Economic Commission for Europe

Inland Transport Committee

Working Party on Road Transport

**Group of Experts on European Agreement Concerning Work of
Crews of Vehicles Engaged in International Road Transport (AETR)**

Twenty-second session
Geneva, 15 October 2019

Sub-appendix 13 - ITS Interface

Submitted by the European Commission

This document, submitted by the European Commission, contains amendment proposals (identified in track changes) which aim at modifying Sub-appendix 13 as presented in Informal document No.2 (June 2019).

Please recycle

Sub-appendix 13 - ITS Interface

TABLE OF CONTENT**S**

| **Deleted:** REGULATIONS AND |

| **Deleted:** 5 |
| **Deleted:** 5 |
| **Deleted:** 6 |

2

1. Introduction

This Sub-appendix specifies the design and the procedures to follow in order to implement the interface with Intelligent Transport Systems (ITS).

The tachographs of vehicles may be equipped with standardised interfaces allowing the data recorded or produced by tachograph to be used in operational mode, by an external device, provided that the following conditions are met:

(a) the interface does not affect the authenticity and the integrity of the data of the tachograph;

(b) the interface complies with the detailed provisions of this Sub-appendix;

(c) the external device connected to the interface has access to personal data, including geopositioning data, only after the verifiable consent of the driver to whom the data relates.

2. Scope

The scope of this Sub-appendix is to specify how applications hosted on external devices can via a Bluetooth® connection obtain data (*the Data*) from a tachograph.

The Data available via this interface is described in the Annex 1 of the present document. This interface does not prohibit the implementation of other interfaces (e.g. via the CAN bus) to transmit the data of the VU to other vehicle processing units.

This Sub-appendix specifies:

— *The Data* available through the ITS interface
— The Bluetooth® profile that is used to transfer the data
— The enquiry and download procedures and sequence of operations
— The pairing mechanism between the tachograph and the external device
— The consent mechanism available to the driver

For clarification, this Sub-appendix does <u>not</u> specify:

— The collection of *the Data* operation and management within the VU (which shall be specified elsewhere within this Agreement or otherwise shall be a function of product design).

— The form of presentation of collected data to application hosted on the external device.

— Data security provisions above what provides Bluetooth® (such as encryption) concerning the content of *the Data* (which shall be specified elsewhere within this Agreement [Sub-appendix 11 Common Security Mechanisms]).

— The Bluetooth® protocols used by the ITS interface

2.1. Acronyms, definitions and notations

The following acronyms and definitions specific to this Sub-appendix are used in this Sub-appendix:

the Communication	exchange of information/data between a master unit (i.e. the tachographs) and an external unit through the ITS interface over Bluetooth®.
the Data	Data sets as specified in Annex 1.

BR	Basic Rate
EDR	Enhanced Data Rate
GNSS	Global Navigation Satellite System
IRK	Identity Resolution Key
ITS	Intelligent Transport System
LE	Low Energy
PIN	Personal Identification Number

Deleted: Appendix

Deleted: as required in Article 10 of Regulation (EU) N°. 165/2014 (*the Regulation*)

Deleted: *The Regulation* specifies that t

Deleted: Article 11 of the Regulation

Deleted: Appendix

Deleted: Appendix

Deleted: Appendix

Deleted: *the Regulation*

Deleted: the Regulation

Formatted: Font: Not Italic

Deleted: Appendix

Deleted: Appendix

Deleted: appendix

Deleted: *the Regulation* Regulation (EU) N°. 165/2014 of the European Parliament and of the Council of 4 February 2014 on tachographs in road transport, repealing Council Regulation (EEC) N°. 3821/85 on recording equipment in road transport and amending Regulation (EC) N°. 561/2006 of the European Parliament and of the Council on the harmonisation of certain social legislation relating to road transport¶

PUC	Personal Unblocking Code
SID	Service Identifier
SPP	Serial Port Profile
SSP	Secure Simple Pairing
TRTP	Transfer Request Parameter
TREP	Transfer Response Parameter
VU	Vehicle Unit

3. Referenced Standards

The specification defined in this Sub-appendix refers to and depends upon all or parts of the following standards. Within the clauses of this Sub-appendix the relevant standards, or relevant clauses of standards, are specified. In the event of any contradiction the clauses of this Sub-appendix shall take precedence.

Standards referenced in this Sub-appendix are:

- ISO 16844 – 4 : Road vehicles – Tachograph systems – Part 4: Can interface
- ISO 16844 – 7 : Road vehicles – Tachograph systems – Part 7: Parameters
- Bluetooth® – Serial Port Profile – V1.2
- Bluetooth® – Core Version 4.2
- NMEA 0183 V4.1 protocol

4. Interface working principles

4.1. Preconditions to data transfer via the ITS interface

The VU shall be responsible to keep updated and maintain the data to be stored in the VU, without any involvement of the ITS interface. The means by which this is achieved is internal to the VU, specified elsewhere in this Agreement, and is not specified in this Sub-appendix.

4.1.1 Data provided through the ITS interface

The VU shall be responsible to update the data that will be available through the ITS interface at a frequency determined within VU procedures, without any involvement of ITS interface. The VU data shall be used as a basis to populate and update *the Data,* the means by which this is achieved is specified elsewhere in *the Regulation* or if there is no such specification is a function of product design and is not specified in this Sub-appendix.

4.1.2 Content of the Data

The content of *the Data* shall be as specified in Annex 1 of this Sub-appendix.

4.1.3 ITS Applications

ITS applications will be using the data made available through the ITS interface for instance to optimize driver activities management while respecting the provisions of this Agreement, to detect possible faults of the tachograph or to use the GNSS data. The specification of the applications is not within the scope of this Sub-appendix.

Contracting Parties may set up restrictions to the transmission of data by ITS applications; those restrictions shall not affect the data provided through the ITS interface in accordance with point 4.1.1. Contracting Parties shall abide by the legislation on data protection in force in their respective territories, in what respects to collection, storage, processing and use of personal data obtained using ITS.

4.2. Communication technology

The Data exchange using the ITS interface shall be performed via a Bluetooth® interface compatible via version 4.2 or later. Bluetooth® operates in the unlicensed industrial, scientific and medical (ISM) band at 2.4 to 2.485 GHz.

Margin change markers:

Deleted: Regulations and

Deleted: Appendix

Deleted: regulations and

Deleted: Appendix

Deleted: Appendix

Deleted: Regulations and s

Deleted: Appendix

Deleted: <#>Regulation (EU) N°. 165/2014 of the European Parliament and of the Council of 4 February 2014 on tachographs in road transport, repealing Council Regulation (EEC) N°. 3821/85 on recording equipment in road transport and amending Regulation (EC) N°. 561/2006 of the European Parliament and of the Council on the harmonisation of certain social legislation relating to road transport.¶
<#>Regulation (EC) N°. 561/2006 of the European Parliament and of the Council of 15 March 2006 on the harmonisation of certain social legislation relating to road transport and amending Council Regulations (EEC) N°. 3821/85 and (EC) No 2135/98 and repealing Council Regulation (EEC) N°. 3820/85.¶

Deleted: the Regulation

Deleted: Appendix

Deleted: Appendix

Deleted: appendix

Deleted: Regulation

Deleted: Appendix

Formatted: Font: Not Bold

Formatted: Font: Not Bold

Bluetooth® 4.2 offers enhanced privacy and security mechanisms and increases speed and reliability of data transfers. For the purpose of this specification is Bluetooth® class 2 radio used with a range up to 10 meters. More information on Bluetooth® 4.2 is available on www.bluetooth.com (https://www.bluetooth.org/en-us/specification/adopted-specifications?_ga=1.215147412.2083380574.1435305676).

The Communication shall be established with the communications equipment after a pairing process has been completed by an authorized device. As Bluetooth® is using a master/slave model to control when and where devices can send data, the tachograph will play the role of master while the external device will be the slave.

When an external device comes within range of the VU for the first time, the Bluetooth® pairing process can be initiated (see also annex 2). The devices share their addresses, names, and profiles and common secret key, which allows them to bond whenever they are together in the future. Once this step is completed, the external device is trusted and is in state to initiate requests to download data from the tachograph. It is not foreseen to add encryption mechanisms beyond what Bluetooth® provides. However, if additional security mechanisms are needed, this will be done in accordance with Sub-appendix 11 Common Security Mechanisms.

The overall communication principle is described in the following figure.
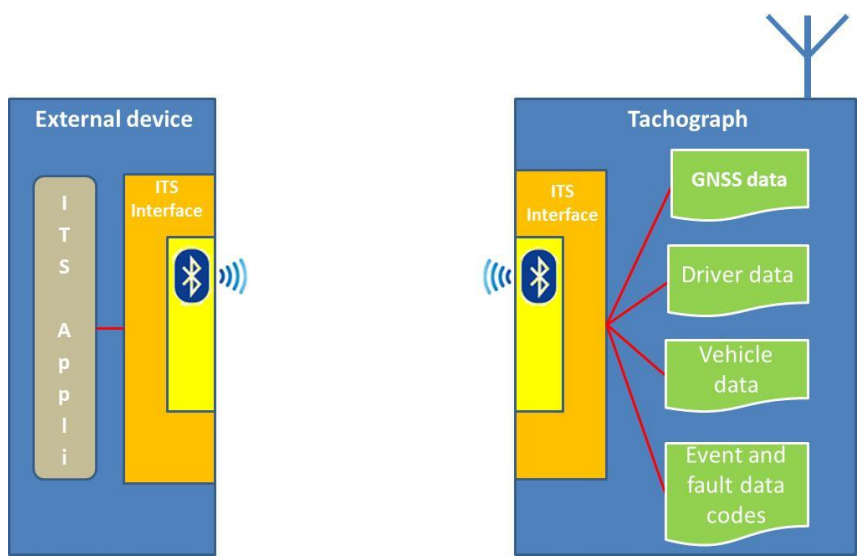
The SPP (Serial Port Profile) profile of Bluetooth® shall be used to transfer data from the VU to the external device.

4.3. PIN authorization

For security reasons, the VU will require a PIN code authorization system separated from the Bluetooth pairing. Each VU shall be able to generate PIN codes for authentication purposes composed of at least 4 digits. Every time an external device pairs with the VU, it must provide the correct PIN code before receiving any data.

Succeeding entering the PIN shall result in putting the device on the whitelist. The whitelist shall store at least 64 devices paired with the particular VU.

Failing to provide the correct PIN code three times in a row shall result in putting temporarily the device on the blacklist. While blacklisted, every new attempt from the device shall be rejected. Further failure to provide the correct PIN code three times in a row shall result in increasingly longer ban duration (See table 1). Providing the

correct PIN code shall reset the ban duration and the number of attempt. Figure 1 in Annex 2 represents the sequence diagram of a PIN validation attempt.

Number of consecutive failure	Ban duration
3	30 seconds
6	5 minutes
9	1 hour
12	24 hours
15	Permanent

Table 1: Ban duration depending on the number of consecutive failure to provide the correct PIN code

Failing to provide the correct PIN code fifteen times (5x3) in a row shall result in a permanent blacklisting of the ITS Unit. Only providing the correct PUC code shall overturn this permanent ban.

The PUC code shall be composed of 8 digits and provided by the manufacturer with the VU. Failing to provide the correct PUC code ten times in a row will irrevocably blacklist the ITS Unit.

While the manufacturer may offer an option to change the PIN code directly through the VU, the PUC code shall not be alterable. Modifying the PIN code, if possible, shall require to enter the current PIN code directly in the VU.

Furthermore any devices stored in the whitelist shall be kept until manual removal of by the user (e.g. via the man-machine-interface of the VU or other means). By doing so lost or stolen ITS-units may be removed from the whitelist. Also, any ITS Unit leaving the Bluetooth connection range for more than 24 hours shall be automatically removed from the VU whitelist and must provide the correct PIN code again when the connection is established again.

The format of the messages between the VU interface and the VU are not provided but left to the discretion of the manufacturer. Said manufacturer shall however ensure the message format between the ITS Unit and the VU interface is respected (see ASN.1 specifications).

Any data request shall thus be met with the proper verification of the sender's credential before any form of treatment. Figure 2 of Annex 2 represents the sequence diagram for this procedure. Any blacklisted device shall receive an automatic rejection, any non-blacklisted non-whitelisted device shall receive a PIN request it needs to fulfill before resending its data request.

4.4. Message Format

All messages exchanged between the ITS Unit and the VU interface shall be formatted with a structure consisting of three parts: A header composed by a target byte (TGT), a source byte (SRC) and a length byte (LEN).

The data field composed by a service identifier byte (SID) and a variable amount of data bytes (maximum 255).

The checksum byte is the 1 byte sum series modulo 256 of all the bytes of the message excluding the CS itself.

The message shall be Big Endian.

Header			Data Field					Checksum
TGT	SRC	LEN	SID	TRTP	CC	CM	DATA	CS
3 bytes			Max. 255 bytes					1 byte

Table 2: General message format.

Header

TGT and SRC : the ID of the Target (TGT) and Source (SRC) devices of the message. The VU Interface shall have the default ID "EE". This ID cannot be changed. The ITS Unit shall use the default ID "A0" for its first message of

the communication session. The VU Interface shall then assign an unique ID to the ITS Unit and informs it of this ID for future messages during the session.

The LEN byte shall only take into account the "DATA" part of the Data Field (see Table 2), the 4 first bytes are implicit.

The VU Interface shall confirm the authenticity of the message's sender by cross-checking its own IDList with the Bluetooth data by checking the ITS Unit listed at the provided ID is currently in the range of the Bluetooth connection.

Data Field

Besides the SID, the Data Field shall also contain other parameters : a transfer request parameter (TRTP) and Counter bytes.

If the data to be handled is larger than the available space in one message, it will be split in several submessages. Each submessage shall have the same Header and SID, but will contain a 2-bytes counter, Counter Current (CC) and Counter Max (CM), to indicate the submessage number. To enable error checking and abort the receiving device acknowledges every submessage. The receiving device can accept the submessage, ask for it to be re-transmitted, request the sending device to start again or abort the transmission.

If not used, CC and CM shall be given the value 0xFF.

For instance, the following message

HEADER	SID	TRTP	CC	CM	DATA	CS
3 bytes	Longer than 255 bytes					1 byte

Shall be transmitted as such:

HEADER	SID	TRTP	01	n	DATA	CS
3 bytes	255 bytes					1 byte

HEADER	SID	TRTP	02	n	DATA	CS
3 bytes	255 bytes					1 byte

…

HEADER	SID	TRTP	N	N	DATA	CS
3 bytes	Max. 255 bytes					1 byte

Table 3 contains the messages the VU and the ITS Unit shall be able to exchange. The content of each parameter is given in hexadecimal. Aren't represented in the table CC and CM for clarity, see above for complete format.

Message	Header			DATA			Checksum
	TGT	SRC	LEN	SID	TRTP	DATA	
RequestPIN	*ITSID*	EE	00	01	FF	███████	
SendITSID	*ITSID*	EE	01	02	FF	*ITSID*	
SendPIN	EE	*ITSID*	04	03	FF	4*INTEGER (0..9)	
PairingResult	*ITSID*	EE	01	04	FF	BOOLEAN (T/F)	
SendPUC	EE	*ITSID*	08	05	FF	8*INTEGER (0..9)	

BanLiftingResult	*ITSID*	EE	01	06	FF	BOOLEAN (T/F)	
RequestRejected	*ITSID*	EE	08	07	FF	Time	
RequestData							
standardTachData	EE	*ITSID*	01	08	01	■	
personalTachData	EE	*ITSID*	01	08	02	■	
gnssData	EE	*ITSID*	01	08	03	■	
standardEventData	EE	*ITSID*	01	08	04	■	
personalEventData	EE	*ITSID*	01	08	05	■	
standardFaultData	EE	*ITSID*	01	08	06	■	
manufacturerData	EE	*ITSID*	01	08	07	■	
ResquestAccepted	*ITSID*	EE	Len	09	TREP	Data	
DataUnavailable							
No data available	*ITSID*	EE	02	0A	TREP	10	
Personal data not shared	*ITSID*	EE	02	0A	TREP	11	
NegativeAnswer							
General reject	*ITSID*	EE	02	0B	SID Req	10	
Service not supported	*ITSID*	EE	02	0B	SID Req	11	
Sub function not supported	*ITSID*	EE	02	0B	SID Req	12	
Incorrect message length	*ITSID*	EE	02	0B	SID Req	13	
Conditions not correct or request sequence error	*ITSID*	EE	02	0B	SID Req	22	
Request out of range	*ITSID*	EE	02	0B	SID Req	31	

Response pending	*ITSID*	EE	02	0B	SID Req	78	
ITSID Mismatch	*ITSID*	EE	02	0B	SID Req	FC	
ITSID Not Found	*ITSID*	EE	02	0B	SID Req	FB	

Table 3: Detailed message content.

RequestPIN (SID 01)
 This message is issued by the VU Interface if a non-blacklisted but non-whitelisted ITS unit is sending any data request.

SendITSID (SID 02)
 This message is issued by the VU Interface whenever a new device is sending a request. This device shall use the default ID "A0" before getting assigned an unique ID for the communication session.

SendPIN (SID 03)
 This message is issued by the ITS Unit to be whitelisted from the VU interface. The content of this message is a 4 INTEGER between 0 and 9 code.

PairingResult (SID 04)
 This message is issued by the VU Interface to inform the ITS Unit if the PIN code it sent was correct. The content of this message shall be a BOOLEAN with the value "True" if the PIN code was correct and "False" otherwise.

SendPUC (SID 05)
 This message is issued by the ITS Unit to lift a blacklist sanction from the VU interface. The content of this message is a 8 INTEGER between 0 and 9 code.

BanLiftingResult (SID 06)
 This message is issued by the VU Interface to inform the ITS Unit if the PUC code it sent was correct. The content of this message shall be a BOOLEAN with the value "True" if the PUC code was correct and "False" otherwise.

RequestRejected (SID 07)
 This message is issued by the VU Interface as a reply to any message from a blacklisted ITS Unit except "SendPUC". The message shall contain the remaining time the ITS Unit is blacklisted, following the "Time" sequence format as defined in Annex 3.

RequestData (SID 08)
 This message for data accessing is issued by the ITS Unit. A one byte transfer request parameter (TRTP) indicates the type of data required. There are several types of data:
- standardTachData (TRTP 01): Data available from the tachograph classified as non-personal.
- personalTachData (TRTP 02): Data available from the tachograph classified as personal.
- gnssData (TRTP 03): GNSS data, always personal.
- standardEventData (TRTP 04): Recorded event data classified as non-personal.
- personalEventData (TRTP 05): Recorded event data classified as personal.
- standardFaultData (TRTP 06): Recorded faults classified as non-personal.
- manufacturerData (TRTP 07): data made available by the manufacturer.

See Annex 3 of this Sub-appendix for more information about the content of each data type.
See Sub-appendix 12 for more information about the format and content of GNSS data.
See Appendix IB and IC for more information about event data code and faults.

ResquestAccepted (SID 09)

This message is issued by the VU Interface if a ITS Unit "RequestData" message has been accepted. This message contains a 1-byte TREP, which is the TRTP byte of the associated RequestData message, and all the data of the requested type.

DataUnavailable (SID 0A)

This message is issued by the VU Interface if, for a certain reason, the requested data aren't available to be sent to a whitelisted ITS Unit. The message contains a 1byte TREP which is the TRTP of the required data and a 1 byte error code specified in the table 3. The Following codes are available:

- No data available (10) : The VU interface can't access the VU data for unspecified reasons.
- Personal data not shared (11) : The ITS Unit tries to retrieve personal data when they are not shared.

NegativeAnswer (SID 0B)

These messages are issued by the VU Interface if a request cannot be completed for any other reason than the unavailability of the data. These messages are typically the result of a bad request format (Length, SID, ITSID…) but aren't limited to that. The TRTP in the Data Field contains the SID of the request. The Data Field contains a code identifying the reason of the negative answer. The following codes are available:

- General Reject (code : 10)
  The action can't be performed for a reason which isn't cited below nor in section (Enter *DataUnavailable* section number).
- Service not supported (code : 11)
  The request's SID isn't understood.
- Sub function not supported (code : 12)
  The request's TRTP isn't understood. It can be for instance missing or out of accepted values.
- Incorrect message length (code : 13)
  The length of the received message is wrong (mismatch between the LEN byte and the actual message length).
- Conditions not correct or request sequence error (code : 22)
  The required service is not active or the sequence of request messages is not correct
- Request out of range (code : 33)
  The request parameter record (data field) is not valid
- Response pending (code : 78)
  The action requested cannot be completed in time and the VU is not ready to accept another request.
- *ITSID* Mismatch (code : FB)
  The SRC *ITSID* doesn't match the associated device after comparison with the Bluetooth information.
- *ITSID* Not Found (code : FC)
  The SRC *ITSID* isn't associated with any device.

Lines 1 through 72 (**FormatMessageModule**) of the ASN.1 code in Annex 3 specify the messages format as described in table 3. More details about the messages content is given below.

4.5. Driver consent

All the data available are classified as either standard or personal. Personal data shall only be accessible if the driver gave his/her consent, accepting his/her tachograph personal data can leave the vehicle network for third party applications.

Driver consent is given when, at first insertion of a given driver card or workshop card currently unknown to the vehicle unit, the cardholder is invited to express his consent for tachograph related personal data output through the optional ITS interface. (see also Appendix I C paragraph 3.6.2).

Deleted: Annex

The consent status (enabled/disabled) is recorded in the memory of the tachograph.

In case of multiple drivers, only the personal data about the drivers who gave their consent shall be shared with the ITS interface. For instance, if there's two drivers in the vehicle, and only the first driver accepted to share his personal data, the ones concerning the second driver shall not be shared.

4.6. Standard data retrieval

Figure 3 of Annex 2 represents the sequence diagrams of a valid request sent by the ITS Unit to access standard data. The ITS Unit is properly whitelisted and isn't requesting personal data, no further verification is required. The diagrams consider the proper procedure illustrated in Figure 2 of Annex 2 has already been followed. They can be equated to the *REQUEST TREATMENT* gray box of Figure 2.

Amongst available data, shall be considered standard:
- standardTachData (TRTP 01)
- StandardEventData (TRTP 04)
- standardFaultData (TRTP 06)

4.7. Personal data retrieval

Figure 4 of Annex 2 represents the sequence diagram for personal data request processing. As previously stated, the VU interface shall only send personal data if the driver has given his explicit consent (see also 4.5). Otherwise, the request must be automatically rejected.

Amongst available data, shall be considered personal:
- personalTachData (TRTP 02)
- gnssData (TRTP 03)
- personalEventData (TRTP 05)
- manufacturerData (TRTP 07)

4.8. Event and fault data retrieval

ITS units shall be able to request events data containing the list of all the unexpected events. These data are considered standard or personal, see Annex 3. The content of each event is in accordance with the documentation provided in Annex 1 of this Sub-appendix.

Deleted: appendix

ANNEX 1

1°) LIST OF AVAILABLE DATA THROUGH THE ITS INTERFACE

Data	Source	Data classification (personal/not personal)
VehicleIdentificationNumber	Vehicle Unit	not personal
CalibrationDate	Vehicle Unit	not personal
TachographVehicleSpeed speed instant t	Vehicle Unit	personal
Driver1WorkingState Selector driver	Vehicle Unit	personal
Driver2WorkingState	Vehicle Unit	personal
DriveRecognize Speed Threshold detected	Vehicle Unit	not personal
Driver1TimeRelatedStates Weekly day time	Driver Card	personal
Driver2TimeRelatedStates	Driver Card	personal
DriverCardDriver1	Vehicle Unit	not personal
DriverCardDriver2	Vehicle unit	not personal
OverSpeed	Vehicle Unit	personal
TimeDate	Vehicle Unit	not personal
HighResolutionTotalVehicleDistance	Vehicle Unit	not personal
ServiceComponentIdentification	Vehicle Unit	not personal
ServiceDelayCalendarTimeBased	Vehicle Unit	not personal
Driver1Identification	Driver Card	personal
Driver2Identification	Driver Card	personal
NextCalibrationDate	Vehicle Unit	not personal
Driver1ContinuousDrivingTime	Driver Card	personal
Driver2ContinuousDrivingTime	Driver Card	personal
Driver1CumulativeBreakTime	Driver Card	personal
Driver2CumulativeBreakTime	Driver Card	personal
Driver1CurrentDurationOfSelectedActivity	Driver Card	personal
Driver2CurrentDurationOfSelectedActivity	Driver Card	personal
SpeedAuthorised	Vehicle Unit	not personal
TachographCardSlot1	Driver Card	not personal
TachographCardSlot2	Driver Card	not personal
Driver1Name	Driver Card	personal
Driver2Name	Driver Card	personal
OutOfScopeCondition	Vehicle Unit	not personal
ModeOfOperation	Vehicle Unit	not personal
Driver1CumulatedDrivingTimePreviousAndCurrentWeek	Driver Card	personal
Driver2CumulatedDrivingTimePreviousAndCurrentWeek	Driver Card	personal
EngineSpeed	Vehicle Unit	personal
RegisteringMemberState	Vehicle Unit	not personal
VehicleRegistrationNumber	Vehicle Unit	not personal
Driver1EndOfLastDailyRestPeriod	Driver Card	personal
Driver2EndOfLastDailyRestPeriod	Driver Card	personal
Driver1EndOfLastWeeklyRestPeriod	Driver Card	personal
Driver2EndOfLastWeeklyRestPeriod	Driver Card	personal
Driver1EndOfSecondLastWeeklyRestPeriod	Driver Card	personal
Driver2EndOfSecondLastWeeklyRestPeriod	Driver Card	personal
Driver1CurrentDailyDrivingTime	Driver Card	personal
Driver2CurrentDailyDrivingTime	Driver Card	personal
Driver1CurrentWeeklyDrivingTime	Driver Card	personal
Driver2CurrentWeeklyDrivingTime	Driver Card	personal
Driver1TimeLeftUntilNewDailyRestPeriod	Driver Card	personal
Driver2TimeLeftUntilNewDailyRestPeriod	Driver Card	personal
Driver1CardExpiryDate	Driver Card	personal
Driver2CardExpiryDate	Driver Card	personal
Driver1CardNextMandatoryDownloadDate	Driver Card	personal
Driver2CardNextMandatoryDownloadDate	Driver Card	personal
TachographNextMandatoryDownloadDate	Vehicle Unit	not personal
Driver1TimeLeftUntilNewWeeklyRestPeriod	Driver Card	personal
Driver2TimeLeftUntilNewWeeklyRestPeriod	Driver Card	personal
Driver1NumberOfTimes9hDailyDrivingTimesExceeded	Driver Card	personal
Driver2NumberOfTimes9hDailyDrivingTimesExceeced	Driver Card	personal
Driver1CumulativeUninterruptedRestTime	Driver Card	personal
Driver2CumulativeUninterruptedRestTime	Driver Card	personal
Driver1MinimumDailyRest	Driver Card	personal
Driver2MinimumDailyRest	Driver Card	personal
Driver1MinimumWeeklyRest	Driver Card	personal
Driver2MinimumWeeklyRest	Driver Card	personal
Driver1MaximumDailyPeriod	Driver Card	personal
Driver2MaximumDailyPeriod	Driver Card	personal
Driver1MaximumDailyDrivingTime	Driver Card	personal
Driver2MaximumDailyDrivingTime	Driver Card	personal
Driver1NumberOfUsedReducedDailyRestPeriods	Driver Card	personal
Driver2NumberOfUsedReducedDailyRestPeriods	Driver Card	personal
Driver1RemainingCurrentDrivingTime	Driver Card	personal
Driver2RemainingCurrentDrivingTime	Driver Card	personal
GnssPosition	Vehicle Unit	personal

2°) CONTINUOUS GNSS DATA AVAILABLE AFTER DRIVER CONSENT

See Sub-appendix 12 – GNSS.

3°) EVENT CODES AVAILABLE WITHOUT DRIVER CONSENT

Event	Storage rules	Data to be recorded per event
Insertion of a non-valid card	- the 10 most recent events.	- date and time of event, - card(s) type, number, issuing Contracting Party and generation of the card creating the event. - number of similar events that day
Card conflict	- the 10 most recent events.	- date and time of beginning of event, - date and time of end of event, - card(s) type, number, issuing Contracting Party and generation of the two cards creating the conflict.

Last card session not correctly closed	- the 10 most recent events.	- date and time of card insertion,
		- card(s) type, number, issuing Contracting Party and generation,
		- last session data as read from the card:
		  - date and time of card insertion,
		  - VRN, Contracting Party of registration and VU generation.

Power supply interruption (2)	- the longest event for each of the 10 last days of occurrence, - the 5 longest events over the last 365 days.	- date and time of beginning of event, - date and time of end of event, - card(s) type, number, issuing Contracting Party and generation of any card inserted at beginning and/or end of the event, - number of similar events that day.
Communication error with the remote communication facility	- the longest event for each of the 10 last days of occurrence, - the 5 longest events over the last 365 days.	- date and time of beginning of event, - date and time of end of event, - card(s) type, number, issuing Contracting Party and generation of any card inserted at beginning and/or end of the event, - number of similar events that day.
Absence of position information from GNSS receiver	- the longest event for each of the 10 last days of occurrence, - the 5 longest events over the last 365 days.	- date and time of beginning of event, - date and time of end of event, - card(s) type, number, issuing Contracting Party and generation of any card inserted at beginning and/or end of the event, - number of similar events that day.
Communication error with the external GNSS facility	- the longest event for each of the 10 last days of occurrence, - the 5 longest events over the last 365 days.	- date and time of beginning of event, - date and time of end of event, - card(s) type, number, issuing Contracting Party and generation of any card inserted at beginning and/or end of the event, - number of similar events that day.
Motion data error	- the longest event for each of the 10 last days of occurrence, - the 5 longest events over the last 365 days.	- date and time of beginning of event, - date and time of end of event, - card(s) type, number, issuing Contracting Party and generation of any card inserted at beginning and/or end of the event, - number of similar events that day.

14

Vehicle motion conflict	- the longest event for each of the 10 last days of occurrence, - the 5 longest events over the last 365 days.	- date and time of beginning of event, - date and time of end of event, - card(s) type, number, issuing Contracting Party and generation of any card inserted at beginning and/or end of the event, - number of similar events that day.	
Security breach attempt	- the 10 most recent events per type of event.	- date and time of beginning of event, - date and time of end of event (if relevant), - card(s) type, number, issuing Contracting Party and generation of any card inserted at beginning and/or end of the event, - type of event.	
Time conflict	- the longest event for each of the 10 last days of occurrence, - the 5 longest events over the last 365 days.	- control device date and time - GNSS date and time, - card(s) type, number, issuing Contracting Party and generation of any card inserted at beginning and/or end of the event, - number of similar events that day.	

4°) EVENT CODES AVAILABLE WITH DRIVER CONSENT

Event	Storage rules	Data to be recorded per event
Driving without an appropriate card	- the longest event for each of the 10 last days of occurrence, - the 5 longest events over the last 365 days.	- date and time of beginning of event, - date and time of end of event, - card(s) type, number, issuing Contracting Party and generation of any card inserted at beginning and/or end of the event, - number of similar events that day.
Card insertion while driving	- the last event for each of the 10 last days of occurrence,	- date and time of the event, - card(s) type, number, issuing Contracting Party and generation, - number of similar events that day
Over speeding (1)	- the most serious event for each of the 10 last days of occurrence (i.e. the one with the highest average speed), - the 5 most serious events over the last 365 days. - the first event having occurred after the last calibration	- date and time of beginning of event, - date and time of end of event, - maximum speed measured during the event, - arithmetic average speed measured during the event, - card type, number, issuing Contracting Party and generation of the driver card (if applicable), - number of similar events that day.

5°) FAULT DATA CODES AVAILABLE WITHOUT DRIVER CONSENT

Fault	Storage rules	Data to be recorded per fault
Card fault	- the 10 most recent driver card faults.	- date and time of beginning of fault, - date and time of end of fault, - card(s) type, number, issuing Contracting Party and generation.

Control device faults	- the 10 most recent faults for each type of fault, - the first fault after the last calibration.	- date and time of beginning of fault, - date and time of end of fault, - type of fault, - card(s) type, number and issuing Contracting Party and generation of any card inserted at beginning and/or end of the fault.

This fault shall be triggered for any of these failures, while not in calibration mode:

– VU internal fault
– Printer fault
– Display fault
– Downloading fault
– Sensor fault
– GNSS receiver or external GNSS facility fault
– Remote Communication facility fault
– ITS interface fault (if applicable)

6°) MANUFACTURER SPECIFIC EVENTS AND FAULTS WITHOUT DRIVER CONSENT

Event or Fault	Storage rules	Data to be recorded per event
To be defined by Manufacturer	To be defined by Manufacturer	To be defined by Manufacturer

Deleted: Recording equipment

Deleted: Member State

ANNEX 2

SEQUENCE DIAGRAMS OF MESSAGES EXCHANGES WITH THE ITS UNIT.
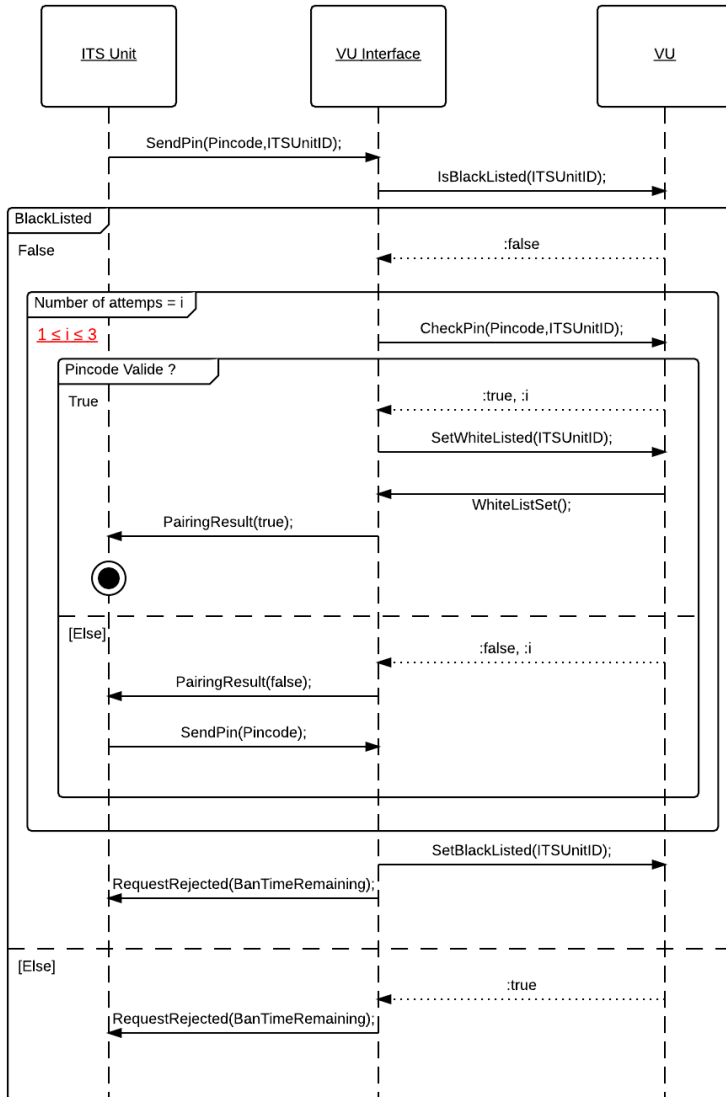
Figure 1. Sequence Diagram for PIN validation attempt
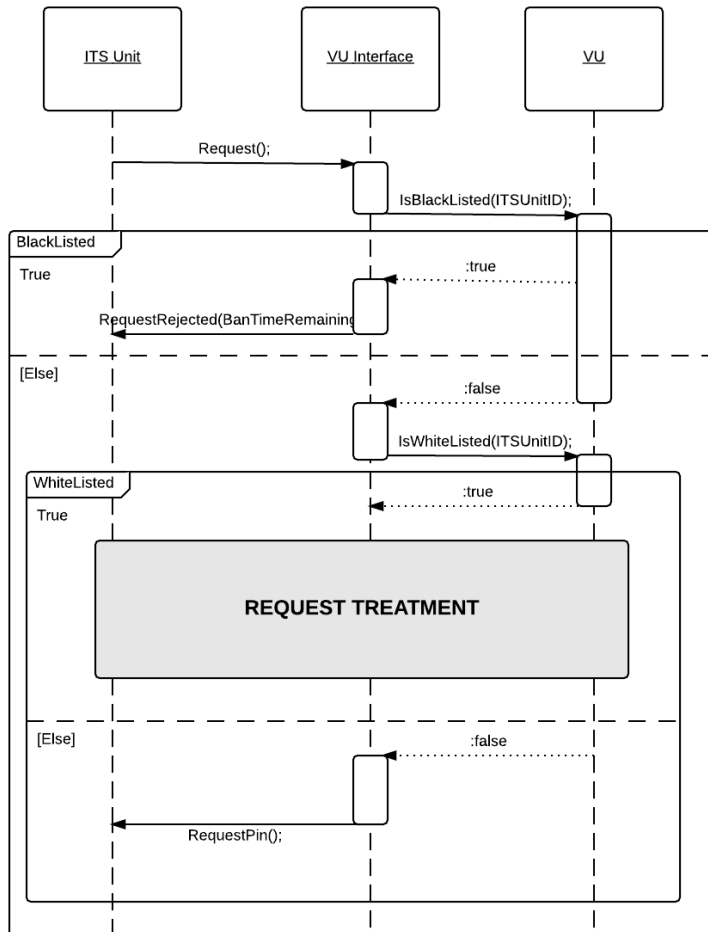
Figure 2. Sequence Diagram for ITS Unit's authorization verification
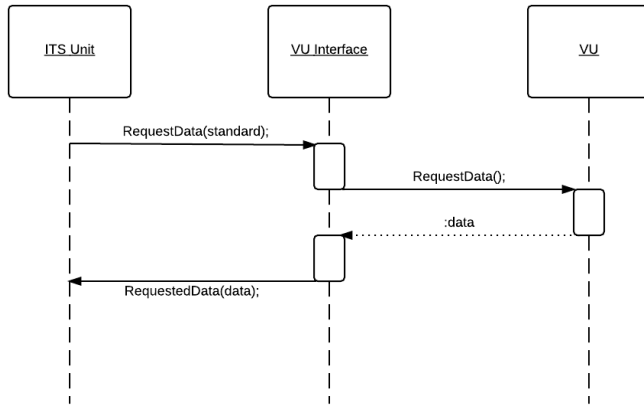
Figure 3. Sequence Diagram to process a request for data classified as non-personal (after correct PIN access)
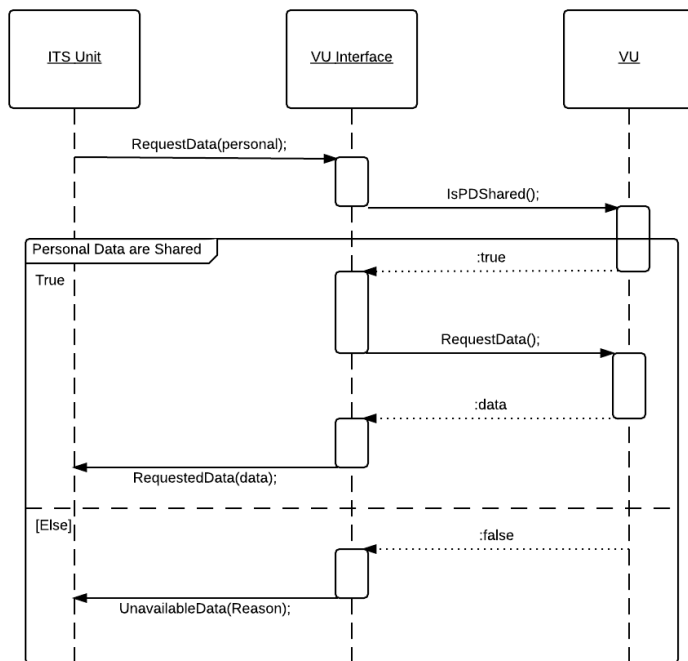
Figure 4. Sequence Diagram to process a request for data classified as personal (after correct PIN access)
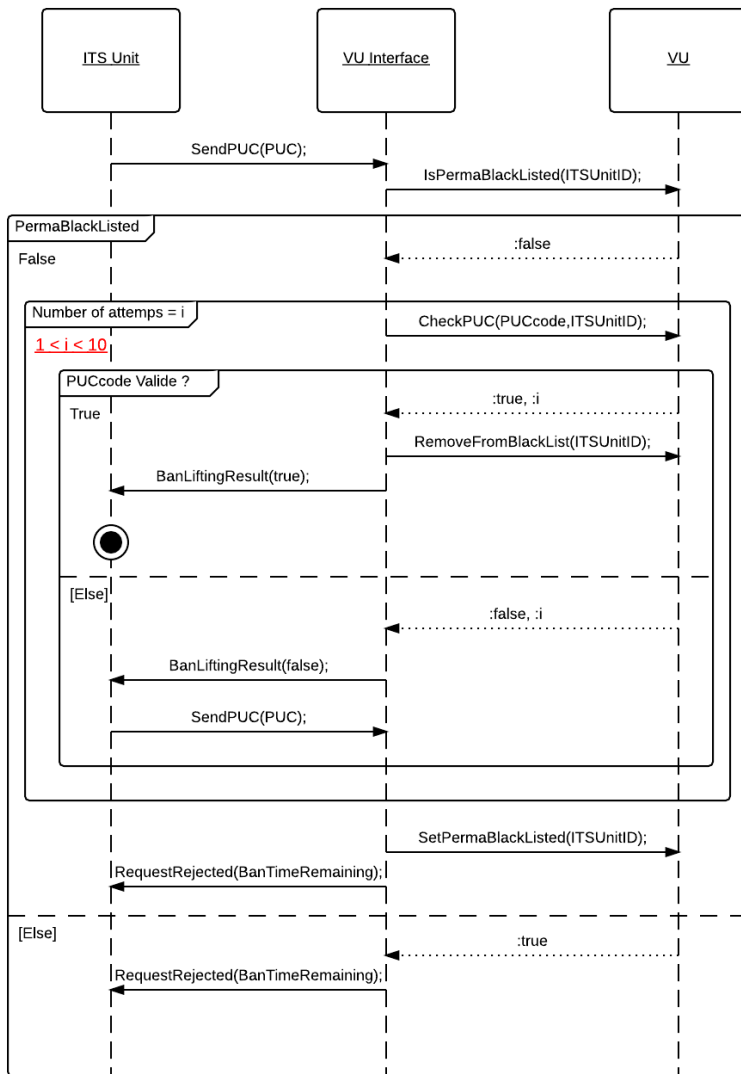
Figure 5. Sequence Diagram for PUC validation attempt

ANNEX 3

ASN.1 SPECIFICATIONS

```
1   FormatMessageModule DEFINITIONS AUTOMATIC TAGS ::= BEGIN
2   EXPORTS ;
3   IMPORTS SendPIN, SendPUC, PairingResult, RequestPIN, RequestRejected,
4           BanLiftingResult FROM PINPUCDataFieldsModule
5           RequestAccepted,RequestData, DataUnavailable FROM
6           RequestDataFieldsModule
7           SendITSID, NegativeAnswer FROM OtherDataFieldsModule;
8
9           CompleteMessage ::=SEQUENCE{
10                  header Header,
11                  data DataField,
12                  checksum Checksum
13          }
14
15          ----------------
16          --HEADER TYPES--
17          ----------------
18
19
20          Header::=SEQUENCE{
21                  tgt IDList,
22                  src IDList,
23                  len BIT STRING (1..255)
24          }
25
26          vuID BIT STRING ::= 'EE'H
27          IDList ::=CHOICE{
28                  vu BIT STRING (vuID),
29                  itsUnits SEQUENCE OF BIT STRING,
30                          --Default hex Value:A0, redefined after first message exchange--
31                          --Each ID will be linked to the Bluetooth ID of the device--
32                  ...
33          }
34
35          --------------------
36          --DATAFIELDS TYPES--
37          --------------------
38          DataField ::=SEQUENCE{
39                  sid BIT STRING,
40                  trtp BIT STRING,
41                  subMBytes SubMessageBytes,
42                  dataField Content,
43                  ...
44          }
45
46          SubMessageBytes ::= SEQUENCE{
47                  currentSubM BIT STRING,
48                  totalSubM BIT STRING
49          }
50
51          Content ::= CHOICE{
52                  requestPIN RequestPIN,
53                  sendITSID SendITSID,
```

21

```
54            sendPin SendPIN,
55            pairRslt PairingResult,
56            sendPUC SendPUC,
57            banlift BanLiftingResult,
58            requestRejected RequestRejected,
59            requestData RequestData,
60            requestOK RequestAccepted,
61            dataUnavailable DataUnavailable,
62            negAns NegativeAnswer
63        }
64
65        ------------------
66        --CHECKSUM TYPES--
67        ------------------
68
69        Checksum ::= SEQUENCE{
70            --SHA2 checksum
71        }
72    END
73
```

```
74    PINPUCDataFieldsModule DEFINITIONS AUTOMATIC TAGS ::= BEGIN
75    EXPORTS SendPIN, SendPUC, PairingResult, RequestPIN, RequestRejected,
76    BanLiftingResult;
77    IMPORTS ;
78
79            ----------
80            ---Utils--
81            ----------
82
83            PUC ::= SEQUENCE (SIZE(8)) OF
84            INTEGER (SIZE(0..9))
85
86            PIN ::=        SEQUENCE (SIZE(4)) OF
87                   INTEGER (SIZE(0..9))
88
89            --------------------------
90            --Messages From ITS Unit--
91            --------------------------
92
93            SendPIN {PIN:pin} ::= SEQUENCE {
94                   sid BIT STRING ('03'H),
95                   pin PIN (pin)
96            }
97
98            SendPUC {PUC:puc} ::= SEQUENCE {
99                   sid BIT STRING ('05'H),
100                  puc PUC (puc)
101           }
102           --------------------
103           --Messages From VU--
104           --------------------
105
106           PairingResult ::= SEQUENCE{
107                  sid BIT STRING ('04'H),
108                  result BOOLEAN
109           }
110
111           RequestPIN {MType:receivedRequest}::= SEQUENCE{
112                  sid BIT STRING ('01'H)
113           }
114
115           RequestRejected ::= SEQUENCE{
116                  sid BIT STRING ('07'H),
117                  banTimeRemaining GeneralizedTime, --PermaBan == 1k years--}
118
119           BanLiftingResult ::= SEQUENCE{
120                  sid BIT STRING ('06'H),
121                  result BOOLEAN
122           }
123    END
124
```

> **Formatted:** French (Belgium)

```
125    RequestDataFields DEFINITIONS AUTOMATIC TAGS ::= BEGIN
126        EXPORTS RequestAccepted,RequestData, DataUnavailable ;
127        IMPORTS StandardEvent, PersonalEvent, StandardFault FROM EventsModule;
128
129        ------------------
130        ---From ITS Unit--
131        ------------------
132        RequestData ::= SEQUENCE{
133            sid BIT STRING ('08'H),
134            requestedData DataTypeCode,
135            ...
136        }
137
138        -----------
139        --From VU--
140        -----------
141        RequestAccepted ::=SEQUENCE{
142            sid BIT STRING ('09'H),
143            trtp DataTypeCode,
144            dataSheet CHOICE{
145                standardData StandardTachDataContent,
146                personalData PersonalTachDataContent,
147                gnss GNSSDataContent,
148                standardEvent StandardEventContent,
149                personalEvent PersonalEventContent,
150                standardFault StandardFaultContent,
151                manufacturerdata ManufacturerDataContent,
152                ...
153            }
154        }
155
156        DataTypeCode ::=CHOICE{
157            standardTachData BIT STRING ('01'H),
158            personalTachData BIT STRING ('02'H),
159            gnssData BIT STRING ('03'H),
160            standardEventData BIT STRING ('04'H),
161            personalEventData BIT STRING ('05'H),
162            standardFaultData BIT STRING ('06'H),
163            manufacturerData BIT STRING ('07'H),
164            ...
165        }
166
167        DataUnavailable ::=SEQUENCE{
168            sid BIT STRING ('0A'H),
169            trtp DataTypeCode,
170            reason UnavailableDataCodes
171        }
172
173        UnavailableDataCodes ::= CHOICE{
174            noDataAvailable BIT STRING ('10'H),
175            personalDataNotShared BIT STRING ('11'H),
176            ...
177        }
178        --------------------------
179        --Complete Tachograph Data--
180        --------------------------
181        --The format of the data was taken from the ISO16844-7 norm, more information
182 available in this ISO document--
183
```

24

```
184          Time ::= SEQUENCE{
185                  seconds INTEGER (0..59.75), --increment: 0.25s--
186                  minutes INTEGER (0..59), --increment: 1min--
187                  hours INTEGER (0..23), --increment: 1h--
188                  day INTEGER (0.25.. 31.75), --increment: 0.25d--
189                  month INTEGER (1..12), --increment: 1month--
190                  year INTEGER (1985..2235), --increment: 1year--
191                  locMinOffset INTEGER (-59..59), --increment: 1min--
192                  locHouroffset INTEGER (-23..23)--increment: 1h--
193          }
194
195          Date ::= SEQUENCE{
196                  month INTEGER (1..12), --increment: 1month--
197                  day INTEGER (0.25.. 31.75), --increment: 0.25d--
198                  year INTEGER (1985..2235) --increment: 1year--
199          }
200
201          DriverName ::=SEQUENCE{
202                  codePageSurname UTF8String, --See ISO/IEC 8859--
203                  surname UTF8String,
204                  codePageFirstname UTF8String, --See ISO/IEC 8859--
205                  firstname UTF8String,
206          }
206a
206b         DriverID ::= SEQUENCE{
206c                 issuingMemberState OCTET STRING (SIZE(3)),
206d                 cardNumber OCTET STRING (SIZE(16))
206e         }
207
208          -------------------
209          --Message Content--
210          -------------------
211
212          StandardTachDataContent ::= SEQUENCE{
213                  trtp DataTypeCode (DataTypeCode.&standardTachData),
214                  personal BOOLEAN (FALSE),
215                  data StandardTachyDataSheet,
216          }
217
218          PersonalTachDataContent ::= SEQUENCE{
219                  trtp DataTypeCode (DataTypeCode.&personalTachData),
220                  personal BOOLEAN (TRUE),
221                  data PersonalTachyDataSheet
222          }
223
224          GNSSDataContent ::= SEQUENCE{
225                  trtp DataTypeCode (DataTypeCode.&gnssData),
226                  personal BOOLEAN (TRUE),
227                  data GNSSDataSheet
228          }
229
230          StandardEventContent ::= SEQUENCE{
231                  trtp DataTypeCode (DataTypeCode.&standardEventData),
232                  personal BOOLEAN (FALSE),
233                  data StandardEventDataSheet
234          }
235
236          PersonalEventContent ::= SEQUENCE{
237                  trtp DataTypeCode (DataTypeCode.&personalEventData),
```

25

```
238            personal BOOLEAN (TRUE),
239            data PersonalEventDataSheet
240        }
241
242     StandardFaultContent ::= SEQUENCE{
243            trtp DataTypeCode (DataTypeCode.&standardFaultData),
244            personal BOOLEAN (FALSE),
245            data StandardFault
246        }
247
248     ManufacturerDataContent ::= SEQUENCE{
249            trtp DataTypeCode (DataTypeCode.&manufacturerData),
250            personal BOOLEAN (TRUE),
251            ...
252        }
253
254     ---------------
255     --DATA SHEETS--
256     ---------------
257
258     --Data sheet format follows ISO 16844-7.--
259     StandardTachyDataSheet ::= SEQUENCE{
260            vin UTF8String (SIZE(17)),
261            calibrationDate Date,
262            driveRecognize BIT STRING ('00'B UNION '01'B),
263            driverCardDriver1 BIT STRING ('00'B UNION '01'B),
264            driverCardDriver2 BIT STRING ('00'B UNION '01'B),
265            timeDate Time,
266            highResolutionTotalVehicleDistance INTEGER (0..21055406), --increment:
267     5m--
268            serviceComponentIdentification INTEGER (0..255),
269            serviceDelayCalendarTimeBased INTEGER (-125..125), --increment: 1week-
270     -
271            nextCalibrationDate Date,
272            speedAuthorised INTEGER (0..250.996), --increment 1/256km/h--
273            tachographCardSlot1 INTEGER (0..4...), --Maximum 250--
274            tachographCardSlot2 INTEGER (0..4...), --Maximum 250--
275            outOfScopeCondition BIT STRING ('00'B UNION '01'B),
276            modeOfOperation INTEGER (0..4...), --Maximum 250--
277            registeringMemberState UTF8String,          vehicleRegistrationNumber
278     SEQUENCE {
279                codePageVRN INTEGER (0..255),
280                vrn OCTET STRING (SIZE(13)),
281            },
282            tachographNextMandatoryDownloadDate Date,
283            ...
284        }
285
286     PersonalTachyDataSheet ::= SEQUENCE{
287            tachographVehicleSpeed INTEGER (0..250.996), --increment 1/256km/h--
288            driver1WorkingState BIT STRING ('000'B UNION '001'B UNION '010'B UNION
289            '011'B UNION '100'B UNION '101'B ...),
290            driver2WorkingState BIT STRING ('000'B UNION '001'B UNION '010'B UNION
291            '001'B UNION '100'B UNION '101'B ...),
292
293            driver1TimeRelatedStates BIT STRING ('0000'B UNION '0001'B UNION
294            '0010'B UNION '0011'B UNION '0100'B UNION '0101'B UNION
295            '0110'B UNION '0111'B UNION '1000'B UNION '1001'B UNION '1010'B
296            UNION '1011'B UNION '1100'B UNION '1101'B ...),
```

```
297
298
299             driver2TimeRelatedStates BIT STRING ('0000'B UNION '0001'B UNION
300             '0010'B UNION '0011'B UNION '0100'B UNION '0101'B UNION
301             '0110'B UNION '0111'B UNION '1000'B UNION '1001'B UNION '1010'B
302             UNION '1011'B UNION '1100'B UNION '1101'B ...),
303
304
305
306             overSpeed BIT STRING ('00'B UNION '01'B),
307             driver1Identification DriverID
308             driver2Identification DriverID
309
310
311             driver1ContinuousDrivingTime INTEGER (0.. 64255), --increment: 1min--
312             driver2ContinuousDrivingTime INTEGER (0.. 64255), --increment: 1min--
313             driver1CurrentDurationOfSelectedActivity INTEGER (0.. 64255), --
314 increment: 1min--
315             driver2CurrentDurationOfSelectedActivity INTEGER (0.. 64255), --
316 increment: 1min--
317             driver1Name DriverName,
318             driver2Name DriverName,
319             driver1CumulatedDrivingTimePreviousAndCurrentWeek INTEGER (0.. 64255),
320 --increment: 1min--
321             driver2CumulatedDrivingTimePreviousAndCurrentWeek INTEGER (0.. 64255),
322 --increment: 1min--
323             engineSpeed INTEGER(0..8031.875), --increment: 0,125r/min--
324             driver1EndOfLastDailyRestPeriod Time,
325             driver2EndOfLastDailyRestPeriod Time,
326             driver1EndOfLastWeeklyRestPeriod Time,
327             driver2EndOfLastWeeklyRestPeriod Time,
328             driver1EndOfSecondLastWeeklyRestPeriod Time,
329             driver2EndOfSecondLastWeeklyRestPeriod Time,
330             driver1CurrentDailyDrivingTime INTEGER (0.. 64255), --increment: 1min-
331 -
332             driver2CurrentDailyDrivingTime INTEGER (0.. 64255), --increment: 1min-
333 -
334             driver1CurrentWeeklyDrivingTime INTEGER (0.. 64255), --increment:
335 1min--
336             driver2CurrentWeeklyDrivingTime INTEGER (0.. 64255), --increment:
337 1min--
338             driver1TimeLeftUntilNewDailyRestPeriod INTEGER (0.. 64255), --
339 increment: 1min--
340             driver2TimeLeftUntilNewDailyRestPeriod INTEGER (0.. 64255), --
341 increment: 1min--
342             driver1CardExpiryDate Date,
343             driver2CardExpiryDate Date,
344             driver1CardNextMandatoryDownloadDate Date,
345             driver2CardNextMandatoryDownloadDate Date,
346             driver1TimeLeftUntilNewWeeklyRestPeriod INTEGER (0.. 64255), --
347 increment: 1min--
348             driver2TimeLeftUntilNewWeeklyRestPeriod INTEGER (0.. 64255), --
349 increment: 1min--
350             driver1NumberOfTimes9hDailyDrivingTimesExceeded INTEGER (0..13),
351             driver2NumberOfTimes9hDailyDrivingTimesExceeced INTEGER (0..13),
352             driver1CumulativeUninterruptedRestTime INTEGER (0.. 64255), --
353 increment: 1min--
354             driver2CumulativeUninterruptedRestTime INTEGER (0.. 64255), --
355 increment: 1min--
```

27

```
356              driver1MinimumDailyRest INTEGER (0.. 64255), --increment: 1min--
357              driver2MinimumDailyRest INTEGER (0.. 64255), --increment: 1min--
358              driver1MinimumWeeklyRest INTEGER (0.. 64255), --increment: 1min--
359              driver2MinimumWeeklyRest INTEGER (0.. 64255), --increment: 1min--
360              driver1MaximumDailyPeriod INTEGER (0..250), --increment: 1h--
361              driver2MaximumDailyPeriod INTEGER (0..250), --increment: 1h--
362              driver1MaximumDailyDrivingTime BIT STRING (SIZE(4)),
363              driver2MaximumDailyDrivingTime BIT STRING (SIZE(4)),
364              driver1NumberOfUsedReducedDailyRestPeriods INTEGER (0..13),
365              driver2NumberOfUsedReducedDailyRestPeriods INTEGER (0..13),
366              driver1RemainingCurrentDrivingTime INTEGER (0.. 64255), --increment:
367  1min--
368              driver2RemainingCurrentDrivingTime INTEGER (0.. 64255), --increment:
369  1min--
370              ...
371          }
372
373      GNSSDataSheet ::=  SEQUENCE {
374      gnssPosition GeoCoordinates
375              --See Sub-appendix 1 for definition of GeoCoordinates--
376      }
377
378      StandardEventDataSheet ::= SEQUENCE{
379              events SEQUENCE OF StandardEvent
380      }
381
382      PersonalEventDataSheet ::= SEQUENCE{
383              events SEQUENCE OF PersonalEvent
384      }
385  END
386
387  EventsModule DEFINITIONS AUTOMATIC TAGS ::= BEGIN
388          EXPORTS ALL;
389          IMPORTS NationAlpha FROM Sub-appendix1; --See Sub-appendix 1 for more
390  information about NationAlpha--
391
392              SecurityBreachEvent ::=SEQUENCE{
393                  --See Appendix 1B for more information--
394              }
395
396              RecordingEquipmentFaultType ::= SEQUENCE{
397                  --See Appendix 1B for more information--
398              }
399
400              StandardEvent::= CHOICE{
401                  insertionInvalidCard InsertionOfANonValidCard,
402                  cardConflict CardConflict,
403                  timeOverlap TimeOverlap,
404                  previousSessionNotClosed LastCardSessionNotCorrectlyClosed,
405                  overSpeeding OverSpeeding,
406                  powerSupplyInterruption PowerSupplyInterruption,
407                  comErrorWithRemoteFacility
408  CommunicationErrorWithTheRemoteCommunicationFacility,
409                  absenceGNSSPosition
410  AbsenceOfPositionInformationFromGNSSReceiver,
410a                 comErrorWithExternalGNSSFacility
410b CommunicationErrorWithTheExternalGNSSFacility,
411                  positionDataError PositionDataError,
412                  motionDataError MotionDataError,
```

	Deleted: Appendix
	Deleted: Appendix
	Deleted: Appendix
	Deleted: Annex
	Deleted: Annex

28

```
418              vehicleMotionConflict VehicleMotionConflict,
419              securityBreachAttempt SecurityBreachAttempt,
420              timeConflict TimeConflict,
421              ...
422          }
423
424      PersonalEvent ::= CHOICE{
425              lackOfAppropriateCard DrivingWithoutAnAppropriateCard,
426              cardInsertionWhileDriving CardInsertionWhileDriving,
427              overSpeeding OverSpeeding,
428              ...
429          }
430
431      StandardFault ::= CHOICE{
432              cardFault CardFault,
433              recordingEquipementFault RecordingEquipmentFault,
434          ...
435          }
436
437          ---------------
438          --EVENTS LIST--
439          ---------------
440
441      InsertionOfANonValidCard::=SEQUENCE{
442              beginDate GeneralizedTime,
443              endDate GeneralizedTime,
444              carsdType SEQUENCE OF UTF8String,
445              cardsNumber SEQUENCE OF INTEGER,
446              issuingMemberState SEQUENCE OF NationAlpha,
447              cardsGeneration SEQUENCE OF INTEGER
448          }
449
450      CardConflict ::= SEQUENCE{
451              beginDate GeneralizedTime,
452              endDate GeneralizedTime,
453              carsdType SEQUENCE OF UTF8String,
454              cardsNumber SEQUENCE OF INTEGER,
455              issuingMemberState SEQUENCE OF NationAlpha,
456              cardsGeneration SEQUENCE OF INTEGER
457          }
458
459      TimeOverlap ::=SEQUENCE{
460              beginDate GeneralizedTime,
461              endDate GeneralizedTime,
462              carsdType SEQUENCE OF UTF8String,
463              cardsNumber SEQUENCE OF INTEGER,
464              issuingMemberState SEQUENCE OF NationAlpha,
465              cardsGeneration SEQUENCE OF INTEGER,
466              numberSimilarEvent INTEGER
467          }
468
469      DrivingWithoutAnAppropriateCard ::= SEQUENCE{
470              beginDate GeneralizedTime,
471              endDate GeneralizedTime,
472              carsdType SEQUENCE OF UTF8String,
473              cardsNumber SEQUENCE OF INTEGER,
474              issuingMemberState SEQUENCE OF NationAlpha,
475              cardsGeneration SEQUENCE OF INTEGER,
476              numberOfSimilarEvent INTEGER
```

29

```
477                      }
478
479              CardInsertionWhileDriving ::= SEQUENCE{
480                      date GeneralizedTime,
481                      carsdType SEQUENCE OF UTF8String,
482                      cardsNumber SEQUENCE OF INTEGER,
483                      issuingMemberState SEQUENCE OF NationAlpha,
484                      numberOfSimilarEvents INTEGER
485              }
486
487              LastCardSessionNotCorrectlyClosed ::=SEQUENCE{
488                      beginDate GeneralizedTime,
489                      endDate GeneralizedTime,
490                      carsdType SEQUENCE OF UTF8String,
491                      cardsNumber SEQUENCE OF INTEGER,
492                      issuingMemberState SEQUENCE OF NationAlpha,
493                      cardsGeneration SEQUENCE OF INTEGER,
494                      oldSession SEQUENCE{
495                              beginDate GeneralizedTime,
496                              endDate GeneralizedTime,
497                              vrn UTF8String,
498                              issuingMemberState NationAlpha,
499                              cardsGeneration INTEGER,
500                      }
501              }
502
503              OverSpeeding ::=SEQUENCE{
504                      beginDate GeneralizedTime,
505                      endDate GeneralizedTime,
506                      maximumSpeed INTEGER,
507                      averageSpeed INTEGER,
508                      cardType UTF8String,
509                      cardNumber INTEGER,
510                      issuingMemberState NationAlpha,
511                      cardGeneration INTEGER,
512                      numberOfSimilarEvents INTEGER
513              }
514
515              PowerSupplyInterruption ::=SEQUENCE{
516                      beginDate GeneralizedTime,
517                      endDate GeneralizedTime,
518                      carsdType SEQUENCE OF UTF8String,
519                      cardsNumber SEQUENCE OF INTEGER,
520                      issuingMemberState SEQUENCE OF NationAlpha,
521                      cardsGeneration SEQUENCE OF INTEGER,
522                      numberOfSimilarEvent INTEGER
523              }
524
525              CommunicationErrorWithTheRemoteCommunicationFacility ::=SEQUENCE{
526                      beginDate GeneralizedTime,
527                      endDate GeneralizedTime,
528                      carsdType SEQUENCE OF UTF8String,
529                      cardsNumber SEQUENCE OF INTEGER,
530                      issuingMemberState SEQUENCE OF NationAlpha,
531                      cardsGeneration SEQUENCE OF INTEGER,
532                      numberOfSimilarEvent INTEGER
533              }
534
535              AbsenceOfPositionInformationFromGNSSReceiver ::= SEQUENCE{
```

```
536                    beginDate GeneralizedTime,
537                    endDate GeneralizedTime,
538                    carsdType SEQUENCE OF UTF8String,
539                    cardsNumber SEQUENCE OF INTEGER,
540                    issuingMemberState SEQUENCE OF NationAlpha,
541                    cardsGeneration SEQUENCE OF INTEGER,
542                    numberOfSimilarEvent INTEGER
543            }
544
539a         CommunicationErrorWithTheExternalGNSSFacility ::= SEQUENCE{
539b                   beginDate GeneralizedTime,
539c                   endDate GeneralizedTime,
539d                   carsdType SEQUENCE OF UTF8String,
539e                   cardsNumber SEQUENCE OF INTEGER,
539f                   issuingMemberState SEQUENCE OF NationAlpha,
539g                   cardsGeneration SEQUENCE OF INTEGER,
539h                   numberOfSimilarEvent INTEGER
539i           }
539j
545          PositionDataError ::= SEQUENCE{
546                    beginDate GeneralizedTime,
547                    endDate GeneralizedTime,
548                    carsdType SEQUENCE OF UTF8String,
549                    cardsNumber SEQUENCE OF INTEGER,
550                    issuingMemberState SEQUENCE OF NationAlpha,
551                    cardsGeneration SEQUENCE OF INTEGER,
552                    numberOfSimilarEvent INTEGER
553            }
554
555          MotionDataError ::= SEQUENCE{
556                    beginDate GeneralizedTime,
557                    endDate GeneralizedTime,
558                    carsdType SEQUENCE OF UTF8String,
559                    cardsNumber SEQUENCE OF INTEGER,
560                    issuingMemberState SEQUENCE OF NationAlpha,
561                    cardsGeneration SEQUENCE OF INTEGER,
562                    numberOfSimilarEvent INTEGER
563            }
564
565          VehicleMotionConflict ::= SEQUENCE{
566                    beginDate GeneralizedTime,
567                    endDate GeneralizedTime,
568                    carsdType SEQUENCE OF UTF8String,
569                    cardsNumber SEQUENCE OF INTEGER,
570                    issuingMemberState SEQUENCE OF NationAlpha,
571                    cardsGeneration SEQUENCE OF INTEGER,
572                    numberOfSimilarEvent INTEGER
573            }
574
575          SecurityBreachAttempt ::= SEQUENCE{
576                    beginDate GeneralizedTime,
577                    endDate GeneralizedTime OPTIONAL,
578                    carsdType SEQUENCE OF UTF8String,
579                    cardsNumber SEQUENCE OF INTEGER,
580                    issuingMemberState SEQUENCE OF NationAlpha,
581                    numberOfSimilarEvent INTEGER,
582                    typeOfEvent SecurityBreachEvent
583            }
584
```

```
585
586        TimeConflict ::= SEQUENCE{
587            beginDate GeneralizedTime,
588            endDate GeneralizedTime,
589            carsdType SEQUENCE OF UTF8String,
590            cardsNumber SEQUENCE OF INTEGER,
591            issuingMemberState SEQUENCE OF NationAlpha,
592            cardsGeneration SEQUENCE OF INTEGER,
593            numberOfSimilarEvent INTEGER
594        }
595
596        ---------------
597        --FAULTS LIST--
598        ---------------
599
600        CardFault ::= SEQUENCE{
601            beginDate GeneralizedTime,
602            endDate GeneralizedTime,
603            carsdType SEQUENCE OF UTF8String,
604            cardsNumber SEQUENCE OF INTEGER,
605            issuingMemberState SEQUENCE OF NationAlpha,
606            cardsGeneration SEQUENCE OF INTEGER,
607        }
608
609        RecordingEquipmentFault ::= SEQUENCE{
610            beginDate GeneralizedTime,
611            endDate GeneralizedTime,
612            faultType RecordingEquipmentFaultType,
613            carsdType SEQUENCE OF UTF8String,
614            cardsNumber SEQUENCE OF INTEGER,
615            issuingMemberState SEQUENCE OF NationAlpha,
616            cardsGeneration SEQUENCE OF INTEGER,
617        }
618    END
```

32