0..MAX)
MillilitrePerKilometre ::=          INTEGER (0..MAX)       -- = litre per 1 000 km
```

The MgpkmOrMgpkwh type shall be used to specify a value in milligrams per kilometre or in milligrams per kilowatthour. The type shall only be used when a user may choose between either of these values.

The MicrogrampkmOrMicrogrampkwh type shall be used to specify a value in micrograms per kilometre or in micorgrams per kilowatthour. The type shall only be used when a user may choose between either of these values.

5.5.12.3 Entity identifier

The type EntityId is used for a global identification of entities and is defined as follows.

```
EntityId ::= OBJECT IDENTIFIER
```

5.5.12.4 Name and address

The NameAndAddress type shall be used to specify a name and address and is defined as follows.

```
NameAndAddress ::= SEQUENCE {
       name                       Text60,
       otherNamesOrInitials       Text60 OPTIONAL,
       address                    Text60
       }
```

The name component shall specify the surname or business name of a natural or artificial person.

The otherNamesOrInitials component, if present, shall specify other name(s) or initial(s). The otherNamesOrInitials component shall be present, where appropriate, and absent if not.

The address component shall specify the address of the natural or artificial person.

NOTE The NameAndAddress type is compliant with EU 2003/127 which also specifies a maximum length of 60 characters.

5.5.12.5 Text

The Text60 type is used for textual ERI data and defined as follows.

```
Text60 ::= UTF8String (SIZE (1..60))
```

The value is a UTF8 string, according to ISO 8824, not exceeding 60 characters.

5.6 Attributes

5.6.1 Introduction

An attribute is a type with an associated identifier.

NOTE 1 Attributes are often used to retrieve or to set a number of parameters of some type.

EXAMPLE In order to get the value of some parameters, one sends their identifiers to receive a list with the values of these parameters together with their identifiers.

NOTE 2 As several parameters can be of the same type, i.e. share the same set of values, the same type may be used more than one attribute.

Using ASN.1 terms, an attribute is an information object from an information object class called ATTRIBUTE.

The attributes can be grouped into sets. This allows for a precise control over the attributes to be used (or not to be used) for a particular application.

This part of ISO 24534 defines some useful sets and some useful types based of these sets.

5.6.2 Useful types

Annex A defines the following useful types for use in other ERI applications:

a) EriAttributeIdList and EriAttributeList;

b) EriBasicAttributeIdList and EriBasicAttributeList;

c) EriEuAttributeIdList and EriEuAttributeList;

d) EriEfcAttributeIdList and EriEfcAttributeList.

EriAttributeIdList and EriAttributeList are defined as follows.

```
EriAttributeIdList ::= SEQUENCE OF EriAttributeId
EriAttrubuteList ::= SEQUENCE OF EriAttribute
EriAttributeId ::= ATTRIBUTE.&id ({EriAttributes })
EriAttribute ::= SEQUENCE {
          id                      ATTRIBUTE.&id ({EriAttributes }),
          value                   ATTRIBUTE.&Type ({EriAttributes } {@.id})
          }
```

The EriAttributeIdList type is defined as a sequence of EriAttributeId values.

The EriAttributeList type is defined as a sequence of EriAttribute values.

The EriAttributeId type is defined as an id of an attribute from the set of attributes EriAttributes.

The EriAttribute type is defined as an id and a value of the identified type of an attribute from the set of attributes EriAttributes.

The EriBasicAttributeIdList and EriBasicAttributeList type are defined in the same way as EriAttributeIdList type and the EriAttributeList type, but with the attributes from the attribute set EriBasicAttributes.

EriEuAttributeIdList and EriEuAttributeList are defined in the same way as EriAttributeIdList type and the EriAttributeList type, but with the attributes from the attribute set EriEuAttributes.

EriEfcAttributeIdList and EriEfcAttributeList are defined in the same way as EriAttributeIdList type and the EriAttributeList type, but with the attributes from the attribute set EriEfcAttributes.

NOTE 1 See Annex A for the details of these definitions.

NOTE 2 By using ANS.1 notation above, it can be guaranteed with ASN.1 tools that an attribute is an attribute from a given set of attributes and that for each id the correct type is used.

5.6.3 Useful sets of attributes

Annex A defines the following useful sets of attributes for use in other ERI applications:

a) EriAttributes;

b) EriBasicAttributes;

c) EriEuAttributes;

d) EriEfcAttributes.

EriAttributes is the set of all attributes defined in Annex A.

EriBasicAttributes is the set of attributes in EriAttributes but not in EriEfcAttributes.

EriEuAttributes is the set of all the attributes used in EU registration certificates, i.e. for the ERI types listed in Table B.1.

EriEfcAttributes is the set of attributes as used for EFC purposes.

5.6.4 Information object class ATTRIBUTES

The information object class ATTRIBUTE is defined as follows.

```
ATTRIBUTE ::= CLASS {
        &Type,
        &id                         Int2 UNIQUE
        }
WITH SYNTAX {&Type IDENTIFIED BY &id}
```

The &Type field shall be used for the identified type.

The &id field shall be used for a unique identifier of type Int2, i.e. with a value between 0 and 65 535 (inclusive).

The ”WITH SYNTAX” provides for easy readable attribute definitions (see 5.6.5).

5.6.5 Attribute definitions

In Annex A, attributes are defined for the vehicle identifiers and for the alternatives of the AdditionalEriRegistrationData type.

The attributes vin and vehicleId are defined as follows.

```
vin ATTRIBUTE ::=
         {VIN IDENTIFIED BY vinId}
vinId INTEGER ::=     15
vehicleId ATTRIBUTE ::=
         {VehicleId IDENTIFIED BY vehicleIdId}
vehicleIdId INTEGER ::= 129
```

The vin attribute of of type VIN and is identified by the value 15.

The vehicleId attribute is of type VehicleID and is identified by the value 129.

Other attributes are defined in the same way.

The use of identifiers is consistent with ISO 14906:2011. Values between 0 and 127 are only used for types used for EFC and, if used, also with the same value.

NOTE This allows for future combinations (unions) of sets of ERI attributes and EFC attributes without conflicts, i.e. while preserving the uniqueness of the identifiers.

5.7 Encoding

When exchanged between an ERT and an ERI reader or writer, ERI data shall be encoded according to the canonical packed encoding rules (CANONICAL-PER) UNALIGNED variant as defined in ISO 8825-2.

Annex A
(normative)

ASN.1 Modules

A.1 General

This Annex contains the following ASN.1 modules:

- the ElectronicRegistrationIdentificationVehicleDataModule in A.2;
- the ElectronicRegistrationIdentificationEfcVehicleDataModule in A.3;
- the ElectronicRegistrationIdentificationVehicleDataAttributesModule in A.4.

In case the ASN.1 specifications given in this Annex are not compliant with illustrations or specifications provided elsewhere in this part of ISO 24534, the specifications given in this Annex shall prevail.

The ASN.1 modules contained in this Annex will be published on http://standards.iso.org/iso/24534/-3.

A.2 Module ElectronicRegistrationIdentificationVehicleDataModule

In all definitions

```
fill    BIT STRING (SIZE(x))
```

each bit of the BIT STRING shall be set to the value ‘0’b.

```
ElectronicRegistrationIdentificationVehicleDataModule {iso(1) standard(0) iso24534 (24534)
vehicleData (1) version1 (1)}


-- This version is upwards compatible with the previous version 0 in the sense that:
-- any receiver can successfully receive data of either version
-- But:
--  a receiver using version 0 will not be able to decode the version 1 extensions
-- The units of the environmental values should be interpreted differently
-- See Annex I of ISO 8824-1:2008 for details


DEFINITIONS AUTOMATIC TAGS ::= BEGIN


-- Electronic Registration Identification (ERI)- Vehicle Data


-- EXPORTS  everything;


IMPORTS
CS4, CS5 FROM AVIAEINumberingAndDataStructures {iso(1) standard(0) iso14816(14816)
asnm1(1) version1(1)}


VehicleDimensions, PassengerCapacity, VehicleWeightLimits, AxleWeightLimits,
VehicleSpecificCharacteristics, TrailerCharacteristics, Engine,
CO2EmissionValue, SoundLevel, ExhaustEmissionValues, DieselEmissionValues FROM
ElectronicRegistrationIdentificationEfcVehicleDataModule {iso(1) standard(0) iso24534
(24534) efcVehicleData (3) version1 (1)}
;
```

```
-- Vehicle Identifier

VehicleId ::= CHOICE {
       vin                    VIN, -- preferred choice / 17 characters
       raSpecificVehicleId   RaSpecificVehicleId,
       ...
       }

VIN ::= CS5

RaSpecificVehicleId ::= SEQUENCE {
       wmi                    UTF8String (SIZE(3)),
       nonIsoStandardId       UTF8String (SIZE (1..20))
       }

-- ERI data

EriData ::= SEQUENCE {
       vehicleId              VehicleId,
       additionalEriData      AdditionalEriData OPTIONAL
       }

AdditionalEriData ::= CHOICE {
       additionalEriRegistrationData AdditionalEriRegistrationData,   -- preferred choice
       ...,
       raSpecificAdditionalEriData   OCTET STRING (SIZE (0..1024))
       -- only to be used if AdditionalEriRegistrationData is not supported
       }

-- Additional ERI registration data

AdditionalEriRegistrationData ::=  SEQUENCE {
       -- Administrative data
       registrationAuthority   RegistrationAuthority OPTIONAL,
       vehicleIdStatus         VehicleIdStatus OPTIONAL,
       dateOfFirstRegistration DateOfFirstRegistration OPTIONAL,
       dateOfRegistration      DateOfRegistration OPTIONAL,
       validThru               ValidThru OPTIONAL,
       chassisNumber           ChassisNumber (SIZE (1..23)) OPTIONAL,
       registrationNumber      RegistrationNumber OPTIONAL,

       -- Vehicle type
       vehicleMake             VehicleMake OPTIONAL,
       vehicleType  VehicleType OPTIONAL,
       vehicleTypeStatus       VehicleTypeStatus OPTIONAL,
       commercialDescription   CommercialDescription OPTIONAL,
       typeApprovalNumber      TypeApprovalNumber OPTIONAL,
       vehicleCategory         VehicleCategory OPTIONAL,
       vehicleTaxCategory      VehicleTaxCategory OPTIONAL,
       euVehicleCategoryCode   EuVehicleCategoryCode OPTIONAL,
       raSpecificVehicleClass1 RaSpecificVehicleClass1 OPTIONAL,
       raSpecificVehicleClass2 RaSpecificVehicleClass2 OPTIONAL,
       raSpecificVehicleClass3 RaSpecificVehicleClass3 OPTIONAL,
       vehicleUse              VehicleUse OPTIONAL,
       privateUse              PrivateUse OPTIONAL,
       colour                  VehicleColour OPTIONAL,

       -- Vehicle shape
       length                  VehicleLength OPTIONAL,
       width                   VehicleWidth OPTIONAL,
       height                  VehicleHeight OPTIONAL,
       wheelbase               Wheelbase OPTIONAL,
       bodyShape               VehicleBodyShape OPTIONAL,
       euBodyWorkType          EuBodyWorkType OPTIONAL,
       iso3833VehicleType      Iso3833VehicleType OPTIONAL,


       -- Vehicle number of passengers, axles, and mass
       numberOfSeats           NumberOfSeats OPTIONAL,   -- including the driver seat
       numberOfStandingPlaces  NumberOfStandingPlaces OPTIONAL,
       maxNumberOfPassengers   MaxNumberOfPassengers OPTIONAL,   -- including the driver
       unladenWeight           UnladenWeight OPTIONAL,
       maxDesignLadenMass      MaxDesignLadenMass OPTIONAL,
       maxAuthorizedLadenMass  MaxAuthorizedLadenMass OPTIONAL,
       maxAuthorizedTrainMass  MaxAuthorizedTrainMass OPTIONAL,
       maxAuthorizedPayload    MaxAuthorizedPayload OPTIONAL,


       numberOfAxles           NumberOfAxles OPTIONAL,
       authorizedAxleLadenMass AuthorizedAxleLadenMass OPTIONAL, -- from front to rear
axle
       maxTowableMassBrakedTrailer MaxTowableMassBrakedTrailer OPTIONAL,
       maxTowableMassUnbrakedTrailer MaxTowableMassUnbrakedTrailer OPTIONAL,


       -- Vehicle engine and power source
       engineId                EngineId (SIZE (1..60)) OPTIONAL,
       primeEngineType         PrimeEngineType OPTIONAL,
       enginePowerSources      EnginePowerSources OPTIONAL,
       primePowerSource        PrimePowerSource OPTIONAL,
       engineMaxNetPower       EngineMaxNetPower OPTIONAL,
       engineDisplacement      EngineDisplacement OPTIONAL,
       ratedEngineSpeed        RatedEngineSpeed OPTIONAL,
       powerWeightRatio        PowerWeightRatio OPTIONAL,
       maxSpeed                MaxSpeed OPTIONAL,
       fuelTanksCapacity       FuelTanksCapacity OPTIONAL,


       -- Environmental characteristics
       stationarySoundLevel    StationarySoundLevel OPTIONAL,
       engineSpeed             EngineSpeed OPTIONAL,
       driveBySoundLevel       DriveBySoundLevel OPTIONAL,
       emissionCO              DriveBySoundLevel OPTIONAL,
       emissionHC              EmissionHC OPTIONAL,
       emissionNOx             EmissionNOx OPTIONAL,
       emissionHCandNOx        EmissionHCandNOx OPTIONAL,
       particulatesForDiesel   ParticulatesForDiesel OPTIONAL,
       correctedAbsorptionCoefficient CorrectedAbsorptionCoefficient OPTIONAL,
       emissionCO2             EmissionCO2 OPTIONAL,
       combinedFuelConsumption CombinedFuelConsumption OPTIONAL,
       environmentalCategory   EnvironmentalCategory OPTIONAL,
       euroType     EuroType OPTIONAL,


       -- Others
       lastOfficialTestData   OfficialVehicleTestData OPTIONAL,
       ...,
       raSpecificData         RaSpecificData OPTIONAL,
       axlesPerAxleGroup       AxlesPerAxleGroup OPTIONAL,
                               -- from front to rear axle group
       authorizedAxleGroupLadenMass AuthorizedAxleGroupsLadenMass OPTIONAL,
                               -- from front to rear axle group


       -- ERI data used for EFC (types imported from
ElectronicRegistrationIdentificationEfcVehicleData
       efcVehicleDimensions    VehicleDimensions,
       efcPassengerCapacity    PassengerCapacity,
       efcVehicleWeightLimits  VehicleWeightLimits,
       efcAxleWeightLimits     AxleWeightLimits,
       efcVehicleSpecificCharacteristics
       VehicleSpecificCharacteristics,
       efcTrailerCharacteristics TrailerCharacteristics,
       efcEngine               Engine,
       efcSoundLevel           SoundLevel,
       efcCO2EmissionValue     CO2EmissionValue,
       efcExhaustEmissionValues                    ExhaustEmissionValues,
       efcDieselEmissionValues DieselEmissionValues
       }


-- Administrative data
RegistrationAuthority    ::= EntityId
VehicleIdStatus          ::= Text60
DateOfFirstRegistration  ::= DATE
DateOfRegistration       ::= DATE
ValidThru                ::= DATE
ChassisNumber            ::= UTF8String (SIZE (1..23)) -- incl. 3 WMI characters
RegistrationNumber  ::= CS4


-- Vehicle type
VehicleMake              ::= Text60
VehicleType              ::= Text60
VehicleTypeStatus        ::= Text60
CommercialDescription    ::= Text60
TypeApprovalNumber       ::= UTF8String (SIZE (1..60))
VehicleCategory          ::= Text60
VehicleTaxCategory       ::= Text60
RaSpecificVehicleClass1 ::= Text60
RaSpecificVehicleClass2 ::= Text60
RaSpecificVehicleClass3 ::= Text60
VehicleUse               ::= Text60
PrivateUse               ::= BOOLEAN -- False = commercial use
VehicleColour            ::= Text60


EuVehicleCategoryCode ::= CHOICE {
       euVehicleCategoryL   EuVehicleCategoryL, -- conforms to EU 2002/24 and UNECE 1999
       euVehicleCategoryM   EuVehicleCategoryM, -- conforms to EU 2001/116 and UNECE 1999
       euVehicleCategoryN   EuVehicleCategoryN, -- conforms to EU 2001/116 and UNECE 1999
       euVehicleCategoryO   EuVehicleCategoryO, -- conforms to EU 2001/116 and UNECE 1999
       euVehilcleCategoryT  NULL, -- conforms to UNECE 1999
       euVehilcleCategoryG  NULL -- conforms to EU 2001/116 and UNECE 1999
       }


EuVehicleCategoryL ::= ENUMERATED {l1, l2, l3, l4, l5, l6, l7}


EuVehicleCategoryM ::= ENUMERATED {m1, m2, m3}


EuVehicleCategoryN ::= ENUMERATED {n1, n2, n3}


EuVehicleCategoryO ::= ENUMERATED {o1, o2, o3, o4}


-- Vehicle shape
VehicleLength       ::= Millimetre
VehicleWidth        ::= Millimetre
VehicleHeight       ::= Millimetre
Wheelbase           ::= Millimetre
VehicleBodyShape    ::= Text60
EuBodyWorkType      ::= UTF8String (SIZE (2))  -- conforms to EU 2001/116


Iso3833VehicleType ::= ENUMERATED {
       passengerCar,          -- term No 3.1.1
       saloon,                -- term No 3.1.1.1 (sedan)
       convertibleSaloon,     -- term No 3.1.1.2
       pullmanSaloon,         -- term No 3.1.1.3
       stationWagon,          -- term No 3.1.1.4
       truckStationWagon,     -- term No 3.1.1.4.1
       coupe,                 -- term No 3.1.1.5 (coupé)
       convertible,           -- term No 3.1.1.6 (open tourer, roadstar, spider)
       multipurposePassengerCar, -- term No 3.1.1.7
       forwardControlPassengerCar, -- term No 3.1.1.8
       specialPassengerCar,   -- term No 3.1.1.9
       bus,                   -- term No 3.1.2
       minibus,               -- term No 3.1.2.1
       urbanBus,              -- term No 3.1.2.2
       interurbanCoach,       -- term No 3.1.2.3
       longDistanceCoach,     -- term No 3.1.2.4
       articulatedBus,        -- term No 3.1.2.5
       trolleyBus,            -- term No 3.1.2.6
       specialBus,            -- term No 3.1.2.7
       commercialVehicle,     -- term No 3.1.3
       specialCommercialVehicle, -- term No 3.1.3.1
       specialVehicle,        -- term No 3.1.4
       trailingTowingVehicle, -- term No 3.1.5 (draw-bar tractor)
       semiTrailerTowingVehicle, -- term No 3.1.6 (fifth wheel tractor)
       trailer,               -- term No 3.2.1
       busTrailer,            -- term No 3.2.1.1
       generalPurposeTrailer, -- term No 3.2.1.2
       caravan,               -- term No 3.2.1.3
       specialTrailer,        -- term No 3.2.1.4
       semiTrailer,           -- term No 3.2.2
       busSemiTrailer,        -- term No 3.2.2.1
       generalPurposeSemiTrailer, -- term No 3.2.2.2
       specialSemiTrailer,    -- term No 3.2.2.3
       roadTrain,             -- term No 3.3.1
       passengerRoadTrain,    -- term No 3.3.2
       articulatedRoadTrain,  -- term No 3.3.3
       doubleRoadTrain,       -- term No 3.3.4
       compositeRoadTrain,    -- term No 3.3.5
       specialRoadTrain,      -- term No 3.3.6
       moped,                 -- term No 3.4
       motorCycle             -- term No 3.5
       }


-- Vehicle number of passengers, axles, and mass
NumberOfSeats                 ::= INTEGER -- including the driver seat
NumberOfStandingPlaces        ::= INTEGER
MaxNumberOfPassengers         ::= INTEGER -- including the driver
UnladenWeight                 ::= Kilogram
MaxDesignLadenMass            ::= Kilogram
MaxAuthorizedLadenMass        ::= Kilogram
MaxAuthorizedTrainMass        ::= Kilogram
MaxAuthorizedPayload          ::= Kilogram


NumberOfAxles                 ::= INTEGER
AxlesPerAxleGroup             ::= SEQUENCE OF Int1 -- from front to rear axle group
AuthorizedAxleLadenMass       ::= SEQUENCE OF Kilogram -- from front to rear axle
AuthorizedAxleGroupsLadenMass ::= SEQUENCE OF Kilogram -- from front to rear axle group
MaxTowableMassBrakedTrailer   ::= Kilogram
MaxTowableMassUnbrakedTrailer ::= Kilogram


-- Vehicle engine and power source
EngineId                      ::= UTF8String (SIZE (1..60))
PrimeEngineType               ::= Text60
EnginePowerSources            ::= SEQUENCE OF PowerSource   -- primary source first
PrimePowerSource              ::= PowerSource
EngineMaxNetPower             ::= Kilowatt
EngineDisplacement            ::= Millilitre
RatedEngineSpeed              ::= PerMinute
PowerWeightRatio              ::= KilowattPerKilogram
```

```
MaxSpeed                      ::= KilometrePerHour
FuelTanksCapacity             ::= Litre


PowerSource ::= INTEGER {
       notPowered             (0),
       humanPowered           (1),
       animalPowered          (2),
       unleadedPetrol         (3),
       leadedPetrol           (4),
       diesel                 (5),
       bioDiesel              (6),
       alcohol                (7),
       otherFuel              (8),
       lpg                    (9),
       hydrogen               (10),
       externalElectricPower  (11),
       battery                (12),
       sun                     (13),
       other                  (14)
       } (0..15)


-- Environmental characteristics


StationarySoundLevel           ::= DB
EngineSpeed                    ::= PerMinute
DriveBySoundLevel              ::= DB
EmissionCO                     ::= MgpkmOrMgpkwh
EmissionHC                     ::= MgpkmOrMgpkwh
EmissionNOx                    ::= MgpkmOrMgpkwh
EmissionHCandNOx               ::= MilligramPerKilometre
ParticulatesForDiesel          ::= MicrogrampkmOrMicrogrampkwh
CorrectedAbsorptionCoefficient ::= PerKm
EmissionCO2                    ::= GramPerKilometre
CombinedFuelConsumption        ::= MillilitrePerKilometre
EnvironmentalCategory          ::= Text60


EuroType ::= INTEGER {
       non-euro                (0),
       euro-1                  (1),
       euro-2                  (2),
       euro-3                  (3),
       euro-4                  (4),
       euro-5                  (5),
       euro-6                  (6)
       } (0..15)


-- Others


OfficialVehicleTestData ::= SEQUENCE {
       date                     DATE OPTIONAL,
       location                 NameAndAddress OPTIONAL,
       odometerValue            Kilometre OPTIONAL,
       emissionCO               MgpkmOrMgpkwh OPTIONAL,
       emissionHC               MgpkmOrMgpkwh OPTIONAL,
       remarks                  Text60 OPTIONAL,
       ...
       }


RaSpecificData ::= Text60


--Measurement units
```

```
MgpkmOrMgpkwh ::= CHOICE {
       milligramPerKilometre    MilligramPerKilometre,
       milligramPerKilowattHour MilligramPerKilowatthour
       }


MicrogrampkmOrMicrogrampkwh ::= CHOICE {
       microgramPerKilometre    MicrogramPerKilometre,
       microgramPerKilowattHour                  MicrogramPerKilowatthour
       }


Kilogram ::=                    INTEGER (0..MAX)
Kilometre ::=                   INTEGER (0..MAX)
Millimetre ::=                  INTEGER (0..MAX)
PerKm ::=                       INTEGER (0..MAX)
Litre ::=                       INTEGER (0..MAX)
Millilitre::=                   INTEGER (0..MAX)
Kilowatt::=                     INTEGER (0..MAX)
KilometrePerHour ::=            INTEGER (0..MAX)
PerMinute ::=                   INTEGER (0..MAX)
DB ::=                          INTEGER (0..MAX)
GramPerKilometre ::=            INTEGER (0..MAX)
MilligramPerKilometre ::=       INTEGER (0..MAX)
MicrogramPerKilometre ::=       INTEGER (0..MAX)
MilligramPerKilowatthour ::=    INTEGER (0..MAX)
MicrogramPerKilowatthour ::=    INTEGER (0..MAX)
KilowattPerKilogram ::=         INTEGER (0..MAX)
MillilitrePerKilometre ::=      INTEGER (0..MAX) -- = litre per 1000 km


--EntityId, names and addresses


EntityId ::= OBJECT IDENTIFIER


NameAndAddress ::= SEQUENCE {
       name                     Text60,
       otherNamesOrInitials     Text60 OPTIONAL,
       address                  Text60
       }


--Text


Text60 ::= UTF8String (SIZE (1..60))


--Integers


Int1 ::= INTEGER (0..255)


END
```

A.3 ElectronicRegistrationIdentificationEfcVehicleDataModule

```
ElectronicRegistrationIdentificationEfcVehicleDataModule {iso(1) standard(0) iso24534
(24534) efcVehicleData (3) version1 (1)} DEFINITIONS AUTOMATIC TAGS ::= BEGIN


IMPORTS
CountryCode FROM AVIAEINumberingAndDataStructures {iso(1) standard(0) iso14816(14816)
```

```
asnm1(1) version1(1)}
;


-- Vehicle dimensions


VehicleDimensions ::= SEQUENCE {
       vehicleLengthOverall     Int1,
       vehicleHeigthOverall     Int1,
       vehicleWidthOverall      Int1
       }


-- Passengers capicity


PassengerCapacity ::= SEQUENCE{
       numberOfSeats            Int1,
       numberOfStandingPlaces   Int1
       }


-- Vehicle weight limits


VehicleWeightLimits ::= SEQUENCE {
       vehicleMaxLadenWeight       Int2,
       vehicleTrainMaximumWeight   Int2,
       vehicleWeightUnladen        Int2
       }


-- Axle weight limits


AxleWeightLimits ::= SEQUENCE{
       maxLadenweightOnAxle1    Int2,
       maxLadenweightOnAxle2    Int2,
       maxLadenweightOnAxle3    Int2,
       maxLadenweightOnAxle4    Int2,
       maxLadenweightOnAxle5    Int2
       }


-- Vehicle Specific Characteristics


VehicleSpecificCharacteristics ::= SEQUENCE {
       environmentalCharacteristics  EnvironmentalCharacteristics,
       engineCharacteristics         EngineCharacteristics,
       descriptiveCharacteristics    DescriptiveCharacteristics,
       futureCharacteristics         FutureCharacteristics
       }


EnvironmentalCharacteristics::= SEQUENCE {
       euroValue                EuroValue,
       copValue                 CopValue
       }


EuroValue::= INTEGER {
       noEntry                  (0),
       euro-1                   (1),
       euro-2                   (2),
       euro-3                   (3),
       euro-4                   (4),
       euro-5                   (5),
       euro-6                    (6)
       } (0..15) -- 4 bits, EURO-Clases as defined in EC directive 88/77/EEC, annex 1
       -- and in 91/542/EEC, 96/1/EC, 1999/96/EC, 2001/27/EC


CopValue ::= INTEGER             {
       noEntry                   (0),
       co2class1                 (1),                 -- below 101 g/km
       co2class2                 (2),                 -- 101 to 120 g/km
       co2class3                 (3),                 -- 121 to 140 g/km
       co2class4                 (4),                 -- 141 to 160 g/km
       co2class5                 (5),                 -- 161 to 200 g/km
       co2class6                 (6),                 -- 201 to 250 g/km
       co2class7                 (7)                  -- above 250 g/km
       } (0..15)  -- 4 bits, reserved for carbon dioxide pollution values as defined in
                -- EC directive 2003/127/EC


EngineCharacteristics::= INTEGER {
       noEntry                   (0),
       noEngine                  (1),
       petrolUnleaded            (2),
       petrolLeaded              (3),
       diesel                    (4),
       lPG                       (5),
       battery                   (6),
       solar                     (7)
       -- (8-255) are reserved for future CEN use
       } (0..255)


DescriptiveCharacteristics ::= INTEGER {
       noEntry                   (0),
       vehicleShape1             (1),
       vehicleShape2             (2),
       vehicleShape3             (3),
       vehicleShape4             (4),
       vehicleShape5             (5),
       vehicleShape6             (6),
       vehicleShape7             (7),
       vehicleShape8             (8),
       vehicleShape9             (9),
       vehicleShape10            (10),
       vehicleShape11            (11),
       vehicleShape12            (12),
       vehicleShape13            (13),
       vehicleShape14            (14),
       vehicleShape15            (15),
       vehicleShape16            (16),
       vehicleShape17            (17),
       vehicleShape18            (18),
       vehicleShape19            (19),
       vehicleShape20            (20),
       vehicleShape21            (21),
       vehicleShape22            (22),
       vehicleShape23            (23),
       vehicleShape24            (24),
       vehicleShape25            (25),
       vehicleShape26            (26),
       vehicleShape27            (27),
       vehicleShape28            (28),
       vehicleShape29            (29),
       vehicleShape30            (30),
       vehicleShape31            (31),
       vehicleShape32            (32),
       vehicleShape33            (33),
       vehicleShape34            (34),
       vehicleShape35            (35),
       vehicleShape36            (36),
       vehicleShape37            (37),
       vehicleShape38            (38),
       vehicleShape39           (39),
       vehicleShape40           (40),
       vehicleShape41           (41),
       vehicleShape42           (42),
       vehicleShape43           (43),
       vehicleShape44           (44),
       vehicleShape45           (45),
       vehicleShape46           (46),
       vehicleShape47           (47),
       vehicleShape48           (48),
       vehicleShape49           (49),
       vehicleShape50           (50)
       -- (1..50) are reserved for future CEN ISO use
       -- (51..255) are reserved for private use
       } (0..255)


FutureCharacteristics ::= INTEGER {
       noEntry                  (0),
       airSuspension            (1)
       -- (2..255) are reserved for future CEN use
       } (0..255)


-- Trailer Characteristics


TrailerCharacteristics ::= SEQUENCE {
       trailerDetails           TrailerDetails,
       trailerMaxLadenWeight    Int2,
       trailerWeightUnladen     Int2
       }


TrailerDetails::= SEQUENCE {
       trailerType              TrailerType,
       trailerAxles             TrailerAxles
       }


TrailerType ::= INTEGER{
       notPresent    (0), -- trailer not attached or only one trailer attached, see
                           -- VehicleAxlesNumber for more information
       Trailer       (1),  -- also known as pull-bar trailer
       semitrailer   (2)   -- also known as articulate trailer
       -- (3..31) reserved for future CEN/ISO use
       } (0..31)


TrailerAxles ::= INTEGER (0..7) -- number of axles of the trailer when available


-- Engine


Engine ::= SEQUENCE{
       engineCapacity              Int2,
       enginePower                 Int2
       }


-- Environmental characteristics


SoundLevel ::= SEQUENCE{
       soundstationary             Int1,
       sounddriveby                Int1
       }
```

```
CO2EmissionValue ::= Int2


ExhaustEmissionValues ::= SEQUENCE {
       unitType                    UnitType,
       emissionCO                  INTEGER (0..32766),
       emissionHC                  Int2,
       emissionNOX                 Int2,
       emissionHCNOX               Int2
       }


DieselEmissionValues ::= SEQUENCE {
       unitType     UnitType,
       particulate                 INTEGER (0.. 32767),
       absorptionCoeff             Int2
       }


UnitType ::= ENUMERATED {
       mg-km,
       mg-kWh
       }


Int1 ::= INTEGER(0..255)

Int2 ::= INTEGER(0..65535)


Int4 ::= INTEGER(0..4294967295)

END
```

A.4 ElectronicRegistrationIdentificationVehicleDataAttributesModule

```
ElectronicRegistrationIdentificationVehicleDataAttributesModule {iso(1) standard(0)
iso24534 (24534) vehicleDataAttributes (4) version1 (1)}
DEFINITIONS AUTOMATIC TAGS ::= BEGIN


IMPORTS
CountryCode FROM AVIAEINumberingAndDataStructures {iso(1) standard(0) iso14816(14816)
asnm1(1) version1(1)}


VehicleId , VIN, RegistrationAuthority, VehicleIdStatus, DateOfFirstRegistration,
DateOfRegistration, ValidThru, ChassisNumber, RegistrationNumber, VehicleMake,
VehicleType, VehicleTypeStatus, CommercialDescription, TypeApprovalNumber,
VehicleCategory, VehicleTaxCategory, EuVehicleCategoryCode, RaSpecificVehicleClass1,
RaSpecificVehicleClass2, RaSpecificVehicleClass3, VehicleUse, PrivateUse, VehicleColour,
VehicleLength, VehicleWidth, VehicleHeight, Wheelbase, VehicleBodyShape,
EuBodyWorkType, Iso3833VehicleType, NumberOfSeats, NumberOfStandingPlaces,
MaxNumberOfPassengers, UnladenWeight, MaxDesignLadenMass, MaxAuthorizedLadenMass,
MaxAuthorizedTrainMass, MaxAuthorizedPayload, NumberOfAxles, AuthorizedAxleLadenMass,
MaxTowableMassBrakedTrailer, MaxTowableMassUnbrakedTrailer, EngineId, PrimeEngineType,
EnginePowerSources, PrimePowerSource, EngineMaxNetPower, EngineDisplacement,
RatedEngineSpeed, PowerWeightRatio, MaxSpeed, FuelTanksCapacity, StationarySoundLevel,
EngineSpeed, DriveBySoundLevel, DriveBySoundLevel, EmissionHC, EmissionNOx,
EmissionHCandNOx, ParticulatesForDiesel, CorrectedAbsorptionCoefficient, EmissionCO2,
CombinedFuelConsumption, EnvironmentalCategory, EuroType, OfficialVehicleTestData,
RaSpecificData, AxlesPerAxleGroup, AuthorizedAxleGroupsLadenMass FROM
ElectronicRegistrationIdentificationVehicleDataModule {iso(1) standard(0) iso24534 (24534)
vehicleData (1) version1 (1)}
```

```
VehicleDimensions, PassengerCapacity, VehicleWeightLimits, AxleWeightLimits,
VehicleSpecificCharacteristics, TrailerCharacteristics, Engine,
CO2EmissionValue, SoundLevel, ExhaustEmissionValues, DieselEmissionValues FROM
ElectronicRegistrationIdentificationEfcVehicleDataModule {iso(1) standard(0) iso24534
(24534) efcVehicleData (3) version1 (1)};


-- Usefull types


EriAttributeIdList ::= SEQUENCE OF EriAttributeId


EriAttrubuteList   ::= SEQUENCE OF EriAttribute


EriAttributeId ::= ATTRIBUTE.&id ({EriAttributes })


EriAttribute    ::= SEQUENCE {
       id                      ATTRIBUTE.&id ({EriAttributes }),
       value                   ATTRIBUTE.&Type ({EriAttributes } {@.id})
       }


EriBasicAttributeIdList ::= SEQUENCE OF EriBasicAttributeId
EriBasicAttrubuteList   ::= SEQUENCE OF EriBasicAttribute


EriBasicAttributeId ::= ATTRIBUTE.&id ({EriBasicAttributes })
EriBasicAttribute    ::= SEQUENCE {
       id                      ATTRIBUTE.&id ({EriBasicAttributes }),
       value                   ATTRIBUTE.&Type ({EriBasicAttributes } {@.id})
       }


EriEuAttributeIdList ::= SEQUENCE OF EriEuAttributeId
EriEuAttrubuteList   ::= SEQUENCE OF EriEuAttribute


EriEuAttributeId ::= ATTRIBUTE.&id ({EriEuAttributes})
EriEuAttribute    ::= SEQUENCE {
       id                      ATTRIBUTE.&id ({EriEuAttributes}),
       value                   ATTRIBUTE.&Type ({EriEuAttributes} {@.id})
       }


EriEfcAttributeIdList ::= SEQUENCE OF EriEfcAttributeId
EriEfcAttrubuteList   ::= SEQUENCE OF EriEfcAttribute


EriEfcAttributeId ::= ATTRIBUTE.&id ({EriEfcAttributes})
EriEfcAttribute    ::= SEQUENCE {
       id                      ATTRIBUTE.&id ({EriEfcAttributes}),
       value                   ATTRIBUTE.&Type ({EriEfcAttributes} {@.id})
       }


-- The Attribute information object class


ATTRIBUTE ::= CLASS {
       &Type,
       &id                     Int2 UNIQUE
       }
WITH SYNTAX {&Type IDENTIFIED BY &id}


EriAttributes ATTRIBUTE ::= {(EriBasicAttributes UNION EriEfcAttributes)}
```

```
EriBasicAttributes ATTRIBUTE ::= {vehicleId | vin | registrationAuthority | vehicleIdStatus
| dateOfFirstRegistration | dateOfRegistration | validThru | chassisNumber |
registrationNumber | vehicleMake | vehicleType | vehicleTypeStatus | commercialDescription
| typeApprovalNumber | vehicleCategory | vehicleTaxCategory | euVehicleCategoryCode
| raSpecificVehicleClass1 | raSpecificVehicleClass2 | raSpecificVehicleClass3 |
vehicleUse | privateUse | colour | length | width | height | wheelbase | bodyShape
| euBodyWorkType | iso3833VehicleType | numberOfSeats | numberOfStandingPlaces |
maxNumberOfPassengers | unladenWeight | maxDesignLadenMass | maxAuthorizedLadenMass |
maxAuthorizedTrainMass | maxAuthorizedPayload | numberOfAxles | authorizedAxleLadenMass |
maxTowableMassBrakedTrailer | maxTowableMassUnbrakedTrailer | engineId | primeEngineType
| enginePowerSources | primePowerSource | engineMaxNetPower | engineDisplacement |
ratedEngineSpeed | powerWeightRatio | maxSpeed | fuelTanksCapacity | stationarySoundLevel
| engineSpeed | driveBySoundLevel | emissionCO | emissionHC | emissionNOx |
emissionHCandNOx | particulatesForDiesel | correctedAbsorptionCoefficient | emissionCO2
| combinedFuelConsumption | environmentalCategory | euroType | lastOfficialTestData |
raSpecificData | axlesPerAxleGroup | authorizedAxleGroupLadenMass, ...}


EriEuAttributes ATTRIBUTE ::= {  registrationNumber | dateOfFirstRegistration |
vehicleMake | vehicleType | commercialDescription | vin | maxDesignLadenMass |
maxAuthorizedLadenMass | maxAuthorizedTrainMass | unladenWeight | validThru |
dateOfRegistration | vehicleCategory | euVehicleCategoryCode | typeApprovalNumber |
numberOfAxles | wheelbase | authorizedAxleLadenMass | maxTowableMassBrakedTrailer
| maxTowableMassUnbrakedTrailer | engineDisplacement | engineMaxNetPower |
enginePowerSources | ratedEngineSpeed | engineId | powerWeightRatio | colour |
numberOfSeats | numberOfStandingPlaces | maxSpeed | stationarySoundLevel | engineSpeed
| driveBySoundLevel | emissionCO | emissionHC | emissionNOx | emissionHCandNOx
| particulatesForDiesel | correctedAbsorptionCoefficient | emissionCO2 |
combinedFuelConsumption | euroType | fuelTanksCapacity, ...}


EriEfcAttributes ATTRIBUTE ::= { efcVehicleLicencePlateNumber |
efcTrailerLicencePlateNumber | efcVehicleDimensions | efcPassengerCapacity |
efcVehicleWeightLimits | efcAxleWeightLimits | efcVehicleSpecificCharacteristics
| efcTrailerCharacteristics | efcEngine | efcSoundLevel | efcCO2EmissionValue |
efcExhaustEmissionValues | efcDieselEmissionValues, ...}


-- Vehicle Identification and administrative data


vehicleId ATTRIBUTE ::={VehicleId IDENTIFIED BY vehicleIdId}


vin ATTRIBUTE ::={VIN IDENTIFIED BY vinId}


registrationAuthority ATTRIBUTE ::={RegistrationAuthority IDENTIFIED BY
registrationAuthorityId}


vehicleIdStatus ATTRIBUTE ::={VehicleIdStatus IDENTIFIED BY vehicleIdStatusId}


dateOfFirstRegistration ATTRIBUTE ::={DateOfFirstRegistration IDENTIFIED BY
dateOfFirstRegistrationId}


dateOfRegistration ATTRIBUTE ::={DateOfRegistration IDENTIFIED BY dateOfRegistrationId}


validThru ATTRIBUTE ::={ValidThru IDENTIFIED BY validThruId}


chassisNumber ATTRIBUTE ::={ ChassisNumber IDENTIFIED BY chassisNumberId}
```

```
registrationNumber ATTRIBUTE ::={ RegistrationNumber IDENTIFIED BY registrationNumberId}

-- Vehicle type

vehicleMake ATTRIBUTE ::={VehicleMake IDENTIFIED BY vehicleMakeId}

vehicleType ATTRIBUTE ::={VehicleType IDENTIFIED BY vehicleTypeId}

vehicleTypeStatus ATTRIBUTE ::={VehicleTypeStatus IDENTIFIED BY vehicleTypeStatusId}

commercialDescription ATTRIBUTE ::={CommercialDescription IDENTIFIED BY
commercialDescriptionId}

typeApprovalNumber ATTRIBUTE ::={TypeApprovalNumber IDENTIFIED BY typeApprovalNumberId}

vehicleCategory ATTRIBUTE ::={VehicleCategory IDENTIFIED BY vehicleCategoryId}

vehicleTaxCategory ATTRIBUTE ::={VehicleTaxCategory IDENTIFIED BY vehicleTaxCategoryId}

euVehicleCategoryCode ATTRIBUTE ::={EuVehicleCategoryCode IDENTIFIED BY
euVehicleCategoryCodeId}

raSpecificVehicleClass1 ATTRIBUTE ::={RaSpecificVehicleClass1 IDENTIFIED BY
raSpecificVehicleClass1Id}

raSpecificVehicleClass2 ATTRIBUTE ::={RaSpecificVehicleClass2 IDENTIFIED BY
raSpecificVehicleClass2Id}

raSpecificVehicleClass3 ATTRIBUTE ::={RaSpecificVehicleClass3 IDENTIFIED BY
raSpecificVehicleClass3Id}

vehicleUse ATTRIBUTE ::={VehicleUse IDENTIFIED BY vehicleUseId}

privateUse ATTRIBUTE ::={PrivateUse IDENTIFIED BY privateUseId}

colour ATTRIBUTE ::={VehicleColour IDENTIFIED BY colourId}

-- Vehicle shape

length ATTRIBUTE ::={VehicleLength IDENTIFIED BY lengthId}

width ATTRIBUTE ::={VehicleWidth IDENTIFIED BY widthId}

height ATTRIBUTE ::={VehicleHeight IDENTIFIED BY heightId}

wheelbase ATTRIBUTE ::={Wheelbase IDENTIFIED BY wheelbaseId}
```

```
bodyShape ATTRIBUTE ::={VehicleBodyShape IDENTIFIED BY bodyShapeId}

euBodyWorkType ATTRIBUTE ::={EuBodyWorkType IDENTIFIED BY euBodyWorkTypeId}

iso3833VehicleType ATTRIBUTE ::={Iso3833VehicleType IDENTIFIED BY iso3833VehicleTypeId}

-- Vehicle number of passengers, axles and mass

numberOfSeats ATTRIBUTE ::={NumberOfSeats IDENTIFIED BY numberOfSeatsId}

numberOfStandingPlaces ATTRIBUTE ::={NumberOfStandingPlaces IDENTIFIED BY
numberOfStandingPlacesId}

maxNumberOfPassengers ATTRIBUTE ::={MaxNumberOfPassengers IDENTIFIED BY
maxNumberOfPassengersId}

unladenWeight ATTRIBUTE ::={UnladenWeight IDENTIFIED BY unladenWeightId}

maxDesignLadenMass ATTRIBUTE ::={MaxDesignLadenMass IDENTIFIED BY maxDesignLadenMassId}

maxAuthorizedLadenMass ATTRIBUTE ::={MaxAuthorizedLadenMass IDENTIFIED BY
maxAuthorizedLadenMassId}

maxAuthorizedTrainMass ATTRIBUTE ::={MaxAuthorizedTrainMass IDENTIFIED BY
maxAuthorizedTrainMassId}

maxAuthorizedPayload ATTRIBUTE ::={MaxAuthorizedPayload IDENTIFIED BY
maxAuthorizedPayloadId}

numberOfAxles ATTRIBUTE ::={NumberOfAxles IDENTIFIED BY numberOfAxlesId}

authorizedAxleLadenMass ATTRIBUTE ::={AuthorizedAxleLadenMass IDENTIFIED BY
authorizedAxleLadenMassId}

maxTowableMassBrakedTrailer ATTRIBUTE ::={MaxTowableMassBrakedTrailer IDENTIFIED BY
maxTowableMassBrakedTrailerId}

maxTowableMassUnbrakedTrailer ATTRIBUTE ::={MaxTowableMassUnbrakedTrailer IDENTIFIED BY
maxTowableMassUnbrakedTrailerId}

-- Vehicle engine and power source

engineId ATTRIBUTE ::={EngineId (SIZE (1..60)) IDENTIFIED BY engineIdId}

primeEngineType ATTRIBUTE ::={PrimeEngineType IDENTIFIED BY primeEngineTypeId}

enginePowerSources ATTRIBUTE ::={EnginePowerSources IDENTIFIED BY enginePowerSourcesId}
```