TACHO·SYS DIGITAL TACHOGRAPH SOLUTIONS

Data Packet Forwarding from digiCentral.

Document Change Record

Revision	Date	Owner	Details
0.1	13th May 2015	James Scott-Evans	Initial Draft.
0.2	20th May 2015	James Scott-Evans	Alteration to New Driver notification.
0.3	12th June 2015	James Scott-Evans	Revision to SOAP packet - compatible for both WCF & ASMX.
0.4	26th June 2015	James Scott-Evans	Extension to Tracking packet. Alteration to Sensors.
0.5	1st Sept 2015	James Scott-Evans	Different Sensor packets.
0.6	11th Sept 2015	James Scott-Evans	Extension to FMS Engine Off.
1.0	28th Jan 2016	James Scott-Evans	Clocking packet added.
1.1	4th Feb 2016	James Scott-Evans	1-Wire Temperature Probes.
1.2	18th Apr 2016	James Scott-Evans	Clocking includes missing Licence Number info.
1.3	31st May 2016	Warren Townsend	Odometer Logging.
1.4	17th Jun 2016	James Scott-Evans	Custom Tags.
1.5	25th Jul 2016	James Scott-Evans	Extending Stoneridge DDS with Working Time values.
1.6	9th Nov 2016	James Scott-Evans	Packet Type table
1.7	14th Nov 2016	Ian Frost	Value detail added for 1-Wire and Accelerometer
2.00	22nd Nov 2016	James Scott-Evans	New Binary Mode, new Tracking packets
2.01	24th Jan 2017	James Scott-Evans	Consolidated DDS packet
2.02	6th Apr 2017	James Scott-Evans	REST web service delivered JSON packet
2.03	9th Nov 2017	James Scott-Evans	Time source comparison. Reduced Tachograph Mode
2.04	19th Mar 2018	James Scott-Evans	Passenger count sensors.
2.05	11th Apr 2018	James Scott-Evans	VDO Counter Daily Weekly Drive notes.
2.06	21st Aug 2018	James Scott-Evans	Alerts added.
2.07	18th Nov 2019	James Scott-Evans	New Clocking packet
2.08	6th Feb 2020	James Scott-Evans	New File Summary packet
2.09	16th Apr 2020	James Scott-Evans	Packet filtering
2.10	25th Jan 2021	James Scott-Evans	New Intellic DDS packet

Table of Contents

1. Introduction

In 2008 Tachosys developed the digiCentral platform to support remote data collection from any Tachosys device.

With the advent of the digital tachograph remote download capability, we extended the functionality of digiCentral to include remote company card authentication.

Now with digiDLe/ex devices collecting Tracking, Vehicle CAN (SAE J1939) & Sensor data, we expect digiCentral will be forwarding more data onto third parties.

The purpose of this document is to outline the information that can be forwarded from the digiCentral platform to third party solutions.

2. References

2.1. Annex 1B

The legislation that introduced the digital tachograph:-
http://eur-lex.europa.eu/legal-content/EN/ALL/?uri=OJ:L:2002:207:TOC

2.2. Annex 1C

The new legislation for Smart Tachographs:-
http://eur-lex.europa.eu/legal-content/EN/TXT/?qid=1468399756621&uri=CELEX:32016R0799

2.3. Amendments to Annex 1C

These amendments are quite extensive and have to be followed alongside the Annex 1C document:-
http://eur-lex.europa.eu/legal-content/EN/TXT/?qid=1524124831090&uri=CELEX:32018R0502

2.4. ISO 16844-7

Road vehicles –Tachograph systems. Part 7: Parameters.

2.5. Driver Decision Support Data via CAN

Stoneridge Document No. 1231/001-900208.

2.6. FMS-Standard

FMS-Standard description – Version 3 14/09/2012.

3. Packet Structure

The data can be forwarded using one of the following methods. The first method sends a JSON formatted structure to a RESTful web service. The second option is using SOAP calls to a web service. The third is by posting an XML document to a TCP or UDP socket. The final method sends a binary packet to a TCP socket and requires a reply.

The `placeholders` will be replaced with the actual values defined as follows:-

Field	Data type	Size	Description
deviceSerialNumber	string	8	Serial number of the device used.
vehicleRegistrationNumber	string	13	VehicleRegistrationNumber (Annex 1B)
data	binary		Packet payload.
customTags	string		Additional data added to every packet.

Note: vehicleRegistrationNumber will be blank for some packet types.

3.1. REST

The data packet will be sent using a HTTP POST request. The body will contain the information to be sent in a JSON structure as shown below:-

```
{ deviceSerialNumber : "string" ,
  vehicleRegistrationNumber : "string" ,
  data : "base64binary/string" ,
  customTags
}
```

A response of 200 OK or 204 No Content will indicate success.

3.2. SOAP

SOAP version 1.1 will be used so the complete packet will look as follows:-

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <soap:Body>
        <digiCentralDataPacket xmlns="digiCentral.Website">
            <deviceSerialNumber>string</deviceSerialNumber>
            <vehicleRegistrationNumber>string</vehicleRegistrationNumber>
            <data>base64binary/string</data>
            customTags
        </digiCentralDataPacket>
    </soap:Body>
</soap:Envelope>
```

The SOAPAction header will be:- `digiCentral.Website/digiCentralDataPacket`

The Packet will be sent using a HTTP Web Request method so will expect a 200 OK response to indicate success.

3.3. XML via TCP or UDP

The XML document is defined as follows:-

```
<digiCentralDataPacket>
  <deviceSerialNumber>string</deviceSerialNumber>
  <vehicleRegistrationNumber>string</vehicleRegistrationNumber>
  <data>base64binary/string</data>
  customTags
</digiCentralDataPacket>
```

A socket will be opened for the broadcasting of each document. No reply will be used to indicate success so the packet will be assumed to be sent successfully unless the socket connection fails.

3.4. Binary mode via TCP

The binary packet will be sent using the following message structure:-

Section	Size	Notes
Header	1 byte	Header byte will be 0x55.
Identifier	1 byte	Identifier byte will be 0xD7.
Length of overall packet contents	2 bytes	Excludes Header, Identifier, Checksum and this section.
Device serial number	8 bytes	7 bit ASCII encoded.
Vehicle registration number	13 bytes	7 bit ASCII encoded.
Length of packet payload	1 byte	
Packet payload		
Length of customTags	1 byte	
customTags		7 bit ASCII encoded.
Checksum	1 byte	The checksum byte is the 8 bit sum series modulo 256 of all the bytes of the message excluding itself.

A positive reply will be sent using the following message structure:-

Section	Size	Notes
Header	1 byte	Header byte will be 0x55.
Identifier	1 byte	Identifier byte will be 0xA0.
Length of overall packet	2 bytes	In this case there is zero length so the bytes will be: 0x00 0x00
Checksum	1 byte	The checksum byte will be 0xF5.

A negative reply will be sent using the following message structure:-

Section	Size	Notes
Header	1 byte	Header byte will be 0x55.
Identifier	1 byte	Identifier byte will be 0xAE.
Length of overall packet	2 bytes	
Error code	2 bytes	
Error message		7 bit ASCII encoded.
Checksum	1 byte	The checksum byte is the 8 bit sum series modulo 256 of all the bytes of the message excluding itself.

A reply will be expected within 20 seconds.

3.5. Custom tags

Additional data can be added to every packet. These elements can be used to enable the receiver to identify the source of the packet. The whole text to be sent is stored with the digiCentral Data Forwarding configuration.

For example, if we wanted to send an account id within the packet we would replace `customTags` with:-

```
<accountID>1234</accountID>
```

This would be suitable for sending via the SOAP and XML methods.

3.6. Error handling

When an unsuccessful attempt to send a packet occurs, all forwarding to that destination will stop. The first retry will occur after 1 minute has passed and each subsequent failure will cause the interval between retries to increase to a maximum of 1 hour.

4. Packet types

The first byte of the data is used to identify the packet. This data can be sent using base64 binary or ASCII Hex. The size of the packet will change with each type. Implementers will need to allow for the size of the data to grow.

The following table lists all the different packet types.

Type	Subtype	Name	Document section
0x41	0x00	Alcohol lock	
0x43	0x00	CAN Logging	
0x44	0x00	New Driver notification	8
0x46	0x80	Vehicle CAN data - FMS Initialisation	15.1
0x46	0x81	Vehicle CAN data - Engine on	15.2
0x46	0x82	Vehicle CAN data - In journey	15.4
0x46	0x83	Vehicle CAN data - Engine off	15.3
0x46	0x84	Vehicle CAN data - On change	15.4
0x47	0x00	Tracking	12
0x47	0x01	Positional update	12.1
0x47	0x02	Tracking suppression	12.2
0x49	0x00	Time source comparison	10
0x4f	--	Odometer	17
0x53	--	Sensors	16
0x61	0x00	Alerts	7
0x64	0x00	Device configuration	
0x66	0x00	Driver Card File	9.1
0x66	0x01	Workshop Card File	9.2
0x66	0x02	Vehicle Unit File	9.3
0x6B	--	Clocking	11
0x4D or 0x6D	0x00	Tachograph mode	13
0x4D or 0x6D	0x02	Stoneridge Duo	14.1
0x4D or 0x6D	0x03	VDO Counter	14.2
0x4D or 0x6D	0x04	Intellic iCounter	14.3
0x4D	0x0C	Consolidated DDS	14.4

5. Packet filtering

A filter can be put in place to restrict the packets that forwarded. This can help reduce the load on the receiving server.

Even with a filter in place implementers need to accept all packet types forwarded and allow for new packets types to be introduced in the future. But they only need to process those that they are interested in. Those that are ignored still need to have a success reply.

6. digiCentral Configuration

Within digiCentral the setup of Data Forwarding is accessed via the **Forwarding** tab and then selecting **Data forwarding** from the menu.

An example of the setup required for REST forwarding is shown in Fig. 1 and Fig. 2 shows an example for SOAP. Please note that the protocol identifier http:// or https:// is required as part of the URL for the Host Name.

Fig. 3 shows an example for the setup for XML via TCP or UDP.
Fig. 4 shows the setup for the Binary Mode method of forwarding.

All the examples show an example for the custom tags text. If this is not required, it can be left blank.

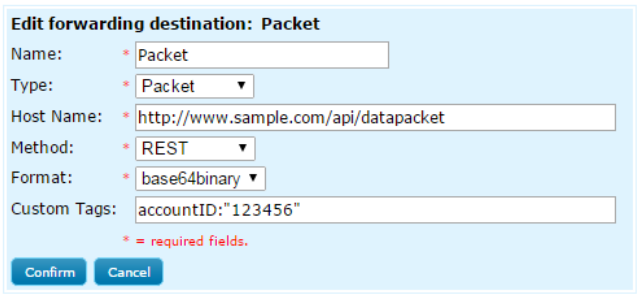

Figure 1

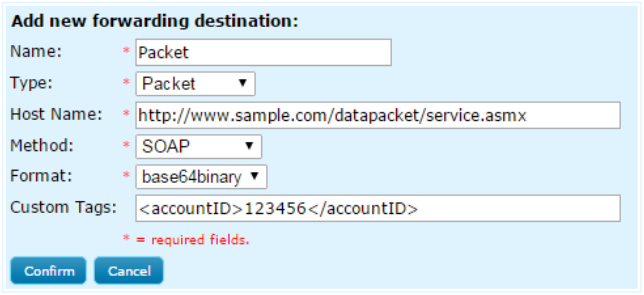

Figure 2

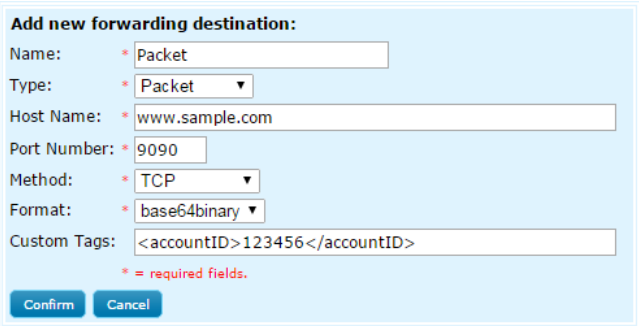

Figure 3

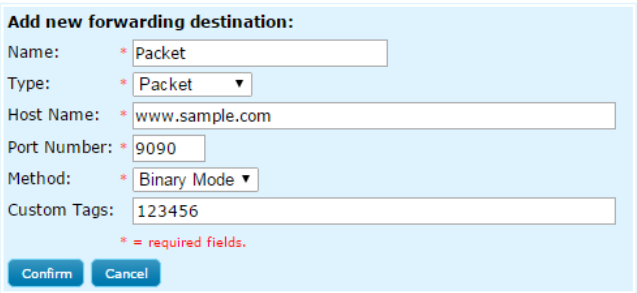

Figure 4

A filter can be applied to restrict which packets are sent to the specified endpoint. Tick the Use filter option and select which packets to send to the endpoint as shown in Fig. 5 below.

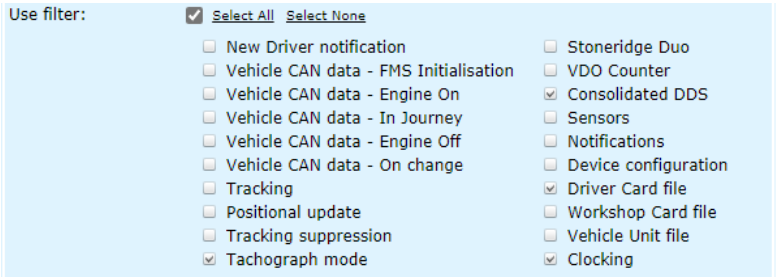

Figure 5

Once a Data forwarding destination has been created it needs to be set against a Location. All the devices within this location will queue data packets to this destination. A location can be set to forward data to more than one destination. Fig. 6 shows that for the Location called **Depot**, Data packets will be sent to **Sample Destination**.

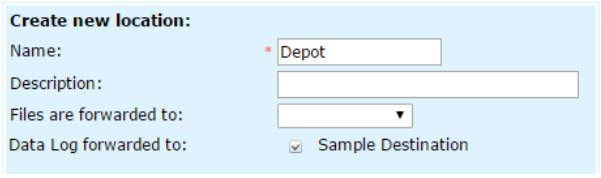

Figure 6

7. Alerts

The digiCentral platform will raise a number of different alerts triggered by devices or situations arising with a driver or vehicle. This notification can be used to import these alerts into an external system. The text of the alert can be localised using the **Data Forwarding** settings.

Offset	Field	Datatype	Size	Notes
0	Type	byte	1	0x61
1	Subtype	byte	1	0x00
2	CardNumber	string	14	CardNumber.driverIdentification (Annex 1B)
16	AlertState	byte	1	0 - clear, 1 - alert, 2 - info
17	AlertType	byte	1	Used to identify the alert.
18	AlertSubType	byte	1	Used to identify the alert.
19	AlertDatetime	TimeReal	4	TimeReal (Annex 1B)
23	Info	byte		NULL char separated alert info

Below is a table listing the order of the info items.

Field	Datatype	Notes
Title	string	The localised description of the type of alert.
Alert	string	The localised text of the alert.
P1	string	Parameters for the alert
P2	string	''
P3	string	''
P4	string	''
P5	string	''
P6	string	''
P7	string	''
P8	string	''
P9	string	''
P10	string	''

Notes:
When alerts are raised that are not in the context of a device, the serial number of **`00000000`** will be used.

8. New Driver notification

This notification is used in order that external systems are made aware of Drivers new to the system. This enables key Driver records to be created before Tachograph Mode or Driver Decision Support data is received.

Offset	Field	Datatype	Size	Notes
0	Type	byte	1	0x44
1	Subtype	byte	1	0x00
2	CardNumber	string	14	CardNumber.driverIdentification (Annex 1B)
16	Surname	string	70	Unicode
86	FirstNames	string	70	Unicode

9. File Summary

These packets are triggered when digiCentral receives a Driver Card, Workshop Card, or Vehicle Unit file. The packet is made up from information held within the file and additionally includes the date and time that the download was taken.

9.1. Driver Card

Offset	Field	Datatype	Size	Notes
0	Type	byte	1	0x66
1	Subtype	byte	1	0x00
2	DownloadDate	TimeReal	4	TimeReal (Annex 1B)
6	CardNumber	string	14	CardNumber.driverIdentification (Annex 1B)
20	Surname	string	70	Unicode
90	FirstNames	string	70	Unicode
160	cardHolderBirthDate	Datef	4	Datef (Annex 1B)
164	cardValidBegin	TimeReal	4	TimeReal (Annex 1B)
168	cardExpiryDate	TimeReal	4	TimeReal (Annex 1B)
172	cardIssuingMemberStateAlpha	string	3	NationAlpha (Annex 1B)
175	drivingLicenceNumber	string	16	CardDrivingLicenceInformation.drivingLicenceNumber (Annex 1B)
191	activityStartDate	TimeReal	4	TimeReal (Annex 1B)
195	activityEndDate	TimeReal	4	TimeReal (Annex 1B)

9.2. Workshop Card

Offset	Field	Datatype	Size	Notes
0	Type	byte	1	0x66
1	Subtype	byte	1	0x01
2	DownloadDate	TimeReal	4	TimeReal (Annex 1B)
6	CardNumber	string	14	CardNumber.driverIdentification (Annex 1B)
20	Surname	string	70	Unicode
90	FirstNames	string	70	Unicode
160	cardValidBegin	TimeReal	4	TimeReal (Annex 1B)
164	cardExpiryDate	TimeReal	4	TimeReal (Annex 1B)
168	cardIssuingMemberStateAlpha	string	3	NationAlpha (Annex 1B)
171	workshopName	string	70	Unicode
241	workshopAddress	string	70	Unicode

9.3. Vehicle Unit

Offset	Field	Datatype	Size	Notes
0	Type	byte	1	0x66
1	Subtype	byte	1	0x02
2	DownloadDate	TimeReal	4	TimeReal (Annex 1B)
6	VIN	string	17	VehicleIdentificationNumber (Annex 1B)
23	vehicleRegistrationNationAlpha	string	3	NationAlpha (Annex 1B)
26	vehicleRegistrationNumber	string	26	Unicode
52	vuManufacturerName	string	70	Unicode
122	vuPartNumber	string	16	VUIdentification.vuPartNumber (Annex 1B)
138	vuSoftwareVersion	string	4	VUIdentification.vuSoftwareIdentification (Annex 1B)
142	authorisedSpeed	byte	1	VUCalibrationRecord.authorisedSpeed
143	nextCalibrationDate	TimeReal	4	VUCalibrationRecord.nextCalibrationDate
147	activityStartDate	TimeReal	4	TimeReal (Annex 1B)
151	activityEndDate	TimeReal	4	TimeReal (Annex 1B)

10. Time source comparison

So that the time source for a device, usually the tachograph, can be checked against UTC, it will generate this log entry once a day or after reboot.

Offset	Field	Datatype	Size	Notes
0	Type	byte	1	0x49
1	Subtype	byte	1	0x00
2	Time Source	TimeReal	4	TimeReal (Annex 1B)
6	Source	byte	1	Indicates which source is in use by the device.
7	UTC	TimeReal	4	TimeReal (Annex 1B)

11. Clocking

Either Tachograph Driver cards or special Clocking cards can be used for Clocking in/out using a digipostpro. Optionally, the user can be allowed to adjust the values of Drive, Work, Available and Rest recorded when they Clock out.

11.1. Packets from the original digipostpro

Offset	Field	Datatype	Size	Notes
0	Type	byte	1	0x6B
1	Subtype	byte	1	0x00
2	[1]DateTime	TimeReal	4	TimeReal (Annex 1B)
6	CardNumber	string	14	CardNumber.driverIdentification (Annex 1B)
20	DrivingLicenceNumber	string	16	CardDrivingLicenceInformation.drivingLicenceNumber (Annex 1B)
36	ClockState	byte	1	1 = OUT, 2 = IN
37	RecordedRest	uint	2	Number of minutes duration
39	RecordedAvailable	uint	2	"
41	RecordedWork	uint	2	"
43	RecordedDrive	uint	2	"
45	AdjustedRest	uint	2	"
47	AdjustedAvailable	uint	2	"
49	AdjustedWork	uint	2	"
51	AdjustedDrive	uint	2	"

Notes:

[1]DateTime will be in the local time of the digipostpro.

11.2. Packets from the new digipostpro

Offset	Field	Datatype	Size	Notes
0	Type	byte	1	0x6B
1	Subtype	byte	1	0x01
2	[1]DateTime	TimeReal	4	TimeReal (Annex 1B)
6	[2]CardNumber	string	14	
20	[2]UserReference	string	16	
36	ClockState	byte	1	1 = OUT, 2 = IN, 3 = BREAK start, 4 = BREAK end
37	*Reserved*		16	
53	FingerprintState	byte	1	0 = Matched 1 = Unavailable 2 = Failed 3 = New fingerprint registered
54	Surname	string	70	Unicode
124	FirstNames	string	70	Unicode

Notes:

[1]DateTime will be UTC.

[2]When a Driver Card is used when Clocking, CardNumber and UserReference will be taken from the card as CardNumber.driverIdentification and CardDrivingLicenceInformation.drivingLicenceNumber respectively.

12. Tracking

GPS data can be collected by enabled devices and sent to digiCentral. By default, the data is sent once a minute while the vehicle is moving. If the vehicle changes heading significantly, starts or stops, then extra tracking points are sent. If the vehicle is not moving, its position is sent to the server once an hour. The shortest interval between tracking points is about five seconds. We require at least 4 satellites to take a GPS fix.

Offset	Field	Datatype	Size	Notes
0	Type	byte	1	0x47
1	Subtype	byte	1	0x00
2	DateTime	TimeReal	4	TimeReal (Annex 1B)
6	Latitude	int	4	n / 600000 = decimal degrees
10	Longitude	int	4	n / 600000 = decimal degrees
14	Altitude	int	2	metres
16	Speed	int	2	n / 10 = knots
18	Heading	int	2	n / 10 = decimal degrees
20	*Reserved*		*2*	
22	Max Speed	int	2	n / 100 = m/s
24	Minimum Acceleration	int	2	n / 100 = m/s²
26	Maximum Acceleration	int	2	n / 100 = m/s²
28	Duration	uint	2	seconds
30	Distance	uint	4	n / 100 = metres
34	[1]Satellites	byte	1	Number of satellites in use
35	[1]*Reserved*		*1*	
36	[1]GPS DateTime	TimeReal	4	TimeReal (Annex 1B)

Notes:

[1]These bytes are not returned by digiDL devices. Implementers will need to check the length of the packet.

12.1. Positional update

This packet will be sent preceding any other packet type. e.g. before the FMS Engine Off packet as the driver turns off the ignition. This packet is not returned by the digiDL device.

Offset	Field	Datatype	Size	Notes
0	Type	byte	1	0x47
1	Subtype	byte	1	0x01
2	DateTime	TimeReal	4	TimeReal (Annex 1B)
6	Latitude	int	4	n / 600000 = decimal degrees
10	Longitude	int	4	n / 600000 = decimal degrees
14	Altitude	int	2	metres
16	Speed	int	2	n / 10 = knots
18	Heading	int	2	n / 10 = decimal degrees
20	GPS DateTime	TimeReal	4	TimeReal (Annex 1B)

12.2. Tracking suppression

This ON packet is sent when the driver chooses to suppress Tracking by using the in-cab button. The OFF packet is sent either when the driver removes the suppression using the button or when the vehicle is stationary for over 15 minutes.

Offset	Field	Datatype	Size	Notes
0	Type	byte	1	0x47
1	Subtype	byte	1	0x02
2	DateTime	TimeReal	4	TimeReal (Annex 1B)
6	Activated	byte	1	1 = ON, 0 = OFF

13. Tachograph Mode

This data can be used to monitor the activities of the Driver and Co-driver. However, the data must be passed through the algorithms detailed in Annex 1B before being used to calculate total driving hours.
Packets are sent on every mode change, card insert/withdrawal or when the device is first powered up.
There is a “reduced” function available that will only send packets while a driver card is not inserted in Slot 1 and send packets when the value of Slot 1 and Slot 2 CardNumber changes.

Offset	Field	Datatype	Size	Notes
0	Type	byte	1	0x4D or 0x6D
1	Subtype	byte	1	0x00
2	Activity Start	TimeReal	4	TimeReal (Annex 1B).
6	Slot 1 CardNumber	string	14	CardNumber.driverIdentification (Annex 1B)
20	Slot 1 Mode	byte	1	ActivityChangeInfo.Activity (Annex 1B)
21	Slot 2 CardNumber	string	14	CardNumber.driverIdentification (Annex 1B)
35	Slot 2 Mode	byte	1	ActivityChangeInfo.Activity (Annex 1B)
36	Odometer	uint	4	n * 0.005 = kilometres

Notes:
On older tachograph when the ignition is turned off this data is no longer available and so mode changes may be missed. This is the case on Stoneridge up to and including 7.3 and VDO up to and including 1.4.

14. Driver Decision Support

This data can be used to monitor the driving hours of the Driver. The calculations are done by the Tachograph and therefore the limitations published by the tachograph manufacturer for each release of their tachograph should be noted.
DDS data is sent to the server on a regular poll while a Driver Card is in slot 1 of the Tachograph. The poll interval can be set per device.

Further information can be found in our "Guide to Driver Decision Support Compatibility" document.

14.1. Stoneridge Duo

Offset	Field	Datatype	Size	Notes
0	Type	byte	1	0x4D or 0x6D
1	Subtype	byte	1	0x02
2	[A]DateTime	TimeReal	4	TimeReal (Annex 1B)
6	[A]CardNumber	string	14	CardNumber.driverIdentification (Annex 1B)
20	[A]Mode	byte	1	ActivityChangeInfo.Activity (Annex 1B)
21	[A]Duration	[4]uint	2	Number of minutes duration
23	[A]TimeRelatedStates	byte	1	See 14.5 Time Related States below
24	[A]ContinuousDrive	[4]uint	2	Number of minutes duration
26	[B]DailyDrive	[4]uint	2	"
28	[B]WeeklyDrive	[4]uint	2	"
30	[A]TwoWeekDrive	[4]uint	2	"
32	[A]CumulativeBreak	[4]uint	2	"
34	[B]CumulativeRest	[4]uint	2	"
36	[B2]TimeLeftUntilNextDailyRest	[4]uint	2	Number of minutes remaining before rest due
38	[B2]TimeLeftUntilNextWeeklyRest	[4]uint	2	"
40	[B2]MinimumDurationOfNextDailyRest	[4]uint	2	Number of minutes duration
42	[B2]MinimumDurationOfNextWeeklyRest	[4]uint	2	"
44	[B2]MaximumDailyDrive	[4]uint	1	Number of hours duration
45	[B2]ExtendedDriveUsed	[4]uint	1	Number of extended drive periods used in week
46	[B2]ReducedRestUsed	[4]uint	1	Number of reduced rests used in week
47	[C2 3]WorkingTimeDaily	[4]unit	2	Number of minutes duration
49	[C2 3]WorkingTimeWeekly	[4]unit	2	"
51	[C2 3]WorkingTimeBreak	[4]unit	2	"

Notes:
[A]Supported by all versions of Stoneridge tachograph.
[B]Requires at least version 7.4.
[C]Requires at least version 7.5.
[2]Requires DDS to be enabled within the Tachograph
[3]Requires latest firmware of digiDL 1.50 or digiDLe/ex 1.18.
[4]Values of 0xFF or 0xFFFF should be treated as NULL.

14.2. VDO Counter

Offset	Field	Datatype	Size	Notes
0	Type	byte	1	0x4D or 0x6D
1	Subtype	byte	1	0x03
2	[1]DateTime	TimeReal	4	TimeReal (Annex 1B)
6	[1]CardNumber	string	14	CardNumber.driverIdentification (Annex 1B)
20	[1]Mode	byte	1	ActivityChangeInfo.Activity (Annex 1B)
21	[1]Duration	[4]uint	2	Number of minutes duration
23	[1]TimeRelatedStates	byte	1	See 14.5 Time Related States below
24	[1]ContinuousDrive	[4]uint	2	Number of minutes duration
26	[2][5]DailyDrive	[4]uint	2	"
28	[2][5]WeeklyDrive	[4]uint	2	"
30	[1]TwoWeekDrive	[4]uint	2	"
32	[1]CumulativeBreak	[4]uint	2	"
34	[3]RemainingDrive	[4]uint	2	Number of minutes remaining
36	[3]TimeLeftUntilNextRestPeriod	[4]uint	2	"
38	[3]DurationOfNextBreak/Rest	[4]uint	2	Number of minutes duration
40	[3]RemainingBreak/Rest	[4]uint	2	Number of minutes remaining
42	[3]TimeLeftUntilNextDrivingPeriod	[4]uint	2	"
44	[3]MaximumDurationOfNextDrivingPeriod	[4]uint	2	Number of minutes duration
46	[3]RemainingDailyDrive	[4]uint	2	Number of minutes remaining
48	[3]TimeLeftUntilNextDailyRest	[4]uint	2	"
50	[3]MinimumDurationOfNextDailyRest	[4]uint	2	Number of minutes duration
52	[3]RemainingWeeklyDrive	[4]uint	2	Number of minutes remaining
54	[3]TimeLeftUntilNextWeeklyRest	[4]uint	2	"
56	[3]MinimumDurationOfNextWeeklyRest	[4]uint	2	Number of minutes duration
58	[3]BitMask	binary	2	uuuu uuuu uuur rudd u – unused r – reduced rest available d – extended drive available

Notes:

[1]Supported by all versions of VDO tachograph.

[2]Requires at least version 2.1.

[3]Requires VDO Counter to be turned on.

[4]Values of 0xFFFF should be treated as NULL.

[5]These values are not output by some models.

14.3. Intellic iCounter

Offset	Field	Datatype	Size	Notes
0	Type	byte	1	0x4D or 0x6D
1	Subtype	byte	1	0x02
2	DateTime	TimeReal	4	TimeReal (Annex 1B)
6	CardNumber	string	14	CardNumber.driverIdentification (Annex 1B)
20	Mode	byte	1	ActivityChangeInfo.Activity (Annex 1B)
21	Duration	[4]uint	2	Number of minutes duration
23	TimeRelatedStates	byte	1	See 14.5 Time Related States below
24	ContinuousDrive	[4]uint	2	Number of minutes duration
26	DailyDrive	[4]uint	2	"
28	WeeklyDrive	[4]uint	2	"
30	TwoWeekDrive	[4]uint	2	"
32	CumulativeBreak	[4]uint	2	"
34	TimeLeftUntilNextDailyRest	[4]uint	2	Number of minutes remaining before rest due
36	TimeLeftUntilNextWeeklyRest	[4]uint	2	"
38	MinimumDurationOfNextDailyRest	[4]uint	2	Number of minutes duration
40	MinimumDurationOfNextWeeklyRest	[4]uint	2	"
42	ExtendedDriveUsed	[4]uint	1	Number of extended drive periods used in week

Notes:

[4]Values of 0xFF or 0xFFFF should be treated as NULL.

14.4. Consolidated DDS

The digiCentral website use the Stoneridge and VDO feeds and combines them into a single view. This packet is generated by digiCentral so that third party solutions can choose to replicate this view. This packet will be sent immediately after either the Stoneridge Duo or VDO Counter packets.

Offset	Field	Datatype	Size	Notes
0	Type	byte	1	0x4D
1	Subtype	byte	1	0x0C
2	DateTime	TimeReal	4	TimeReal (Annex 1B)
6	CardNumber	string	14	CardNumber.driverIdentification (Annex 1B)
20	Mode	byte	1	ActivityChangeInfo.Activity (Annex 1B)
21	Duration	uint	2	Number of minutes duration
23	TimeRelatedStates	byte	1	See 14.5 Time Related States below
24	TimeRelatedLevel	byte	1	"
25	ContinuousDrive	uint	2	Number of minutes duration
27	DailyDrive	uint	2	"
29	WeeklyDrive	uint	2	"
31	TwoWeekDrive	uint	2	"
33	MaximumDailyDrive	uint	1	Number of hours duration
34	MaximumWeeklyDrive	uint	1	"
35	MaximumTwoWeekDrive	uint	1	"
36	RemainingDrive	uint	2	Number of minutes duration
38	DurationOfNextBreakRest	uint	2	"
40	ExtendedDriveAvailable	uint	1	Number of extended drive periods still available
41	BreakRestTaken	uint	2	Number of minutes duration
43	RequiredBreakRest	uint	2	"
45	MaximumDurationOfNextDrivingPeriod	uint	2	"
47	ReducedRestAvailable	uint	1	Number of reduced rest periods still available
48	NextDailyRestDue	uint	4	TimeReal (Annex 1B)
52	NextWeeklyRestDue	uint	4	TimeReal (Annex 1B)
56	MinimumDurationOfNextDailyRest	uint	2	Number of minutes duration
58	MinimumDurationOfNextWeeklyRest	uint	2	"
60	WorkingTimeDaily	unit	2	"
62	WorkingTimeWeekly	unit	2	"
64	WorkingTimeBreakTaken	unit	2	"

Notes:
Values of 0xFF or 0xFFFF should be treated as NULL. This indicates that the values were unavailable in the original feed.

14.5. Time Related States

The parameter is used to indicate when the driver approaches/exceeds driving or working time limits. If two or more pre-warnings/warnings are simultaneously active, the pre-warning/warning corresponding to the lowest value shall be indicated. (Nevertheless, an actual warning should always have a higher priority than a pre-warning.)

Value	Description	Level
0000	No time related warning active	0
0001	Continuous driving time pre-warning active (15 min before 4h30)	1
0010	Continuous driving time warning active (4h30 exceeded)	2
0011	Daily driving time pre-warning active	1
0100	Daily driving time warning active	2
0101	Daily/weekly rest pre-warning active	1
0110	Daily/weekly rest warning active	2
0111	Weekly driving time pre-warning active	1
1000	Weekly driving time warning active	2
1001	Two week driving time pre-warning active	1
1010	Two week driving time warning active	2
1011	Card expiry warning active	1
1100	Next mandatory driver card download warning active	1
1101	other	0

The VDO tachographs only output values 0000, 0001 and 0010. This is also the case for Stoneridge tachographs up to and including 7.3. When Stoneridge introduced the 7.4 tachograph the full list was supported.

15. Vehicle CAN data

The digiDLex can collect SAE J1939 data by listening to the Vehicle CAN bus or by using an FMS standard interface. From these, calculations such as fuel consumption or idle time and assessments of driving style can be made.

15.1. Initialisation

This is triggered after a device has been reset and an FMS interface is begin used to get Vehicle CAN data.

Offset	Field	Datatype	Size	Source Message	Notes
0	Type	byte	1		0x46
1	Subtype	byte	1		0x80
2	DateTime	TimeReal	4		TimeReal (Annex 1B)
6	Requests & Diagnostics Supported	byte	1	FMS	uuuu rrdd u – unused r – requests supported d –diagnostics supported
7	Software Version	string	4	FMS	ab.cd

15.2. Engine On Message

This is triggered when the Engine Speed (EEC1) changes from 0 to >0.

Offset	Field	Datatype	Size	Source Message	Notes
0	Type	byte	1		0x46
1	Subtype	byte	1		0x81
2	DateTime	TimeReal	4		TimeReal (Annex 1B)
6	CardNumber	string	14		CardNumber.driverIdentification (Annex 1B)
20	Total Fuel Used	[1]uint	4	HRLFC / LFC	n * 0.001 = litres
24	Fuel Level	uint	1	DD	n * 0.4 = %
25	Odometer	[1]uint	4	VDHR	n * 0.005 = kilometres
29	Engine Temperature	uint	1	ET1	n - 40 = °C
30	Ambient Temperature	[1]uint	2	AMB	n * 0.03125 - 273 = °C
32	Service Distance	[1]uint	2	SERV	n * 5 – 160635 = kilometres

Notes:

[1]The Little-Endian convention is used to store these unsigned integer values.

Calculations may result in signed integer values.

15.3. Engine Off Message

Triggered when Engine Speed (EEC1) changes to 0.

Offset	Field	Datatype	Size	Source Message	Notes
0	Type	byte	1		0x46
1	Subtype	byte	1		0x83
2	DateTime	TimeReal	4		TimeReal (Annex 1B)
6	Total Fuel Used	[1]uint	4	HRLFC / LFC	n * 0.001 = litres
10	Fuel Level	uint	1	DD	n * 0.4 = %
11	Odometer	[1]uint	4	VDHR	n * 0.005 = kilometres
15	Engine Temperature	uint	1	ET1	n - 40 = °C
16	Ambient Temperature	[1]uint	2	AMB	n * 0.03125 - 273 = °C
18	Journey Duration	[1]uint	2		Number of seconds duration
20	Journey Fuel Used	[1]uint	3	HRLFC / LFC	n * 0.001 = litres
23	Journey Distance	[1]uint	3	VDHR	n * 0.005 = kilometres
26	Cruise Control Duration	[1]uint	2	CCVS	Number of seconds duration
28	Cruise Control Fuel Used	[1]uint	2	HRLFC / LFC	n * 0.001 = litres
30	Idle Duration	[1]uint	2	EEC1, CCVS	Idle if Engine Speed > 0 & Wheel Speed = 0
32	Fuel Used in Idle	[1]uint	2	HRLFC / LFC	n * 0.001 = litres
34	PTO Duration	[1]uint	2	CCVS	Number of seconds duration
36	Fuel Used in PTO	[1]uint	2	HRLFC / LFC	n * 0.001 = litres
38	Number of Brake Presses	[1]uint	2	CCVS	
40	Number of Harsh Braking Events	[1]uint	2	CCVS	[2] < -1.5m/s²
42	Duration of Harsh Braking	[1]uint	2	CCVS	Number of seconds duration
44	Number of Harsh Acceleration Events	[1]uint	2	CCVS	[2] > 1.5m/s²
46	Duration of Harsh Acceleration	[1]uint	2	CCVS	Number of seconds duration
48	Number of Clutch Presses	[1]uint	2	CCVS	
50	Number of Short Idle Events	uint	1	EEC1, CCVS	[2]Idle Duration > 2 minutes
51	Number of Long Idle Events	uint	1	EEC1, CCVS	[2]Idle Duration > 5 minutes
52	Duration of High Engine Power	[1]uint	2	EEC2	[2] > 90% Number of seconds duration
54	Number of High Power Events	uint	2	EEC2	
56	Duration of High Engine Speed	uint	2	EEC1	[2] > 90% Number of seconds duration
58	Number of High Speed Events	uint	2	EEC1	
60	Current Max Engine Speed	uint	2	EEC1	n * 0.125 = rpm

Notes:

[1]The Little-Endian convention is used to store these unsigned integer values.

[2]Thresholds can be set on a per device basis.

Calculations may result in signed integer values.

15.4. In Journey Message

Once a minute while Engine Speed (EEC1) > 0.

Offset	Field	Datatype	Size	Source Message	Notes
0	Type	byte	1		0x46
1	Subtype	byte	1		0x82
2	DateTime	TimeReal	4		TimeReal (Annex 1B)
6	Total Fuel Used	[1]uint	4	HRLFC / LFC	n * 0.001 = litres
10	Fuel Level	uint	1	DD	n * 0.4 = %
11	Odometer	[1]uint	4	VDHR	n * 0.005 = kilometres
15	Engine Temperature	uint	1	ET1	n - 40 = °C
16	Ambient Temperature	[1]uint	2	AMB	n * 0.03125 - 273 = °C
18	Duration Since Last Log	[1]uint	2		Number of seconds duration
20	Fuel Used Since Last Log	[1]uint	2	HRLFC	n * 0.001 = litres
22	Distance Travelled Since Last Log	[1]uint	2	VDHR	n * 0.005 = kilometres
24	Average Vehicle Speed	[1]uint	2	CCVS	n / 256 = km/h
26	Average Engine Speed	[1]uint	2	EEC1	n * 0.125 = rpm
28	Maximum Vehicle Speed	[1]uint	2	CCVS	n / 256 = km/h
30	Maximum Engine Speed	[1]uint	2	EEC1	n * 0.125 = rpm

Notes:

[1]The Little-Endian convention is used to store these unsigned integer values.

Calculations may result in signed integer values.

15.5. On Change Message

Triggered when one of the included items (not DateTime) changes.

Offset	Field	Datatype	Size	Source Message	Notes
0	Type	byte	1		0x46
1	Subtype	byte	1		0x84
2	DateTime	TimeReal	4		TimeReal (Annex 1B)
6	Tell Tale	binary	32	FMS1	
38	Door Control 1	binary	1	DC1	
39	Door Control 2	binary	8	DC2	

16. Sensors

The digiDLe/ex can support the monitoring of up to 4 switches and 2 analogue feeds. It can also send back information based on its built-in accelerometer. The switches will trigger a message on change. The analogue feeds will be polled on an interval set per device. The accelerometer will only trigger a message when crossing a threshold. These thresholds are set per device. On device start up, all the current values of these sensors will be sent to digiCentral.

Each sensor type uses a different structure for the data packet.

16.1. Switch

Offset	Field	Datatype	Size	Notes
0	Type	byte	1	0x53
1	Sensor Number	byte	1	0x0 - 0x3 – Switch 0 - 3.
2	DateTime	TimeReal	4	TimeReal (Annex 1B)
6	Value	uint	1	

16.2. Analogue

Offset	Field	Datatype	Size	Notes
0	Type	byte	1	0x53
1	Sensor Number	byte	1	0x4 - 0x5 – Analogue 0 - 1.
2	DateTime	TimeReal	4	TimeReal (Annex 1B)
6	Value	uint	2	

16.3. Accelerometer

Offset	Field	Datatype	Size	Notes
0	Type	byte	1	0x53
1	Sensor Number	byte	1	0x6 – Accelerometer.
2	DateTime	TimeReal	4	TimeReal (Annex 1B)
6	Value	uint	2	Magnitude only n * 0.001 g or n * 0.00980665 m/s^2
8	Duration	uint	2	n * 0.1 = number of seconds duration

16.4. 1-Wire Temperature Probes

Offset	Field	Datatype	Size	Notes
0	Type	byte	1	0x53
1	Sensor Number	byte	1	0x10 – 0x19.
2	DateTime	TimeReal	4	TimeReal (Annex 1B)
6	Value	int	2	Signed value n * 0.0625 centigrade (range -40 to +85)
8	ProbeId	byte	8	1-Wire Registration Number

16.5. Passenger Counters

Offset	Field	Datatype	Size	Notes
0	Type	byte	1	0x53
1	Sensor Number	byte	1	0x20 – 0x2F.
2	DateTime	TimeReal	4	TimeReal (Annex 1B)
6	Door Open/Shut	byte	1	0x0 – Open, 0x1 - Shut
7	RFU	byte	1	
8	Passenger entry count	byte	1	
9	Passenger exit count	byte	1	

17. Odometer

The digiDL can be configured to monitor the odometer value and return it to the server at the beginning and end of each journey. This is designed to work with both actual and simulator tachographs.

Offset	Field	Datatype	Size	Notes
0	Type	byte	1	0x4F
1	Journey Start/End	byte	1	0x0 – Journey End, 0x1 – Journey Start
2	DateTime	TimeReal	4	TimeReal (Annex 1B)
6	VIN	string	17	VehicleIdentificationNumber (Annex 1B)
23	Odometer	uint	4	n * 0.005 = kilometres
27	Source	byte	1	0x0 – Tachograph, 0x1 – Simulator