



# Inflow Process Specification

**Confidential** – By no means does Car-Pass authorize the reader to internal use the present information. The information contained in the present proposal may not be disclosed to any other people or organization without prior authorization of Car-Pass. This information cannot be sold or given for commercial purpose.

**version 8 – 21/12/2018**

Version	Date	Changes
1.0	23/04/2006	First public version
2.0	28/06/2006	- several clarifications after meeting with importers/software companies - added feedback type in header row of an input file to allow FTP feedback and e-mail feedback to a specified e-mail address - XML Schema's updated - reception feedback simplified
3.0	11/09/2006	several clarifications (production ftp url, sender id = user id, production http upload url)
4.0	14/04/2010	added section about directly opening history screen in the web application
5.0	17/06/2010	clarified how to select a language when directly opening history screen
6.0	14/05/2014	New chapter of Web services
6.1	03/05/2014	Added a Status for "Request Odometer Reading Status Response" method in Webservice
6.1.1	11/08/2014	Corrections in examples page

6.1.2	17/09/2014	Corrected Web Service examples in 5.2
6.1.3	5/1/2015	Added correct production WSDL links
6.1.4	10/02/2015	Host added
6.1.5	17/02/2015	Correction in examples xsi (xsi:noNamespaceSchemaLocation)
6.1.6	27/01/2016	FTP protocol Deprecated Webservice flow added
6.1.7	25/10/2016	Added section OK Label Correction in mandatory field flat file (issue id)
6.1.8	08/09/2017	Updated to reflected changes on issue feedback when the issue can't be corrected
6.2	20/09/2017	Delivery of data by FTP is set to END OF LIFE
7	05/07/2018	HTTPS, FTP removed (stop 1/03/2019)
8	21/12/2019	Added V2 of the Professional Web Services, adding extra functionality for Vehicle History Reports

## Table of content

<b>1</b>	<b>Context and Objectives .....</b>	<b>4</b>
<b>2</b>	<b>Car-Pass Dictionary .....</b>	<b>5</b>
<b>3</b>	<b>Business processes .....</b>	<b>6</b>
3.1	Send a new odometer reading .....	6
3.1.1	Feedback via Web services .....	8
3.2	Send an odometer reading correction .....	9
3.3	Request a vehicle history report .....	12
3.4	Request OK Label .....	13
<b>4</b>	<b>Communication channels .....</b>	<b>15</b>
4.1	FTPS .....	16
4.2	Web Services .....	16
4.2.1	Versioning .....	16
4.2.2	General requirements .....	16
4.2.2.1	Transfer of the data .....	16
4.2.2.2	Checks to ensure correct transfer and subsequent receipt by Car-Pass .....	17
4.2.2.3	The DMS system displays the error messages with the correct problem number, issue type and description 17	
4.2.2.4	The DMS system allows problems to be corrected .....	17
4.2.2.5	The Multithreading (multi socket) is not allowed .....	17
4.2.2.6	Built-in checks to avoid system-based errors .....	17
4.2.2.7	Checks to ensure the correct data type .....	18
4.2.2.8	Check to ensure that 17 characters are entered for VIN .....	18
4.2.2.9	The system makes it possible to exclude operations that are not mandatory to report (e.g. till sales, etc.) and certain categories of vehicles from being transferred to CAR-PASS (trucks, foreign number plates, etc.) .....	18
4.2.2.10	The system transfers the correct combination of date of works/odometer reading .....	18
4.2.2.11	The user must be able to check his user codes and password .....	18
4.2.2.12	The system generates a correction form in order to ensure that a double error (issue type 129) can be corrected by Car-Pass or alternatively, the garage can retrieve this document via the website = exclusion criterion .....	19
4.2.2.13	History request .....	19
4.2.3	WSDL .....	19
4.2.3.1	Location .....	19
4.2.3.2	Security .....	19
4.2.3.3	Dataflow .....	20
<b>5</b>	<b>Data structure .....</b>	<b>21</b>
5.1.1	Register Odometer Reading: PROC-INPUT-3 .....	21
5.1.1.1	Provide new Odometer Readings .....	22
5.1.1.2	Correct Issues .....	23
5.1.1.3	Request Vehicle History Report .....	24
5.1.1.4	Register Odometer Reading Response .....	24
5.1.2	Request Odometer Reading Status: PROC-FEEDBACK-1C .....	25
5.1.2.1	Request Odometer Reading Status Response .....	26
<b>6</b>	<b>Modified business rules .....</b>	<b>29</b>
6.1	Correction of certain Issues .....	29
6.2	Correction PDF .....	30
6.3	Changed Issues .....	32
6.3.1	New Issues .....	32
6.3.2	Issues that can't be corrected .....	32

---

---

# 1 Context and Objectives

Car-Pass is the organization created by **FEBIAC** (Belgian federation of the Car and Two-wheeler Industries), **GOCA** (Regroups recognized organizations in charge of the technical car control) and **Traxio** (Federation of car vendors, car service professionals as well as linked sectors), charged by the legislator with a task of public interest i.e. to protect buyers and to promote the fair trade in used vehicles by combating fraud with the odometer.

The reason of existence of the organization arises from the federal law of 11/06/2004. The purpose of this legislation is the prevention and repression of fraud linked to odometer manipulations. The essential principles are the following:

- Mileage fraud is considered a serious crime sanctioned with severe penalties (up to 1 year imprisonment)
- Creation of a central database containing the odometer readings of all vehicles (cars and vans) registered in Belgium
- All professional car dealers and repair shops are required to transfer VIN number, mileage and date when repairing or maintaining a vehicle or replacing parts (ex. tyres, windscreens,...)
- The seller of a second hand vehicle is obliged by law to deliver a certificate showing the mileage history of the vehicle to the buyer. If he fails to do so the transaction is void

The Car-Pass nonprofit was certified by royal decree of 4/5/2006 to manage the database and issue the mileage certificates.

To fulfill its mission, Car-Pass developed an information system covering the following **objectives**:

- Centralize odometer status information originating from multiple data providers located all over the Belgian territory.
- Store the kilometer information about all cars and vans registered in Belgium and gathered at different moments during their 'lifetime'
- Guarantee the quality, security and confidentiality of all centralized data.
- Deliver a certificate and information about the mileage history of a car as a guarantee and validation method for the buyer of a second hand car.
- Support prevention as well as repression efforts opposing odometer fraud.

In June 2018 the federal government agreed upon a modification of the 2004 law. One of the objectives is to improve the accuracy of the communicated odometer readings. The 2004 legislation allows companies 5 working days to communicate their data to Car-Pass. If companies make a mistake in registering the odometer reading it is, of course, very difficult to rectify as the vehicle has long since left the workshop. This is why the new wording stipulates that the data must be sent to Car-Pass immediately, i.e. while the vehicle is still on the company premises. Therefore, if a company uses file transfer for communicating its data to Car-Pass, it will break the law. That is why only the use of webservices and input via the Car-Pass website will be supported as from 2019.

## 2 Car-Pass Dictionary

Throughout the document different terms are used. Here are definitions of the user terminology.

To explain some terms an example enterprise is given:

Garage "Cool Cars" has 3 sites where Vehicles are sold and repaired and one headquarter (Cool Car HQ) where all administration is done.

- Cool Car 1  
→ Establishment number 2xxxxxxx1
- Cool Car 2  
→ Establishment number 2xxxxxxx2
- Cool Car 3  
→ Establishment number 2xxxxxxx3
- Cool Car HQ  
→ Enterprise number 04xxxxxxx1

Establishment number and enterprise number refers to the official data in KBO/BCE (Kruispuntbank van ondernemingen/ Banque Carrefour des Enterprises),

Odometer Readings are read in the three sites and the headquarters centralizes all data and sends it to Car-Pass.

Term	Description
VIN	Vehicle Identification Number. The unique number for the identification of a vehicle
Odometer Reading	The value of the Odometer
Enterprise n°	In the used example this must be 04xxxxxxx1
User code for the party that entered the data (usually the establishment n°)	Identifies the Organisation that performed work on the Vehicle and noted the Odometer Reading  In the used example this can be <ul style="list-style-type: none"> <li>- 2xxxxxxx1</li> <li>- 2xxxxxxx2</li> <li>- 2xxxxxxx3</li> </ul>
internal code (optional)	Can be: <ul style="list-style-type: none"> <li>- Name of the technician</li> <li>- Invoice number</li> <li>- Order number</li> <li>- ...</li> </ul>

---

---

## 3 Business processes

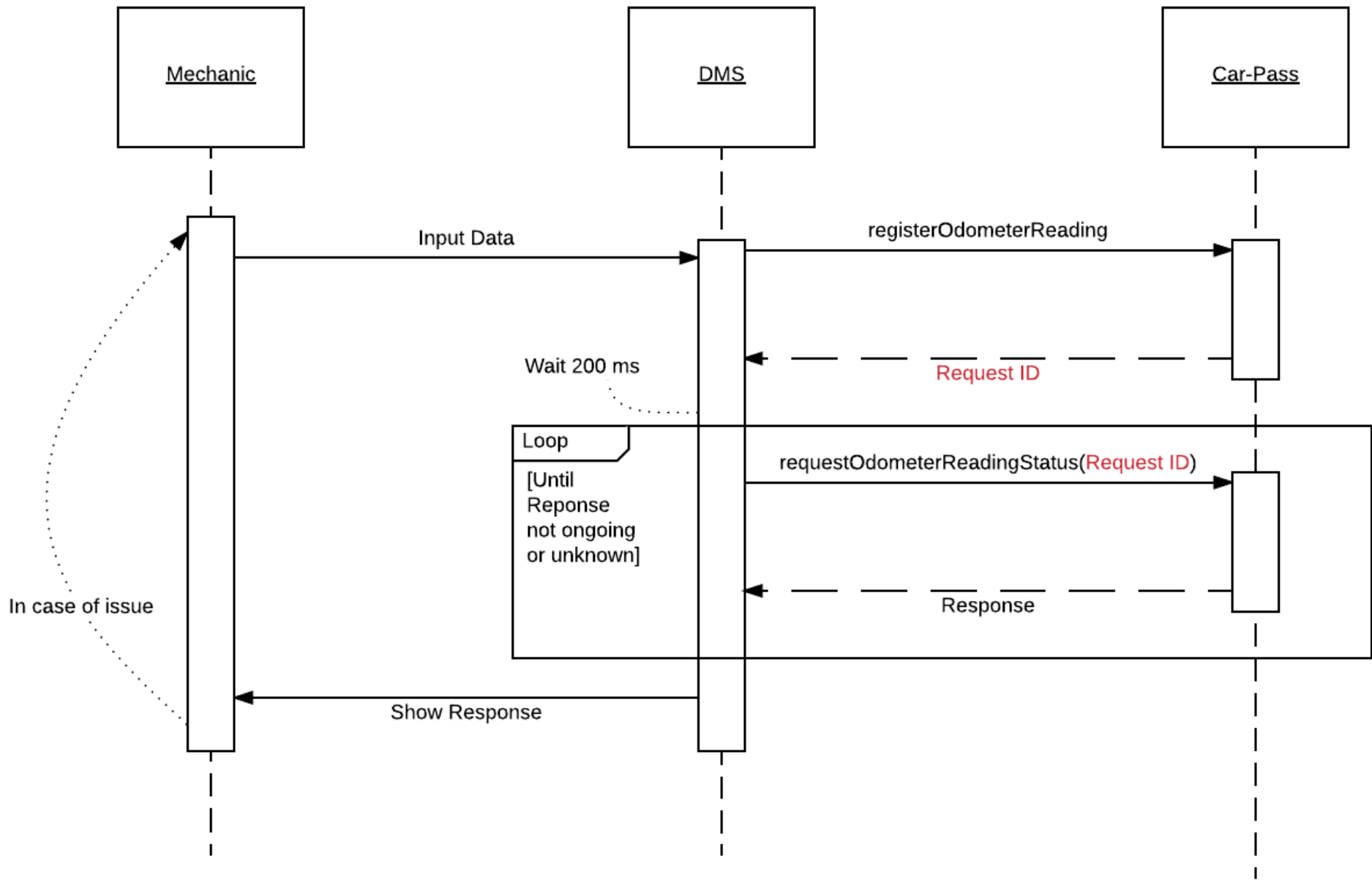
As mentioned, the first objective is the centralization of information. As to this centralization we can identify two main processes. The first covers the regular upload of new odometer status information while the second handles data correction requests. There are two secondary processes, the first to request a vehicle history request of a vehicle. The second to request an OK Label.

---

### 3.1 Send a new odometer reading

The main input into the system consists of odometer status information communication. This communication can contain data about one observation. Five steps can be identified in this procedure.

- **Take note of the odometer reading:** this operation is performed by a sector professional and requires the presence of a vehicle.
- **Send the odometer reading to Car-Pass:** the professional sends the kilometer information related to one vehicle or which he has the legal obligation. He identifies himself in order to inform Car-Pass about the origin of the data.
- **Validate and parse data:** reception, validation and parsing by Car-Pass of kilometer information according to the known type and format (see below for exact definitions)
- **Store information in DB:** transformation of the raw data to the Car-Pass format as well as verification of the information based on a set of quality standards, technical information (odometer) and transformation rules. The concerned cars historical information combined with alert rules allow the definition of additional criteria making it possible to detect fraud and reject the information.
- **Provide feedback:** Information about the provided method or the result of the validation of the provided data is returned to the professional.



### 3.1.1 Feedback via Web services

- Web service feedback ([PROC-FEEDBACK-1C](#)).  
Once the provided request (1 request contains 1 record, related to 1 vehicle) is handled by the backend system nothing is prepared. The professional will ask a status request and then the feedback is retrieved and returned. One feedback message will be returned per transfer.



## 3.2 Send an odometer reading correction

As a consequence of technical or human factors, errors can occur resulting in a transfer of corrupt or incorrect data. For most error we expect a Correction to be performed, in a rare circumstance an odometer reading correction can't be corrected. In that case we'll return feedback about the issue type but **without** issue number. Based on the nature and the impact of the error, three categories can be distinguished.

- **Technical or logical errors** due to corrupt data, wrong file structure, badly formatted data or impossible values (non existing codes, date in the future, unexisting VIN,...)
- **Possible errors** identified using based on probabilities and assumptions and not linked to potential fraud. An example of such an error could be an odometer status raise of over 15 000 km in less than one month. It's very unlikely that such an event occurs. These are called warnings.
- **Anomaly in the kilometer progression** for a car. As this could be an indicator for fraud, *this is the most important kind of error*. All the issues can be found here: <https://www.car-pass.be/files/files/file/1/issuetypes%2020171019.xls>

In all of these cases, the sender will be informed of the problematic event ([PROC-FEEDBACK-1C](#)). Three types of issues will be generated corresponding to the three issue types described above:

- **Error** feedback reports technical or logical errors
- **Warnings** refer to possible errors
- **Anomaly** inform about anomalies in kilometer progression

As corrections in case of errors or warnings do not cause a risk for fraudulent exploitation, an automated correction procedure will be put in place making it possible to replace erroneous data. In case of an **error**, the original data will never be loaded to the system. In case of a **warning**, there is a chance that the original data were correct. Therefore, these data are initially loaded to the system. In case the warned data provider finds out that they are not correct, a correction procedure can be invoked and the original data will be replaced on the system.

Whereas corrections can be executed in reaction on errors and warnings, **alerts** can only be corrected once. If this correction causes a new error the system will block and an administrative procedure will have to be invoked to eventually change the initially loaded information. Kilometer information invoking an alert will be stored on the system in the same way as correct information.

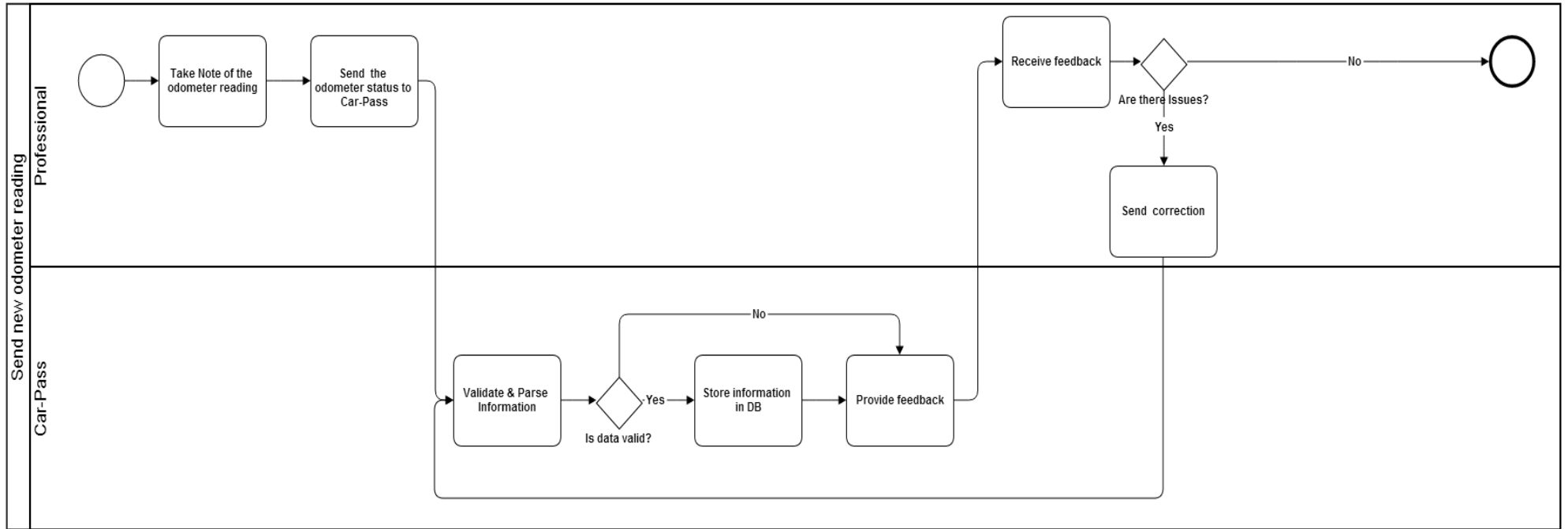
The table below presents a summarized overview of the different issue categories:

issue	issue type	data stored	correction possible
technical or logical errors	error	<input type="checkbox"/>	<input checked="" type="checkbox"/>
probable errors	warning	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> (*)
anomaly in kilometer progression	anomaly	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> (*)

(\*) Only one correction possible, for more details, see [7.1](#)

When a correction is handled successfully, all Issues related to the same odometer reading will be closed.

The correction procedure for errors, warnings and anomalies can be graphically represented as follows :



As a result of an odometer status correction, the standard feedback procedures will be invoked. In case the correction solved the problem, the load will be reported in the load feedback. In case the correction recreated the problem or invoked a new one, an issue feedback will be triggered.

---

### 3.3 Request a vehicle history report

Through the web service a vehicle history report can be requested. This can only be requested by professionals who are allowed to retrieve this data.

To request a Vehicle History a new Odometer Reading must be provided. Therefor the same web service method is used as to provide a new Odometer Reading. The distinction is made based on the provided data type (010).

When the professional requests the status of the provided Odometer Reading, feedback is returned with the Vehicle History report attached.

If the provided data caused an issue (see 3,2), an issue feedback is returned . *This issue has to be solved before the user can retrieve the Vehicle History.*

Issues can be corrected on the same way as a normal correction. When the correction valid and the data is accepted by Car-Pass the Vehicle History Report will be generated. The professional can retrieve the Vehicle History report by performing a status request.

---

### 3.4 Request OK Label

1. Before you can make an OK Label request you'll need an authenticated User ID and Password from Car-Pass, you need an account for the Staging environment and another for the Production environment
2. You can find the WSDL on the following links
  1. Staging: <https://professionals.beta.car-pass.be/cpoklabel/services/carpassoklabel?wsdl>
  2. Production: <https://professionals.car-pass.be/cpoklabel/services/carpassoklabel?wsdl>
3. The following is an example of a request

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:cpl="http://www.car-pass.be/CpLabel">
  <soapenv:Header>
    <wsse:Security
      soapenv:mustUnderstand="1"
      xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
      <wsse:UsernameToken
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
        <wsse:Username>[Personal_User_ID]</wsse:Username>
        <wsse:Password>[Personal_Password]</wsse:Password>
      </wsse:UsernameToken>
    </wsse:Security>
  </soapenv:Header>
  <soapenv:Body>
    <cpl:CpLabelRequest>
      <ChassisNumber>[VIN_Number]</ChassisNumber>
      <KilometerStatus>[Provided_Odometer_Value]</KilometerStatus>
      <InternalRef>[Optional]</InternalRef>
    </cpl:CpLabelRequest>
  </soapenv:Body>
</soapenv:Envelope>
```

4. There are 4 mandatory fields and one optional:
  1. Username: provide the User ID of your company
  2. Password: fill in your password
  3. ChassisNumber: provide the VIN for the Vehicle you're requesting the OK Label for
  4. KilometerStatus: provide the Odometer Value for the Vehicle
  5. InternalRef: Optional field, can be used to uniquely identify your Users/Customers

5. The following is an example of a possible response

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns2:CpLabelResponse xmlns:ns2="http://www.car-pass.be/CpLabel">
      <CarpassLabel>OK</CarpassLabel>
    </ns2:CpLabelResponse>
  </soap:Body>
</soap:Envelope>
```

```
</ns2:CpLabelResponse>  
</soap:Body>
```

```
</soap:Envelope>
```

6. The possible statuses are
  1. OK
  2. NOK
  3. INSUFFICIENT\_DATA
  4. DIGITS\_5
  5. UNKNOWN
  6. VIN\_NOT\_UNIQUE
  7. IMPORT
  8. ERROR

---

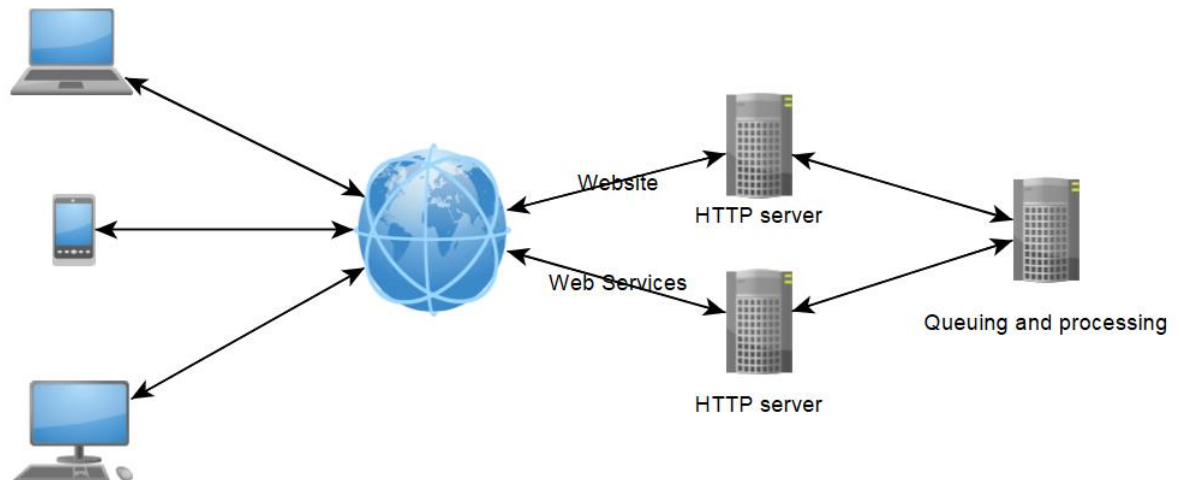
---

## 4 Communication channels

The first objective of the Car-Pass system is the centralization of information originating from multiple sources. The range of potential data providers is wide, going from large companies and organizations as the car importers, tyre shop networks or technical inspection to the local independent car service professional. To cover all of these potential data providers, 2 channels will be proposed by Car-Pass. The following alternatives are presented.

- Use the Car-Pass **website**  
A client can provide odometer readings, correct data and request odometer reports. No implementation is needed, only an updated computer.
- Integrate the **webservices** in a **third party application**.  
To make it possible for existing applications to integrate with the Car-Pass system. The user interfaces send and correct their data directly into the Car-Pass system.

An overview of the client and communication layer of the Car-Pass system is presented in the figure below.



Only WEB SERVICES will be supported to interface with the Car-Pass system.

---

## 4.1 FTPS

### FTPS will be stopped at 01/03/2019

The change of the law foreseen for the 1<sup>st</sup> of March 2019, will stipulate that a odometer reading has to be send to Car-Pass while the vehicle is still in the workplace. And the correction of faults has to be done immediately.

Because FPTs isn't a real time communication channel we will stop our FTP server. Please be ready with your changes before March 2019.

---

## 4.2 Web Services

This solution offers the possibility to provide Odometer Readings through Web services. **Only authenticated users** will be allowed to connect to the Web Service. Before users will be able to connect to the Web Service, they will have to activate their account at the Car-Pass website.

There are just 2 operations:

- registerOdometerReading (can be used for normal registrations, history requests and to make corrections)
- requestOdometerReadingStatus

### 4.2.1 Versioning

- Version 2 of the web services has been activated on the beta environment on 20/12/2018 and will become active on production on 1/3/2019
- Version 2 provides extra functionalities for Vehicle History Report but has mostly the same structure as Version 1
- All new development should happen on version 2, and users already working with version 1 are advised to switch to version 2 as version 1 will no longer be updated.
- Version 1 wsdl: <https://ws-professionals.beta.car-pass.be/carpas-web-services-professional/WSCarPassPro?wsdl=CarPassWSPro.wsdl>
- Version 2 wsdl: <https://ws-professionals.beta.car-pass.be/carpas-web-services-professional/WSCarPassPro2?wsdl=CarPassWSPro2.wsdl>

### 4.2.2 General requirements

#### 4.2.2.1 Transfer of the data

The data will be transferred in one of the following ways

- a) Automatic transfer (automatically after entering data on, for example, a work order and without the user having to explicitly press a button).
- b) Manually, with installation of the necessary guarantees (warnings, reminders, etc.) that the data is being *sent before the vehicle file is closed or before the vehicle leaves the repairshop.*



#### **4.2.2.2 Checks to ensure correct transfer and subsequent receipt by Car-Pass**

The DMS system must contain the necessary checks to verify whether the data has actually been sent to Car-Pass and whether Car-Pass has received it properly.

- 1) In the first phase of the web service, Car-Pass sends back a requestID.  
Is this requestID captured and stored, so that this requestID can be used even after restarting the PC/server?
- 2) Before launching the requestOdometerReadingStatus, is there a delay time of 0.5 seconds as minimum?
- 3) Is a record kept of the status having returned an OK, so that the status is only retrieved once after a positive message?
- 4) Has the user been warned if a message is returned = ERROR?
- 5) Has the requestOdometerReadingStatusResponse been captured so that any issues can be improved?
- 6) Is the system capable of sending corrections?
- 7) Is an attempt made to resend the messages after an ERROR OR UNKNOWN message?

#### **4.2.2.3 The DMS system displays the error messages with the correct problem number, issue type and description**

In the DMS system, the outstanding problems must be visible, showing the correct problem number, possibly the error code (issue type) but certainly the description of the problem.

Like:

- Insertion of a double observation Issue number 150
- Insertion of an unknown fin Issue No. 131
- Insertion of an 'unexpected high' Issue Number 209
- Insertion of a 'lower than previous' Issue number 001

#### **4.2.2.4 The DMS system allows problems to be corrected**

The system allows corrections to be made. It should be possible to confirm problems as correct or to remove them from the list of outstanding issues without correction.

The DMS system will show those issues that cannot be corrected, but ensures that they cannot be corrected.

These corrections will be implemented in the local database as well as in the database of Car-Pass.

The feedback on these improvements will be correctly loaded.

The issue will disappear from the list of outstanding problems

#### **4.2.2.5 The Multithreading (multi socket) is not allowed**

To be derived from the software documentation to be delivered

The maximum number of threads is limited to 1 in the communication to Car-Pass; this is to prevent everything being blocked by 1 user.

#### **4.2.2.6 Built-in checks to avoid system-based errors**

The DMS system carries out a check to ensure if the entered kilometre reading is possible (precheck of data in the own system)

- a. The DMS system gives a warning if the kilometre reading that is entered is lower than the previous odometer reading in the garage database (local issue 001)

- b. The DMS system gives a warning if the kilometre reading that is entered is excessively high compared to the previous odometer reading in the garage database .(local issue 209)
- c. The DMS system gives a warning if the kilometre reading that is entered is the same as a previous odometer reading in the garage database
- d. The DMS system gives a warning system if the date for the works is in the future
- e. The DMS system gives a warning if the date for the garage works is in the past.

#### **4.2.2.7 Checks to ensure the correct data type**

- a. Transfer of the odometer reading: code 120 (or 121)
- b. Transfer of the correction: code 120 (or 121) + issue\_id
- c. Closing the problem without correction: code 080
- d. If the garage repairs or replaces the odometer, this must be reported to Car-Pass via code 121

#### **4.2.2.8 Check to ensure that 17 characters are entered for VIN**

If a chassis number is entered with less than 17 characters, the user receives a warning. It should still be possible to send the data to Car-Pass.

When a chassis number with more than 17 characters is entered, the user receives an error message. It should not be possible to send the data to Car-Pass.

An empty or impossible chassis number (00000000000000000) is not allowed. Only alphanumeric values are allowed.

#### **4.2.2.9 The system makes it possible to exclude operations that are not mandatory to report (e.g. till sales, etc.) and certain categories of vehicles from being transferred to CAR-PASS (trucks, foreign number plates, etc.)**

The user must have the option of excluding these operations/vehicles from being transferred to Car-Pass.

#### **4.2.2.10 The system transfers the correct combination of date of works/odometer reading**

The system links the correct date to the correct kilometer reading, i.e. the date when the vehicle was present in the repair shop and the odometer reading was read off.

Possible errors that give rise to failure on this criterion (not exhaustive list):

- Using the invoice date
- Using the date of the odometer reading from the expertise linked to the date of works
- Date of the appointment
- Re-sending the odometer reading when creating a credit note / re-invoicing
- Claims under guarantee: cut-off date
- Etc.

#### **4.2.2.11 The user must be able to check his user codes and password**

Test: The DMS system must have a screen showing the contact details that were entered at the time of activating the Car-Pass user account. These are at the minimum the following:

- The user code
- The user's VAT number (enterprise code)
- The password

- The email address
- Optionally: telephone number of the Car-Pass helpdesk and link to the contact form.

If the enterprise changes its password and/or user code locally in the DMS system, at least one warning must appear warning that it must also be changed on the Car-Pass website.

#### **4.2.2.12 The system generates a correction form in order to ensure that a double error (issue type 129) can be corrected by Car-Pass or alternatively, the garage can retrieve this document via the website = exclusion criterion**

The user will be able to generate a correction request form in which at least the following details have been pre-filled:

- name of the enterprise
- user code
- problem number
- issue type
- data that was originally sent: km, chassis number and date
- improved data: km, chassis number and date
- a check whether the data has also been improved locally.

This document should be sent by email to [info@car-pass.be](mailto:info@car-pass.be)

#### **4.2.2.13 History request**

The system allows you to consult the history via web services.

Test: retrieval of the history for a random chassis number.

The DMS system opens the submitted pdf document and allows you to print the history.

### **4.2.3 WSDL**

#### **4.2.3.1 Location**

To provide Odometer Readings ([PROC-INPUT-3](#)) or to request Vehicle History Reports at Car-Pass, a (SOAP) web service has been created.

The WSDL for the web services can be found at: <https://ws-professionals.beta.car-pass.be/carpass-web-services-professional/WSCarPassPro2?wsdl>

#### **4.2.3.2 Security**

The Authorization header is constructed as follows:

1. Username and password are combined into a string "username:password"
2. The resulting string is then encoded using the RFC2045-MIME variant of Base64, except not limited to 76 char/line
3. The authorization method and a space i.e. "Basic " is then put before the encoded string.

The web service can only be accessed with a username and password. These credentials must always be provided while making a request to Car-Pass.

For each request the following data must be provided:

- POST: the url of the web service WSDL
  - <https://ws-professionals.beta.car-pass.be/carpass-web-services-professional/WSCarPassPro2?wsdl> (\*)
- Accept-Encoding:
  - gzip,deflate
- Content-Type:
  - Text/xml;charset=UTF-8
- SOAPAction:
  - Leave empty
- Authorization:
  - Basic authentication is used. The authentication follows the following pattern:
    - **"Basic "**
    - **Username:password**, base64 encrypted ( the RFC2045-MIME variant is used)
- Content-Length:
  - 848
- Host: the host of the web service
  - [ws-professionals.beta.car-pass.be](https://ws-professionals.beta.car-pass.be)
- Connection:
  - keep-alive
- User-Agent:
  - Apache-HttpClient/ 4.1.1

(\*) the production url is: <https://ws-professionals.beta.car-pass.be/carpass-web-services-professional/WSCarPassPro2?wsdl>

See the complete necessary header in the example below:

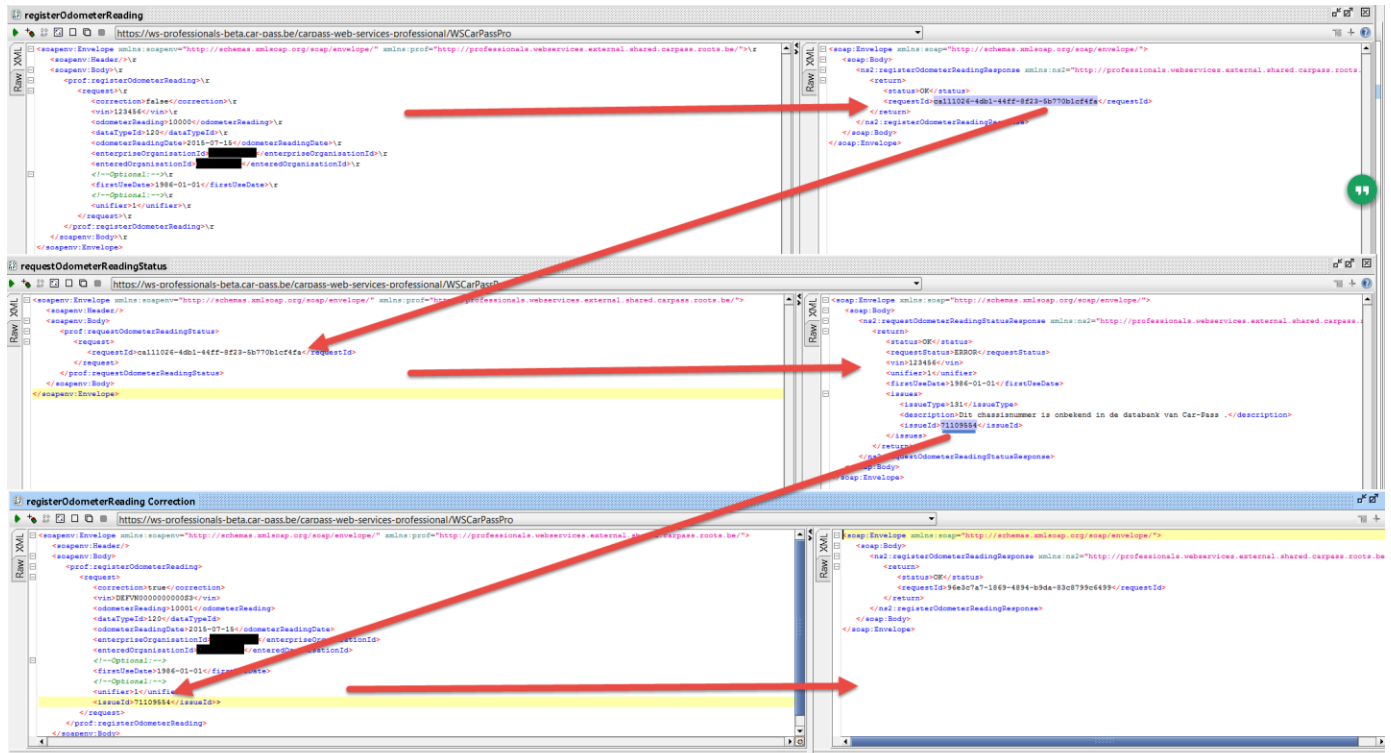
```
POST https://ws-professionals.beta.car-pass.be/carpass-web-services-professional/WSCarPassPro2?wsdl
Accept-Encoding: gzip,deflate
Content-Type: text/xml;charset=UTF-8
SOAPAction: ""
Authorization: Basic MjAwKKAwNjAwMTpQcjBlZSRza!:uYWw=
Content-Length: 848
Host: ws-professionals.beta.car-pass.be
Connection: Keep-Alive
User-Agent: Apache-HttpClient/4.1.1

<soapenv:Envelope ...>
...
</soapenv:Envelope>
```

### 4.2.3.3 Dataflow

The WEBSERVICES are setup *asynchronously*.

In the picture below you see the different steps that can be taken.



We recommend to implement a minimum delay time of 1 second between the registerOdometerReading and the requestOdometerReadingStatus.

## 5 Data structure

### 5.1 Web Services

This paragraph describes the data exchanged with the Car-Pass system through Web services for each of the input and feedback procedures. Both content and structure of the files or messages will be discussed.

Please, use only the WSDL for generating your code. The examples in the Inflow Process Specification are only indicative.

#### 5.1.1 Register Odometer Reading: PROC-INPUT-3

The following elements are contained in the odometer status:

- Correction
- VIN of the Observed Vehicle
- Odometer Reading status
- Type of the data
- Date of the observation
- Code (enterprise n°) for the party legally responsible for the information
- Code (user code) for the party that entered the data
- Date of first Use (optional)
- Unifier (optional)
- Issue id (optional)

This request can be used for 3 purposes:

- Provide new Odometer Reading

- Correct Issue
- Request Vehicle History Report

When the request is successful a response is returned with a unique key that will be the reference for all further communication with Car-Pass.

### 5.1.1.1 Provide new Odometer Readings

To provide a new Odometer Reading, make sure that you use the following rules

- Correction:
  - set to “false”
- VIN:
  - maximum 17 characters
- OdometerReading:
  - provided to correct Odometer reading value
- dataTypeId:
  - use 120 for a normal Odometer Reading (equals service reparation)
  - use 121 for a replaced or repaired odometer
- OdometerReadingDate:
  - format “yyyy-MM-dd”
- enterpriseOrganisationID:
  - the legally responsible
- enteredOrganisationId:
  - the User code of the party that entered the data.
- firstUseDate:
  - this is an optional field
  - format “yyyy-MM-dd”
- unifier:
  - When the VIN contains less than 17 characters, the Unifier is useful for unique identification of the vehicle.

See a full example below:

```
POST https://ws-professionals.car-pass.be/carpass-web-services-professional/WSCarPassPro2?wsdl
Accept-Encoding: gzip,deflate
Content-Type: text/xml;charset=UTF-8
SOAPAction: ""
Authorization: Basic:MjAwKKAwNjAwMTpQcjBlZSRza!:uYWw=
Content-Length: 848
Host: ws-professionals.car-pass.be
Connection: Keep-Alive
User-Agent: Apache-HttpClient/4.1.1

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:prof="https://professionals.webservices.carpass. ... .be/">
  <soapenv:Header/>
  <soapenv:Body>
    <prof:registerOdometerReading>
      <request>
        <correction>false</correction>
        <vin>VNWS000000005250</vin>
        <odometerReading>1234</odometerReading>
        <dataTypeId>120</dataTypeId>
        <odometerReadingDate>2014-04-25</odometerReadingDate>
        <enterpriseOrganisationId>0400006000</enterpriseOrganisationId>
        <enteredOrganisationId>2000006001</enteredOrganisationId>
        <firstUseDate>2000-01-01</firstUseDate>
        <unifier>1</unifier>
      </request>
    </prof:registerOdometerReading>
  </soapenv:Body>
</soapenv:Envelope>
```

```
</soapenv:Body>
</soapenv:Envelope>
```

### 5.1.1.2 Correct Issues

For the correction of issues the same web service method, registerOdometerReading, is used. To provide a new Correction, make sure that you use the following rules

- Correction:
  - set to “true”
- dataTypeId:
  - Use “120” for correction of the Issue. Make sure you provided the update value of the field that needs to be corrected
  - Use “121” for correction of the Issue in case of a replaced or repaired odometer
  - Use “080” to confirm the first provided Odometer Reading that caused the Issue as correct or to close the Issue without further correction.
- IssueId:
  - Provide Issue ID that you want to correct.

See a full example below:

```
POST https://ws-professionals.car-pass.be/carpass-web-services-
professional/WSCarPassPro2?wsdl
Accept-Encoding: gzip,deflate
Content-Type: text/xml;charset=UTF-8
SOAPAction: ""
Authorization: Basic:MjAwKKAwNjAwMTpQcjBlZSRza!:uYWw=
Content-Length: 848
Host: ws-professionals.car-pass.be
Connection: Keep-Alive
User-Agent: Apache-HttpClient/4.1.1

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:prof="https://professionals.webservices.carpass. ... .be/">
  <soapenv:Header/>
  <soapenv:Body>
    <prof:registerOdometerReading>
      <request>
        <correction>true</correction>
        <vin>VNWS000000005250</vin>
        <odometerReading>5214</odometerReading>
        <dataTypeId>080</dataTypeId>
        <odometerReadingDate>2014-04-25</odometerReadingDate>
        <enterpriseOrganisationId>0400006000</enterpriseOrganisationId>
        <enteredOrganisationId>2000006001</enteredOrganisationId>
        <firstUseDate>2000-01-01</firstUseDate>
        <unifier>1</unifier>
        <issueId>8850</issueId>
      </request>
    </prof:registerOdometerReading>
  </soapenv:Body>
</soapenv:Envelope>
```

### 5.1.1.3 Request Vehicle History Report

A Vehicle History Report can be requested through the web service. To retrieve it, a new Odometer Reading must be provided.

The same web service method as for a normal Odometer Reading is used; registerOdometerReading.

Use the rules from 5.2.1.1 except for:

- Correction:
  - set to “false”
- dataTypeId:
  - use “010” for a vehicle history report
- enteredOrganisationId:
  - the Organisation number of the enterprise or establishment that requested the Vehicle History Report and who is allowed to request it.

See a full example below:

```
POST http://ws-professionals.car-pass.be/carpass-web-services-professional/WSCarPassPro2?wsdl
Accept-Encoding: gzip,deflate
Content-Type: text/xml;charset=UTF-8
SOAPAction: ""
Authorization: Basic:MjAwKKAwNjAwMTpQcjBlZSRza!:uYWw=
Content-Length: 848
Host: ws-professionals.car-pass.be
Connection: Keep-Alive
User-Agent: Apache-HttpClient/4.1.1 (java 1.5)

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:prof="
https://professionals.webservices.carpass. ... .be /">
  <soapenv:Header/>
  <soapenv:Body>
    <prof:registerOdometerReading>
      <request>
        <correction>false</correction>
        <vin>VNWS000000005250</vin>
        <odometerReading>1265</odometerReading>
        <dataTypeId>010</dataTypeId>
        <odometerReadingDate>2014-04-28</odometerReadingDate>
        <enterpriseOrganisationId>040006000</enterpriseOrganisationId>
        <enteredOrganisationId>200006001</enteredOrganisationId>
        <firstUseDate>2000-01-01</firstUseDate>
        <unifier>1</unifier>
      </request>
    </prof:registerOdometerReading>
  </soapenv:Body>
</soapenv:Envelope>
```

### 5.1.1.4 Register Odometer Reading Response

Each web service Request will return a response. This will inform the requestor on:

- The Status of the Request
- A unique ID to track the Request and to retrieve the status

The following elements are contained in the odometer status response:

- Status: Possible values are



name	description
OK	The request was successful, all parameters are in the correct format. Car-Pass is processing the request.
VALIDATION_ERROR	There was an error during the validation of the parameters. e.g. the date was in a invalid format
CONNECTION_ERROR	There was a problem while connecting to Car-Pass. This request is not invalid, but should be resend again.
AUTHENTICATION_ERROR	The credentials used to connect with the web service are invalid.

- RequestId: A GUID of 36 characters, format xxxxxxxx-xxxx-xxxx-xxxxxxxxxxxx

**This GUID needs to be saved, because this is the key to the further processing of the request**

See a full example below:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns2:registerOdometerReadingResponse xmlns:ns2="https://ws-professionals.car-pass.be/carpass-web-services-professional/WSCarPassPro2?wsdl" >
      <return>
        <status>OK</status>
        <requestId> 52d71495-9c2b-46d7-862f-12543299ff3f</requestId>
      </return>
    </ns2:registerOdometerReadingResponse>
  </soap:Body>
</soap:Envelope>
```

### 5.1.2 Request Odometer Reading Status: PROC-FEEDBACK-1C

For each new Odometer reading that is provided or a Vehicle History Report that is requested through the web service a status can be retrieved.

This can be done using the web service method "requestOdometerReadingStatus"

The following elements are contained in the Request Odometer reading status:

- requestId
- includeHistoryReportPdf- Boolean – values: true or false
  - Optional field, if not provided it will be handled as false
  - Set this as true if the request was a Vehicle History Report and you want to include the PDF in the answer, see below

The requestId is the returned id, described in paragraph 5.1.1.4.

See a full example below:

```
POST https://ws-professionals.car-pass.be/carpass-web-services-professional/WSCarPassPro2?wsdl
Accept-Encoding: gzip,deflate
Content-Type: text/xml;charset=UTF-8
```

```

SOAPAction: ""
Authorization: Basic:MjAwKkAwNjAwMTpQcjBlZSRza!:uYWw=
Content-Length: 848
Host: ws-professionals.car-pass.be
Connection: Keep-Alive
User-Agent: Apache-HttpClient/4.1.1 (java 1.5)

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:prof="http://professionals.webservices.carpass.be/">
  <soapenv:Header/>
  <soapenv:Body>
    <prof:requestOdometerReadingStatus>
      <request>
        <requestId> 52d71495-9c2b-46d7-862f-12543299ff3f</requestId>
        <includeHistoryReportPdf>true</includeHistoryReportPdf>
      </request>
    </prof:requestOdometerReadingStatus>
  </soapenv:Body>
</soapenv:Envelope>

```

### 5.1.2.1 Request Odometer Reading Status Response

Each status request will return a response.

The following elements are contained in the odometer status response:

- Status: Possible values are listed in 5.1.1.4
- Requeststatus:

name	description
READY	The Odometer Reading is valid and accepted at Car-Pass. It is possible one or more issues are returned that need to be confirmed or corrected. When a Vehicle History Report is requested, it is returned in the response
ONGOING	The request id provided is known at Car-Pass, but it the processing is not completed. Please try again later
UNKNOWN	The provided request id is not known. Please try again later.
ERROR	There is an blocking Issue with the provided Odometer Reading, the registration is not accepted by Car-Pass. One or more issues are returned. There is no Vehicle History Report available
DELETED	The Vehicle History Report is no longer available at Car-Pass. Please provide a new request to retrieve a new Vehicle History Report.
NOT_ALLOWED	The Sender is not allowed to request the status for the provided RequestId

- Vehicle History Report: Object is only included when the request was for a Vehicle History Report (Datatype 010)
  - publicUrl: a url that shows the Vehicle History Report data, can be used in advertisements about a second hand car that is being sold



## Example 2: Odometer Reading status without issues

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns2: requestOdometerReadingStatusResponse
xmlns:ns2="http://professionals.webservices.carpass. ... .be/ ">
      <return>
        <status>OK</status>
        <requestStatus>READY</requestStatus>
        <vin>VNWS0000000005259</vin>
        <unifier>1</unifier>
        <firstUseDate>2000-01-01</firstUseDate>
      </return>
    </ns2: requestOdometerReadingStatusResponse >
  </soap:Body>
</soap:Envelope>
```

## Example 3: With issues

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns2: requestOdometerReadingStatusResponse
xmlns:ns2="https://professionals.webservices.carpass. ... .be ">
      <return>
        <status>OK</status>
        <requestStatus>READY</requestStatus>
        <vin>VNWS0000000005259</vin>
        <unifier>1</unifier>
        <firstUseDate>2000-01-01</firstUseDate>
        <issues>
          <issueType>208</issueType>
          <description>Le kilométrage est un nombre arrondi, évitez d'arrondir les
kilomètres.</description>
          <issueId>8850</issueId>
        </issues>
      </return>
    </ns2: requestOdometerReadingStatusResponse >
  </soap:Body>
</soap:Envelope>
```

## Example 4: With issue that can't be correct, without Issue ID

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns2: requestOdometerReadingStatusResponse
xmlns:ns2="https://professionals.webservices.carpass. ... .be ">
      <return>
        <status>OK</status>
        <requestStatus>READY</requestStatus>
        <vin>VNWS0000000005259</vin>
        <unifier>1</unifier>
        <firstUseDate>2000-01-01</firstUseDate>
        <issues>
          <issueType>150</issueType>
          <description>Double Observation</description>
        </issues>
      </return>
    </ns2: requestOdometerReadingStatusResponse >
  </soap:Body>
</soap:Envelope>
```

---

---

## 6 Modified business rules

This paragraph describes how the correction on an anomaly or Issues that can only be solved once should be handled.

---

### 6.1 Correction of certain Issues

When a new Odometer Reading is loaded in the database, the following Issues can occur:

Issue code	description
001	The provided Odometer Reading is lower than the previous one
203	Date of the provided Odometer Reading is too old
205	The Odometer Reading has more digits than defined vehicle type
208	The provided Odometer Reading is a round number
209	The provided Odometer Reading is unexpected high
210	Mini anomaly

Issues can only be corrected **once** by the User via Website or Web services.

When this correction causes a new Issue, it can only be corrected by sending a correction PDF to Car-Pass (see [6.2](#)). This will be indicated by returning the Issue 129 together with the second Issue (**Issue can only be solved manually**).

## 6.2 Correction PDF

When an Issue must be corrected manually a correction PDF must be send to Car-Pass.  
 See an example and the description of the correction PDF below. This PDF can be downloaded on the public website for professionals when doing a correction.  
 The following issues and issue types don't need to be filled in because they can't be corrected: Type 125, 126, 127, 128, 130, 150, 250



### Correctieverzoek

Per fax te sturen naar <sup>(1)</sup>: **02/771 58 64**

Gebruikerscode<sup>(2)</sup>:   
 Bedrijf:   
 Mijn faxnummer:

Issue ID	Type				
Chassisnummer				Km	
Datum opname		Datum 1ste inschrijving		Werken aan km-teller	
Chassisnummer	<input style="width: 20px; height: 15px;" type="text"/>	<input style="width: 20px; height: 15px;" type="text"/>	<input style="width: 20px; height: 15px;" type="text"/>	Km	<input style="width: 20px; height: 15px;" type="text"/>
Datum opname <sup>(3)</sup>	.. / .. / .....	Datum 1ste inschrijving <sup>(4)</sup>	.. / .. / .....	Werken aan km-teller <sup>(5)</sup>	<input type="checkbox"/>

Datum van verzending: .. / .. / .....  
 Naam<sup>(6)</sup>: .....  
 Functie: .....  
 Handtekening:

Ondergetekende deelt de bovenstaande gegevens mee aan de v.z.w. CAR-PASS in het kader van zijn verplichtingen voortvloeiend uit artikel 6 §3 van de wet van 11 juni 2004. Ondergetekende verklaart dat de meegedeelde kilometerstanden op de aangegeven datum overeenstemmen met de afgebeelde waarde op de kilometerteller van het door het chassisnummer gekenmerkte voertuig. Hij is zich ervan bewust dat het doorgeven van foutieve of onvolledige gegevens aan de v.z.w. CAR-PASS vervolgd en bestraft zal worden.

(1) ook per post mogelijk naar CAR-PASS vzw, Woluwedal 46/2, 1200 Brussel  
 (2) uw vestigingsnummer of uitzonderlijk uw ondernemingsnummer  
 (3) formaat datum : DD/MM/JJJJ bij voorbeeld : 02/09/2006  
 (4) optioneel veld  
 (5) enkel aanvinken bij vervanging of herstelling van kilometerteller  
 (6) de ondertekenaar moet bevoegd zijn de onderneming te verbinden

Note that the blue fields contain the original data, the white fields contain the correction data.  
 The following fields can be provided on the correction PDF

field	description	mandatory
User code (gebruikerscode)	See dictionary	<input checked="" type="checkbox"/>
Enterprise (bedrijf)	The company name of the user	<input checked="" type="checkbox"/>
Fax (Mijn faxnummer)	The fax number of the entered organization	<input type="checkbox"/>
Issue ID (Issue ID)	The Issue number that is returned to the professional	<input checked="" type="checkbox"/>
Type (type)	The Issue type code that is returned to the professional together with the type 129. The latter doesn't need to be filled in E.g. 001	<input type="checkbox"/>
VIN (Chassisnummer)		<input checked="" type="checkbox"/>
Odometer Reading value		<input checked="" type="checkbox"/>

Observation Date (datum opname)		<input checked="" type="checkbox"/>
First Use date (datum 1ste inschrijving)		<input type="checkbox"/>
Odometer replaced (Werken aan km-teller)	Has to be checked when Odometer has been replaced or repaired.	<input type="checkbox"/>
Send date (datum van verzending)	The date when the correction PDF is sent to Car-Pass	<input checked="" type="checkbox"/>
Name (naam)	The name of the person who also signed the document. This must be an authorised person of the enterprise or establishment.	<input checked="" type="checkbox"/>
Title (functie)	The title of the person whose name is filled in the name field	<input checked="" type="checkbox"/>
Signature (handtekening)	A manually written signature of the authorised person whose name is filled in the name field	<input checked="" type="checkbox"/>

Note: When there are multiple issues on the same odometerreading: when you correct one issue, the other(s) will also be closed.

---

## 6.3 Changed Issues

When an Odometer Reading is not valid, one or more Issues can be returned through feedback. The complete list of the different error and warning messages will be published separately as an addendum. (see [www.car-pass.be](http://www.car-pass.be) (Professionelen/Professionnels) – section 'Technical Specifications')

With the new version of the Car-Pass application some Issues are removed and some are new.

### 6.3.1 New Issues

The following Issues are new in the Car-Pass system

- Issue 134: Invalid file extension.  
This Issue is returned while sending a file through FTP and the provided file extension is different from the allowed file extensions:
  - TXT
  - XMLThe type of the Issue is “Error” and can’t be solved through a correction. The only way to solve this is by sending a new file with the correct file extension. This will be processed as a new request.
  
- Issue 150: Double observation entered  
This Issue is returned when an Odometer Reading is provided with the same
  - VIN
  - Odometer Reading date
  - Odometer Reading value
  - Entered OrganisationThe type of the Issue is “Error”.
  
- Issue 250: Sender ID Inactive  
This Issue is returned when the Sender of the Odometer Reading has an Inactive status at Car-Pass  
The type of the Issue is “Warning”.  
Data will be accepted as long as the User is active. No correction is needed.
  
- Issue 160: Correction without Issue number  
To correct an Issue an Issue number is necessary.  
When a professional wants to correct an Odometer Reading that is provided and accepted without any Issues, he will have to contact the helpdesk of Car-Pass.  
The type of the Issue is “Error”

### 6.3.2 Issues that can’t be corrected

The following Issues can’t be corrected.

- Type 125: issue-id is empty/does not exist
- Type 126: Invalid Issue-Id
- Type 127: Unknown issue-id (in the db)
- Type 128: Issue-id is already solved or term to correct an issue is passed (90d)
- Type 130: This problem can only be solved by the user who has delivered the original odometer reading
- Type 150: you have already provided the odometer reading
- Type 250: Sender ID Inactive



