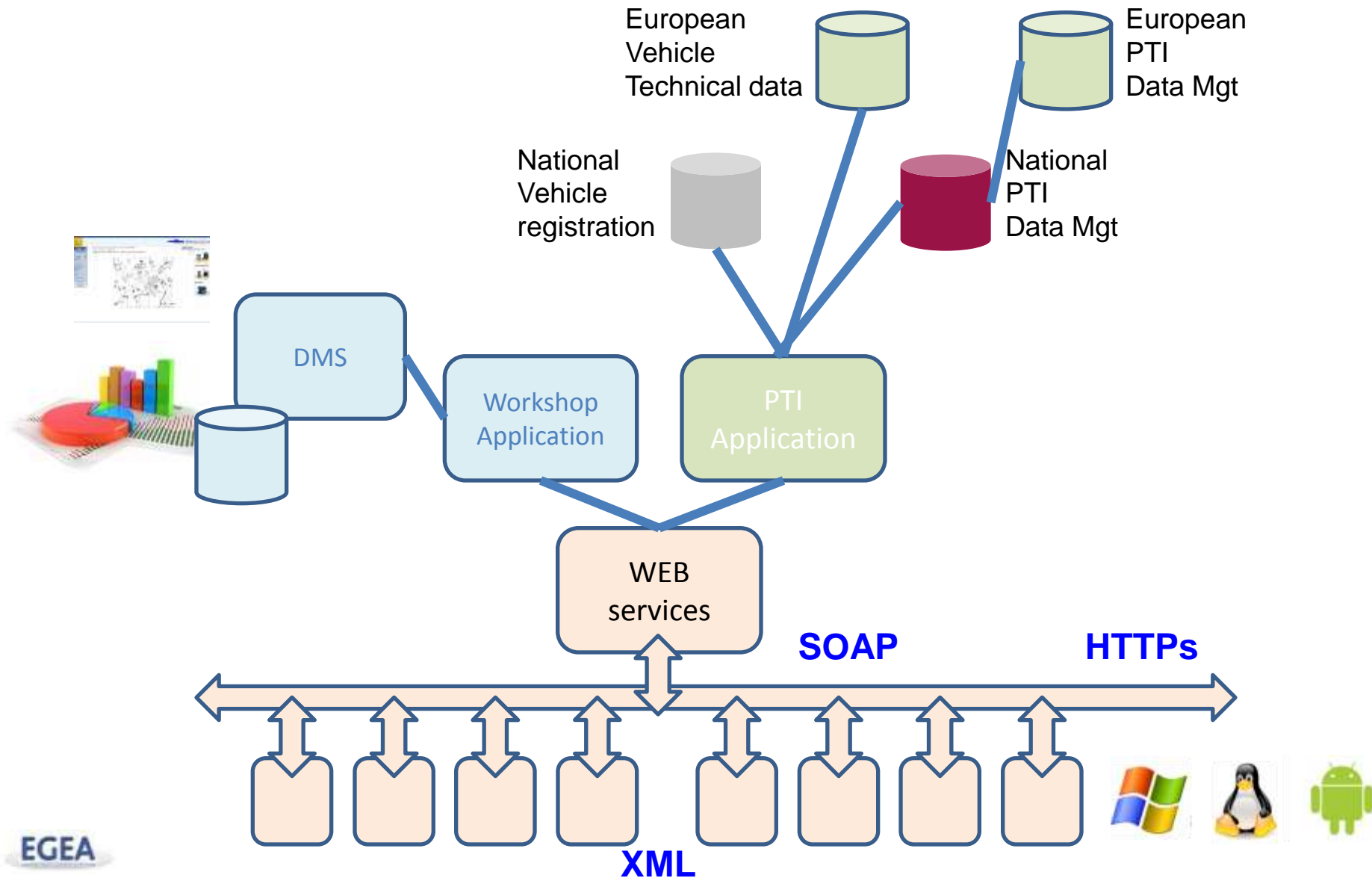


EGEA WG10

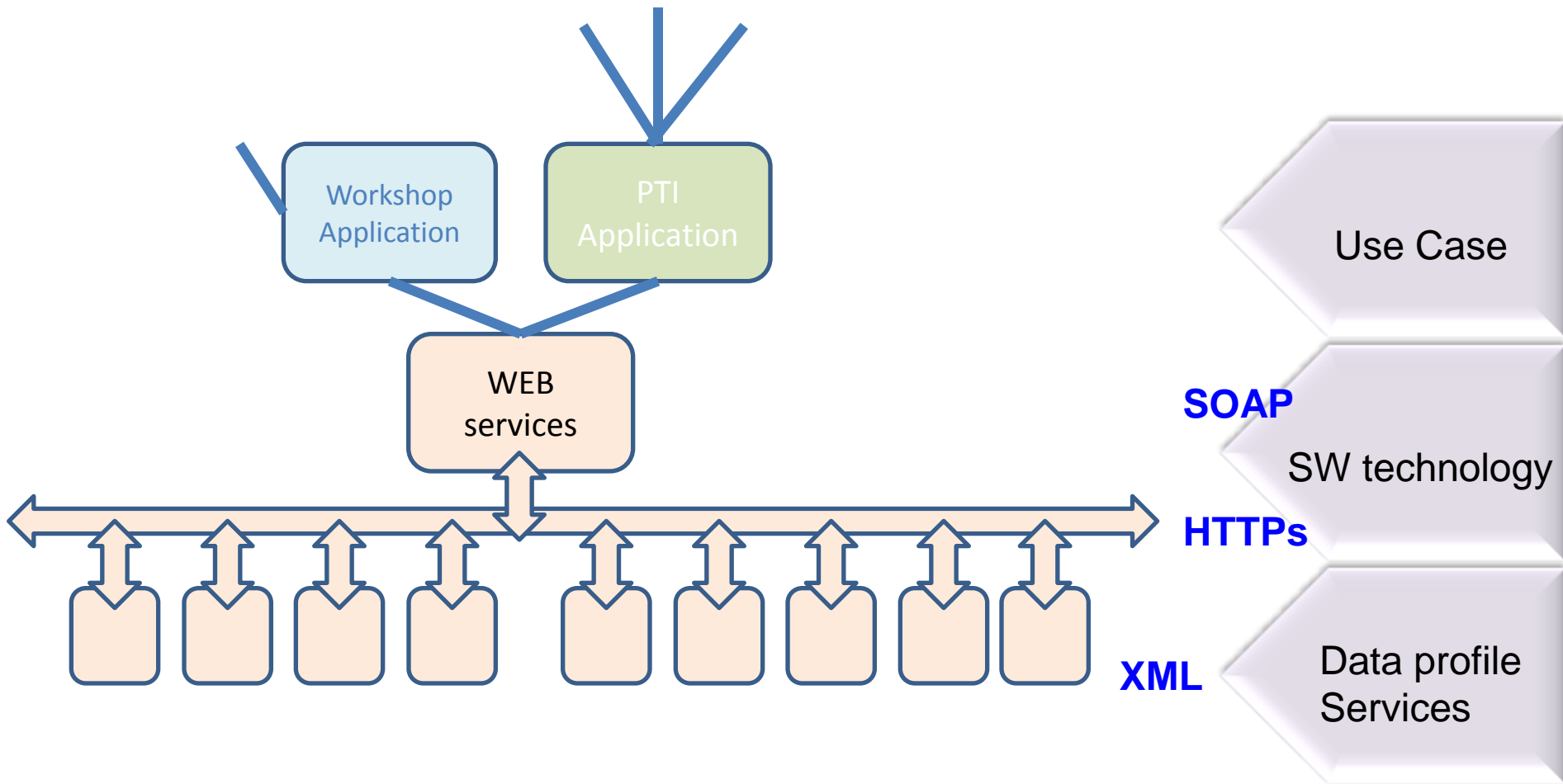
18th -19th Bruxelles



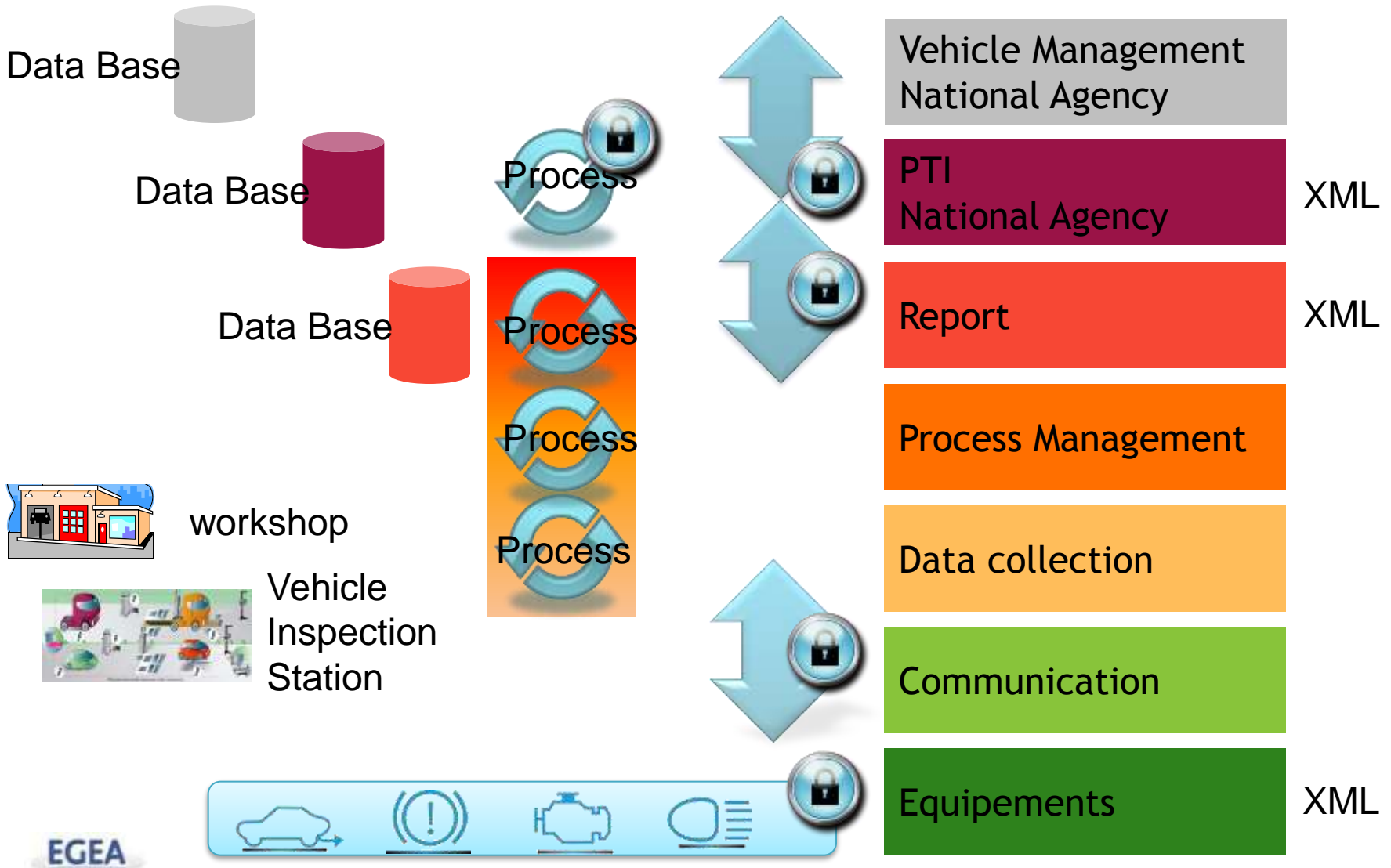
Architecture



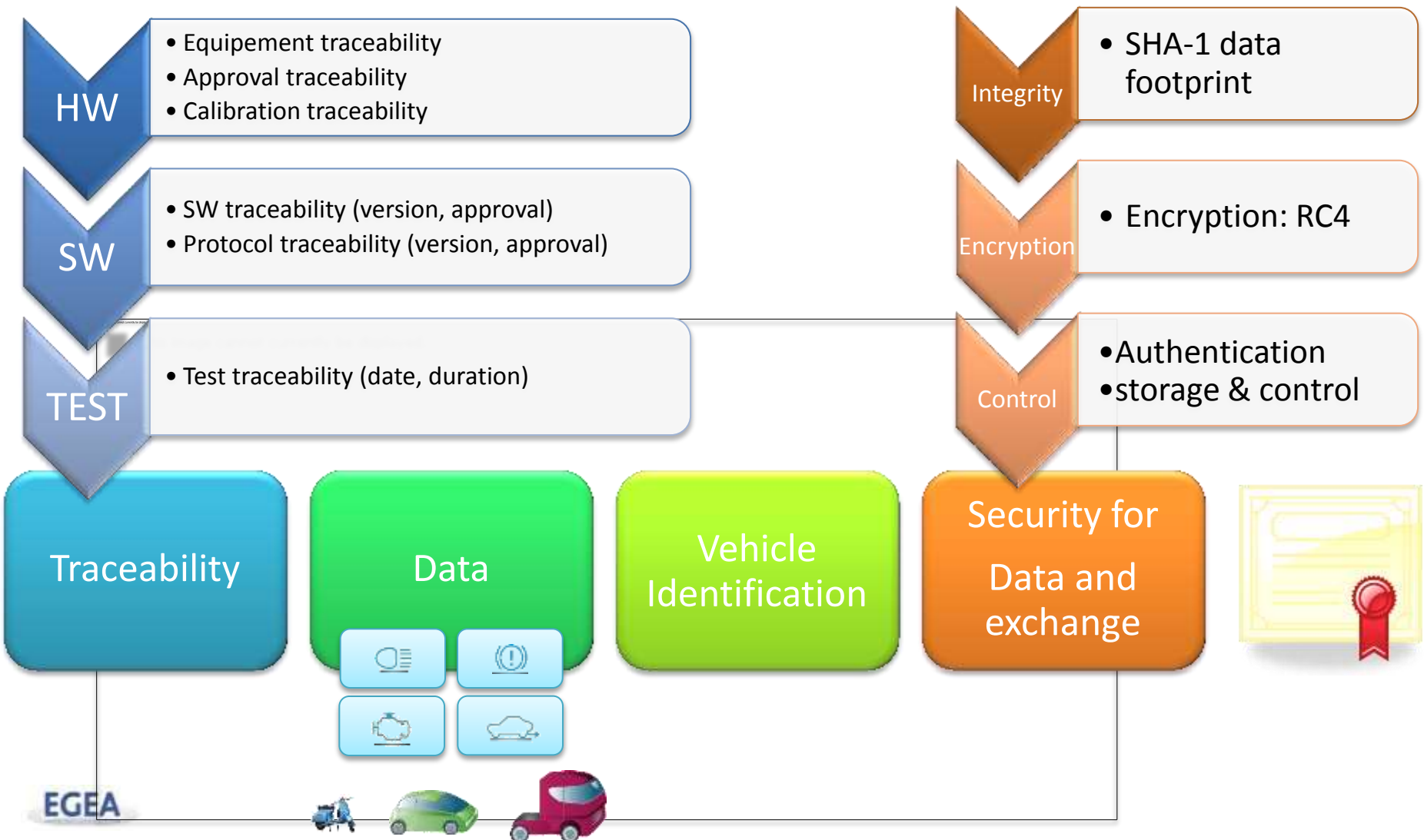
Architecture



Global architecture, various layers



Tracability & Security



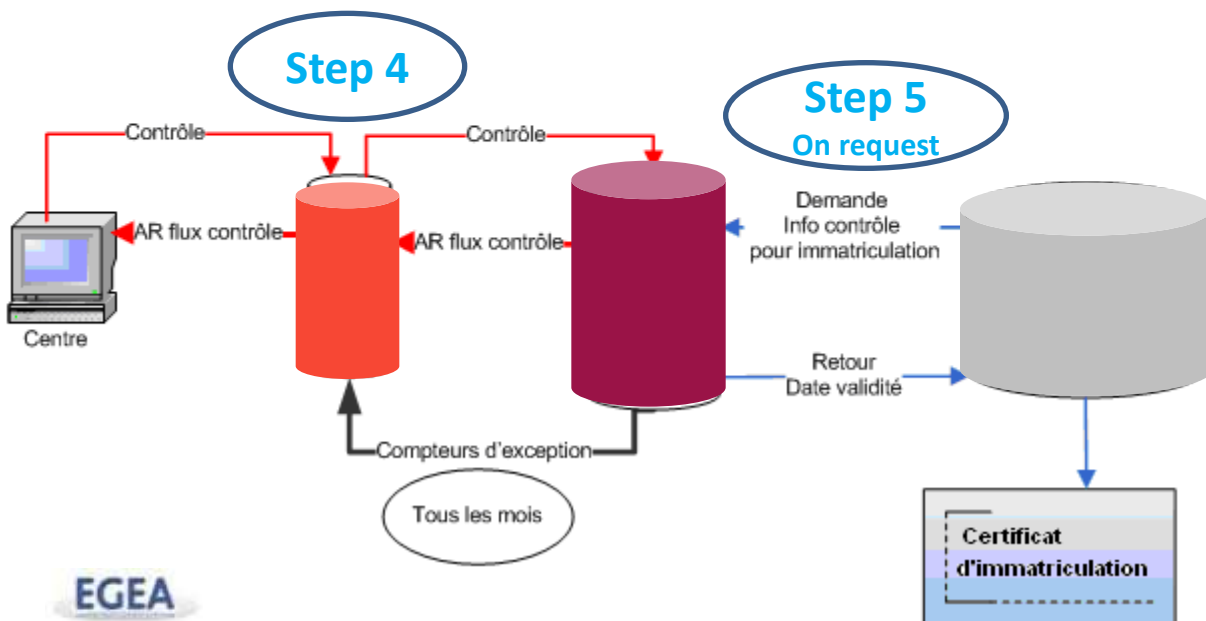
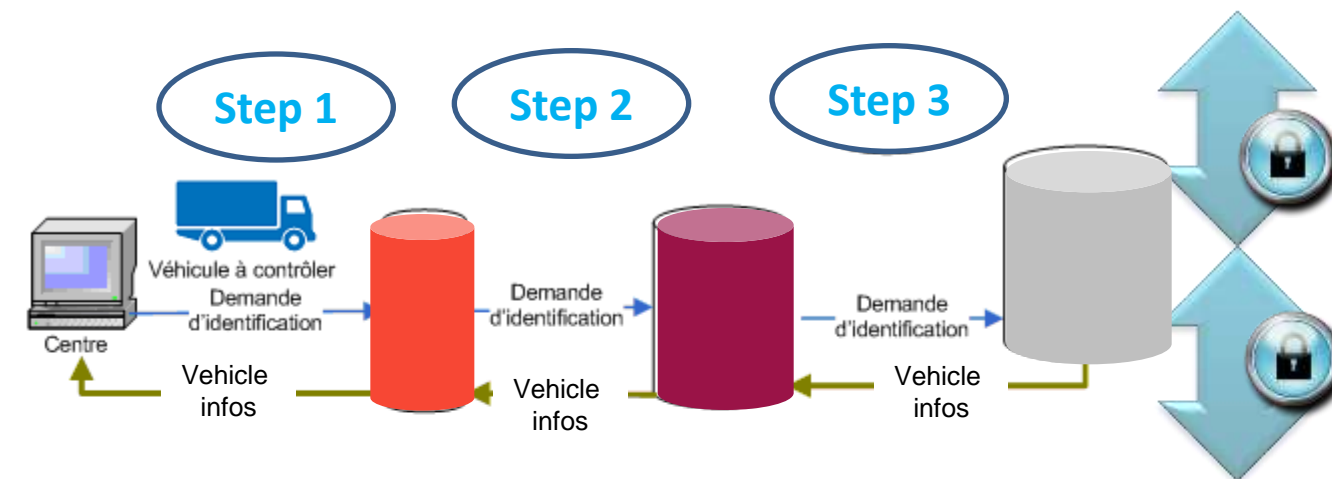
PTI: Vehicle information grabbing from National data base

Use Case

Vehicle Management
National Agency

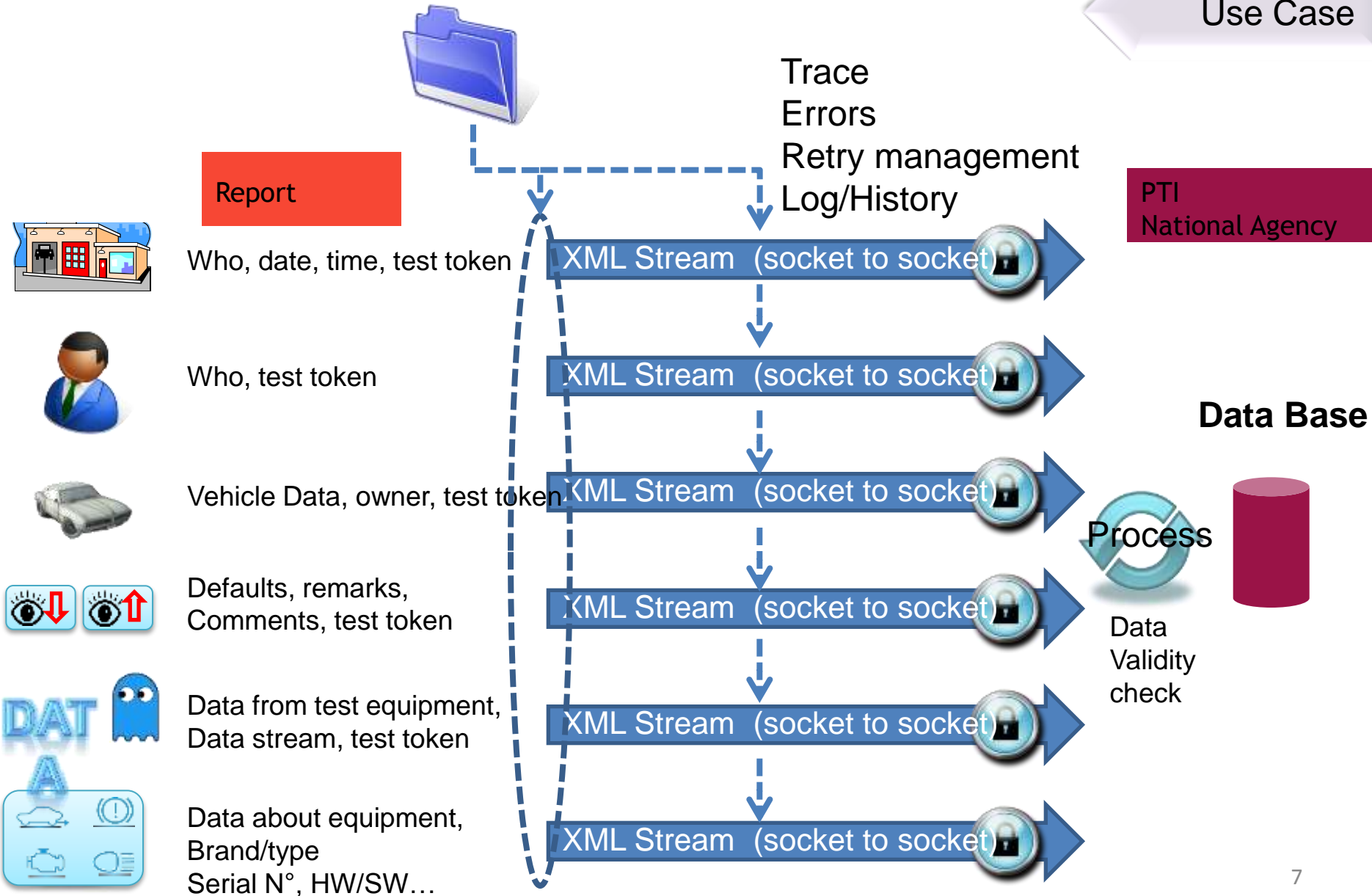
PTI
National Agency

Report



Data collection Protocol

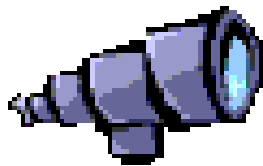
Use Case



Data management: fraud tracking

Use Case

PTI
National Agency



Off line Test
result check



Tracks
Errors
Retry management
Log/History

Log file of exceptions
(human modification)

Statistics

Data Base



From national to European solution

European Commission
ROADWORTHINESS PACKAGE expectation

Use Case

PTI
National Agency

National
Data Base



Data gateway

Types approval

Registrations

Results of
inspections

Road side
Checks base

Electronic
Certificate



SOAP: *Simple Object Access Protocol*

SW technology

SOAP can form the foundation layer of a [web services protocol stack](#), providing a basic messaging framework upon which web services can be built.

This XML based protocol consists of three parts:

- an envelope, which defines what is in the message and how to process it,
- a set of encoding rules for expressing instances of application-defined datatypes,
- a convention for representing procedure calls and responses

SOAP has three major characteristics:

- **Extensibility** (security and WS-routing are among the extensions under development),
- **Neutrality** (SOAP can be used over any transport protocol such as [HTTP](#), [SMTP](#), [TCP](#), or [JMS](#))
- **Independence** (SOAP allows for any programming model).

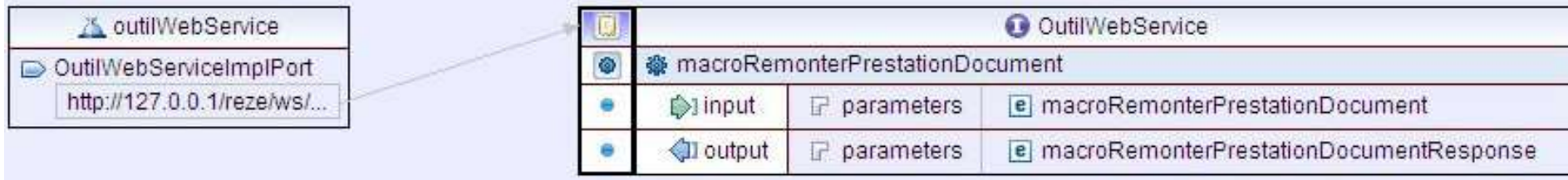
Architecture

Data profile
Services

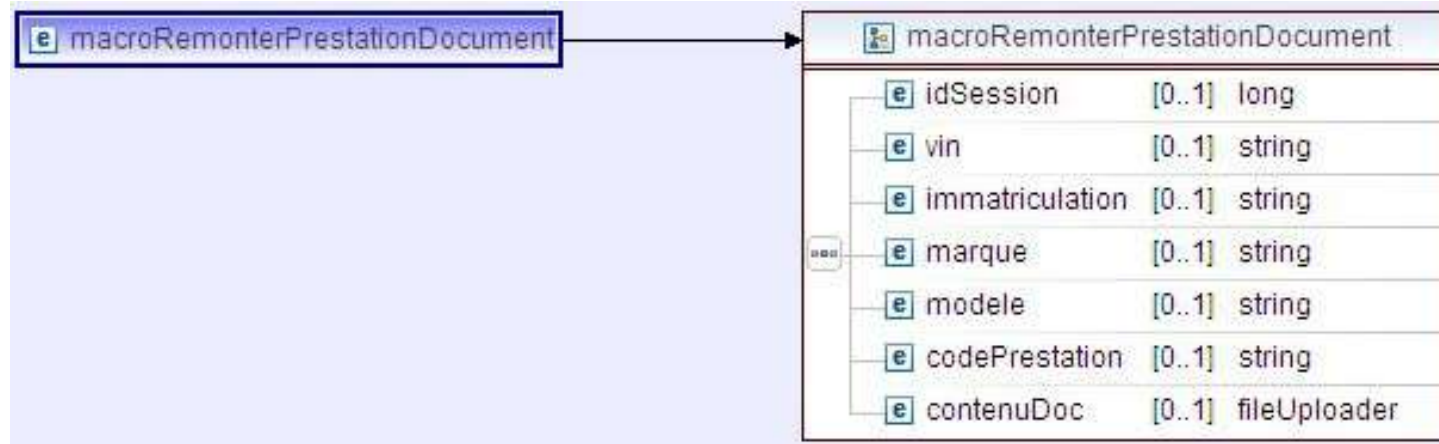
A SOAP message is an ordinary XML document containing the following elements:

Element	Description	Required
Envelope	Identifies the XML document as a SOAP message.	Yes
Header	Contains header information.	No
Body	Contains call and response information.	Yes
Fault	Provides information about errors that occurred while processing the message.	No

Exemple: webservice's name for uploading a document is **macroRemonterPrestationDocument**



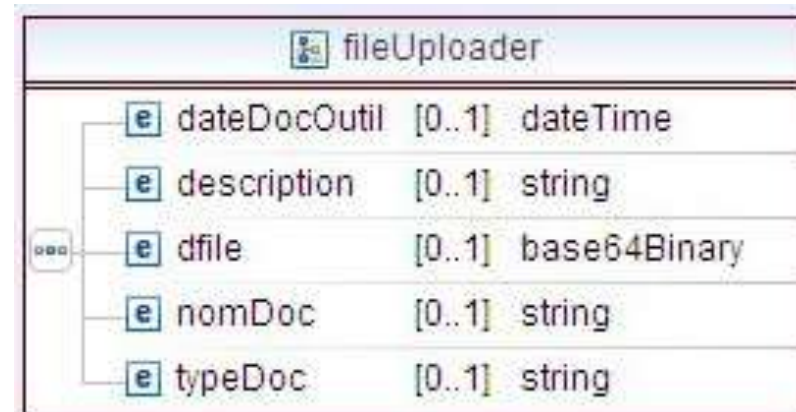
Request contents



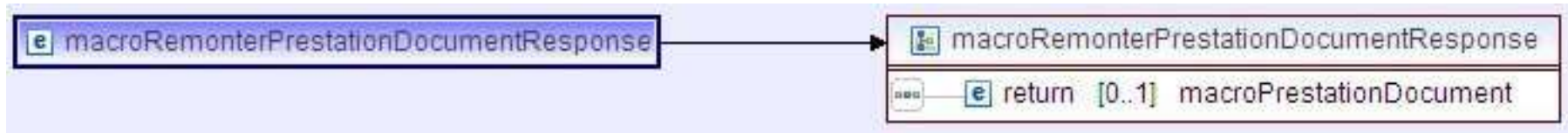
- **idSession** The vehicle session's identifier on the server. If the session is not known, it must be omitted
- **vin** The car's VIN number (17 characters) If the session is not known, the VIN number is mandatory
- **immatriculation** The car's license plate number. (Example: AB1234CD)
- **marque** The car brand. (Example: PEUGEOT)
- **modele** The car model. (Example: 308)

Containing:

- dateDocOutil The date and time of the document to be uploaded
- description A description of the document
- dfile The contents of the file to be uploaded, encoded in BASE-64
- nomDoc The name of the file (example: VF123456789012345_27-05-2013.owx)
- typeDoc The type of the file (example: DiagboxReport)
- Only dfile, nomDoc and typeDoc are mandatory fields.



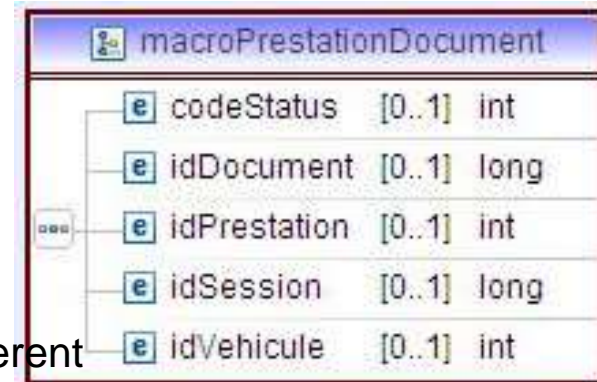
Response contents



Webservice returns an object indicating the result of the operation as well as the object identifiers that have been found or created on the server

codeStatus:

- | | |
|----------|------------------------------------------------------------|
| 0 | Successful operation |
| 1 | Given session does not exist |
| 2 | Given session exists but is closed |
| 3 | Vehicle exists but VIN number and car plate are incoherent |
| 4 | VIN number is not given |
| 5 | VIN number not found and car brand+model not given |
| 6 | Document exists already for this service |



In case of success, the following identifiers are returned:

- | | |
|---------------------|-----------------------------------------------------------|
| idVehicule | Allows to locate the vehicle object on the server |
| idSession | Allows to locate the vehicle session object on the server |
| idPrestation | Allows to locate the service object on the server |
| idDocument | Allows to locate the document on the server |