Telematics System Architecture to deploy the Electronic Transport Document and to improve Emergency Management in the Transport of Dangerous Goods

Research and Development Project of the Federal Ministry of Transport, Building and Urban Development

Interface Specification

Version 01-00-00
Content

1. Document attributes........................................................................................................... 5
   1.1 Purpose......................................................................................................................... 5
   1.2 History......................................................................................................................... 5
   1.3 Project team................................................................................................................ 5

2. Scope .................................................................................................................................. 6

3. System concept..................................................................................................................... 7

4. DGT System Architecture ..................................................................................................10

5. Design decisions..................................................................................................................11

6. Documentation of Web Services .........................................................................................13
   6.1 Introduction.....................................................................................................................13
   6.2 WSDL TP1.....................................................................................................................13
   6.3 WSDL TP2.....................................................................................................................27

7. References..........................................................................................................................31
Figures

Figure 1: System architecture........................................................................................................8
Figure 2: DGT System Architecture...............................................................................................10
Figure 3: WSDL- Trusted Party 1 ....................................................................................................13
Figure 4: registerTransportRequest - Element................................................................................15
Figure 5: registerTransportResponse - Element.............................................................................16
Figure 6: deleteTransportRequest - Element.................................................................................17
Figure 7: deleteTransportResponse – Element................................................................................18
Figure 8: verifyCertificateRequest - Element................................................................................19
Figure 9: verifyCertificateResponse - Element................................................................................19
Figure 10: verifyAccessRequest - Element......................................................................................20
Figure 11: verifyAccessResponse - Element....................................................................................20
Figure 12: getDGTDocumentServiceEndpointRequest - Element..................................................21
Figure 13: getDocumentServiceEndpointResponse – Element.......................................................22
Figure 14: getTrustedCertificatesResponse - Element....................................................................23
Figure 15: getDGTDocumentRequest - Element (Proxy-Mode)......................................................24
Figure 16: getDGTDocumentResponse - Element (Proxy-Mode)......................................................26
Figure 17: WSDL - Trusted Party 2 System.......................................................................................27
Figure 18: getDGTDocumentRequest - Element (Redirect-Mode)..................................................28
Figure 19: getDGTDocumentResponse - Element (Redirect-Mode)..................................................29
Figure 20: getStatusResponse - Element........................................................................................30
Abbreviations

ACL Access Control List, Zugriffssteuerungsliste
CEN European Committee for Standardization
CENELEC European Committee for Electrotechnical Standardization
DoS Denial of Service – It is an attempt to make a machine or network resource unavailable to its intended users.
HTTP Hypertext Transfer Protocol
ITU International Telecommunications Union
PKI Public Key Infrastructure
PSAP Public Service Answering Point
SLA Service Level Agreement
SOAP Simple Object Access Protocol
TLS Transport Layer Security
TP1 Trusted Party 1 – Authentication and authorization system
TP2 Trusted Party 2 – System for providing dangerous goods information
URI Uniform Resource Identifier
UUID Universally Unique Identifier
VIN Vehicle Identification Number
WG Working Group
WSDL Web Service Description Language
XML Extensible Markup Language
XSD XML Schema Definition
1. Document attributes

1.1 Purpose

This is a system description and an interface specification within the research project “Telematics system architecture to deploy the electronic transport document and to improve emergency management in the transport of dangerous goods” of the Federal Ministry of Transport, Building and Urban Development.

The research project is funded by means of the Federal Ministry of Transport, Building and Urban Development.

1.2 History

<table>
<thead>
<tr>
<th>Version</th>
<th>State</th>
<th>Cause</th>
<th>Date</th>
<th>Reviser</th>
</tr>
</thead>
<tbody>
<tr>
<td>01-00-00</td>
<td>Final results</td>
<td>Circulation within WG on Telematics</td>
<td>22.07.2013</td>
<td>Lüpges</td>
</tr>
</tbody>
</table>

1.3 Project team

**Responsibility** | **Partner**
---|---
**Lead** | AlbrechtConsult GmbH
System architecture | Theaterstraße 24
Data modelling | 52062 Aachen
| Dr. rer. nat. Josef Kaltwasser
| Dipl.-Ing (FH) Christian Lüpges
| josef.kaltwasser@albrechtconsult.com
| christian.luepges@albrechtconsult.com

**(IT-)Standards / Trigger** | Harrod Booth Consulting Limited
| Denton, New Park Road,
| Cranleigh, Surrey GU6 7HJ
| United Kingdom
| Dr. Jonathan Harrod Booth
| Jon@harrodbooth.com

**IT-Security/ Service Level Agreements** | otten software GmbH
| Röntgenring 7
| D-40878 Ratingen
| Dr. Marcus Otten
| mo@otten-software.de
2. Scope

The regulatory frameworks for dangerous goods transport on inland transport modes in Europe require certain information to be contained in a (paper) transport document (see paragraph 5.4.1 ADR/RID/AND). Electronic storage of this information as an alternative to paper is, in principle, foreseen but the regulations require functional equivalence to paper without clarifying what exactly this is supposed to mean. This document proposes a system and an applicable interface specification that could replace the information on paper and – if endorsed by the regulations – could provide improved access to dangerous goods information to emergency responders whilst at the same offering benefits for the freight industry by modernising business processes for shippers, by enabling the use of paperless transport.
3. System concept

The principle idea of the proposed system is to replace the access to the information on paper by a query to an electronic data storage system.

The basic assumption is that the vast majority of dangerous goods transports are carried today by transport modes that maintain a connection to an operations centre run by the transport company. This connection may be sporadic or it may even today already be quasi-permanent (e.g. for transport tracking).

It is further assumed that a majority of e.g. emergency responders at incident sites are also connected to a command & control centre that manages their operations.

These two assumptions create the opportunity to convey the electronic exchange of information required on-site to an electronic exchange between the two back-office systems. The advantage of this approach is that it is not required to standardise and certify a – potentially huge – number of mobile devices in the field. The actual communication between on-site systems and their back-offices effectively does not require standardisation in this setup. What the system requires instead is a standardised electronic exchange between the two back-office systems, which can be established by using proven solutions and standards from domains like eCommerce – but also Public Sector Information.

In order to allow access to specific information, the emergency responder of course needs to be able to specify exactly for which specific transport vehicle he needs information. The concept therefore consists of a second element – called a “trigger” for the purpose of this document – that scopes information access to a particular situation. Such a trigger may be created by a vehicle and conveyed directly to the emergency response centre – as would e.g. be the case when an eCall or a TARV emergency call message is triggered – or it may be created by the emergency responder on-site, based on, for example, visual observation.

Figure 1 depicts the system in a schematic overview. Blue lines indicate connections that are expected to exist already (i.e. transport mode – operations centre, and emergency responder – command & control centre). The red connection between emergency responder and vehicle is only included to highlight that technical interoperability between field devices is not needed in the approach.
The new information flows are depicted in green. The PSAP (or the C&C centre of control officers) will need to physically perform the access to the Dangerous Goods transport data via the backbone service indicated by the shipper.

Note that this does by no means imply who actually operates the service. It is assumed that large companies are happy to provide the service themselves, thus avoiding to have to trust in third parties handling their sensitive data (e.g. large lorry on the left in the figure). Small companies might fear the cost and complexity of operating high-availability services on the Internet and might prefer to delegate this task to dedicated Service Providers (e.g. small lorry on the right).

The C&C centre needs to fulfil two requirements when accessing the service:

- It must be able to prove that the authority issuing the request is in principle entitled to perform the access – this includes a general proof of authorisation. This has to be checked by a Central Management Service which manages an access control list (grey box).
- It must provide a justification dataset that provides the reason and context for access (emergency, control) and provides for Non-Repudiation; this automatically includes the ability of being able to specify exactly the shipment for which data is requested.

If a transport mode carries only one shipment, the access interface can be easily provided by an exact specification of the service interface plus a transport ID. In case a vehicle carries multiple shipments, the service of the shipper must provide the access details of the individual shipments.

The architecture provided does not depend on information about the dangerous goods being carried on the vehicle (e.g. in an On-Board Unit). It should be noted though that it also by no
means prevents such information from being on-board, which would lead to improved reliability of access in case of communication black spots. The impact of this additional option on standardisation and certification requirements will nevertheless have to be assessed based on technology available. In all case such an additional local communication options should be seen as an optional addition to the back-office path, not as an alternative replacement.
4. DGT System Architecture

The following diagram gives an overview over actors involved within the system architecture. In any case authorised command and control centres (CCC) and authorities directly request dangerous goods transport related documents (electronic transport document, proofs and certificates according to RID, ADR and ADN) from Trusted Party 2 (Content Server) and from Trusted Party 1 (Management Server) in specific cases. Hereafter all data related to dangerous goods transport are defined as DGT data.

Figure 2: DGT System Architecture

The possibilities a CCC can get specific information (TransportID (UUID) und des TP2-Service-Endpoint (URI)) necessary to request DGT data from TP2 are twofold:

- Via a trigger (e.g. eCall HGV or TRAV) that provides these information directly,
- Via a casual observer onsite that notifies an accident and reports defined visual characteristics (e.g. numberplate, UIC wagon number or ENI number), with its help, the CCC is able to determine these information indirectly.

The figure shows the entire system with all interactions though carrier related actions in order to register a transport on TP1 via TP2 and to store a transportID (UUID) and TP2-Service-Endpoint (URI) are not focus of this system specification. These interactions are on hands of those who want to provide a TP2 system.

All systems communicate via secure point-to-point connections, i.e. they are secured by transport layer security (TLS) with client-server authentication.

Certificates used to secure the connections are distributed by one or more existing public key infrastructures, which were classified as trusted in context to a dangerous good telematics infrastructure.
5. Design decisions

The following list of design decisions has been agreed within the WG on Telematics on 3rd and 4th of June 2013:

- No regulations for Member States or emergency responders. Their internal behaviour and how they make use of the system is entirely up to them.
- The access interface itself will be operated using Web Services technology e.g. [WSDL] & [SOAP] that enables machine-to-machine access to information via the Internet.
- The required security aspects can be addressed by appropriate IT-security mechanism, in particular the use of end-to-end security ([HTTPS]) based on digital certificates ([X.509])
- Existing Public Key Infrastructures (PKI) will be (re-)used. This implies a central registry where certificates are registered and assigned to roles.
- Certificates are associated to organisations, not to individuals. This may have impact of organisational procedures and does have an impact on non-repudiation.
- Access is not distinguished on content (e.g. no dedicated access right for particular Dangerous Goods classes).
- Certificates are used for securing the end-to-end link and for digital signatures of the content. Data is not encrypted outside the communication channel
- Services can (and shall) be certified in the future to ensure interoperability. There is a need to consider the establishment of the organisational framework for accredited certification organisations.
- The Service Level of the Trusted Party 2 services (TP2 services) will not be constantly monitored.
  - The basic legal assumption is the equivalence to the current (paper) situation: the carrier is responsible for the service to work when needed;
  - Nevertheless, suitable service levels – ideally based on internationally accredited standards – shall be specified, but no SLAs;
  - There should be provisions regarding DoS attacks in the service level descriptions.
- TP2 systems shall register with a central registration service (→ TP1) operated by an sovereign instance
- Virtual Private Networks (VPN), e.g. eTESTA shall not be required for the backbone, but associated IT-security issues must be taken into account
- The service interfaces shall be fully specified (WSDL & XSD). The actual development of the services will be WSDL-first.
- The system specification shall contain self-inspection methods in order to support migrations paths in case of future evolution.
- A logging interface shall provide access to evidence (details, e.g. storage period, to be determined).
- The system shall support two different types of access scenarios:
- Access with knowledge of service end point and vehicle ID
  - e.g. "electronic trigger" via eCall, TARV, etc.;

- Access with context knowledge only (e.g. number plates, UIC Wagon Number, ENI-Number…) - e.g. "casual observer". This implies services to look up service end point and Transport-ID depending on descriptive parameters, depending on mode of transport.

- One single successful data access shall provide ALL DGT data needed for emergency response / control. The carrier has to have the full data of the goods transported – it is not enough to have a reference to another system operated, e.g. by the consignor system.

- The data structure should reflect the organising principles used currently for paper documents for the different modes of transport (e.g. by wagon for trains).
6. Documentation of Web Services

6.1 Introduction

In the following, the WSDL of the systems TP1 and TP2 are presented and explained. The SOAP-Messages, which are exchanged during the execution of several web service methods, respectively use an ordinary element for request and response. Its structure is based on a referring XML schema [XMLschema].

These elements contain a version attribute. In case of backwards-compatible changes (e.g. adding new methods, extension of parameters with optional information) the decimal place of the version number will be increased. In case of non-compatible changes (e.g. additional mandatory or changed attributes) the main version will be increased and the namespace of the XML schema will be extended by the date of change (basic namespace/yyyy/mm/dd).

But it has to be noted that a change of the main version requires a change of the client software.

6.2 WSDL TP1

6.2.1 Overview

This TP1 WSDL comprises all methods, which are provided to other systems by the help of a TP1 system.
6.2.2 registerTransport

This is a method, which is invoked by the carrier in order to indicate the beginning of a transport of dangerous goods. The TP1 system saves information on which certificate had registered which transport with which mode of transport.

Request: registerTransportRequest
Response: registerTransportResponse
registerTransportRequest: Contains the unique ID (transportID), the descriptive data of the transport modes (lorryMetaData, trainMetaData, shipMetaData) and the endpoint of the responsible TP2 system, which provides the dangerous goods documents.
registerTransportResponse: Contains a numeric return value of the method (0 = successful registration), a plain text for the return value, the point of time at which the response was produced as well as an optional signature of resultCode, resultMessage and issued by the TP1 system.

6.2.3 deleteTransport

This is a method, which is invoked by a TP2 system in order to indicate that a transport of dangerous good has been completed.

Request: deleteTransportRequest
Response: deleteTransportResponse
Figure 6: deleteTransportRequest - Element

deleteTransportRequest: Contains, which transport of dangerous goods (transportID), in which point of time (timestamp) has been completed, signed by the TP2 system, which has issued the deletion.
6.2.4 verifyCertificate

This is a method, which is applied by CCCs, authorities and TP2 in order to identify whether a certificate is valid or not.

A certificate is valid, if

- the time of inquiry is within the period of certificate validity
- the certificate is not locked
- the certificate is known through a previous process of registration

Request: verifyCertificateRequest
Response: verifyCertificateResponse
verifyCertificateRequest: DER-encoded X509v3 certificate as binary object, which validity has to be verified.

VerifyCertificateResponse: Boolean:
- true: certificate is valid
- false: certificate is not valid

6.2.5 verifyAccess
This is a method, which is invoked by TP1 and TP2 in order to identify whether a certificate
- is valid and
- has the requested authorisation.

Request: verifyAccessRequest
Response: verifyAccessResponse
verifyAccessRequest: Contains for which certificate the authorisation is requested, whether it is a global right (right) or depends on the type of data (dtright) and for which type of data the authorisation shall be valid.

verifyAccessResponse: Boolean 
true: authorisation granted
false: authorisation denied

6.2.6 getDGTDocumentServiceEndpoint
This is a method, which determines by the help of transport meta data,
- if a dangerous goods transport is registered (previous registerTransport entry regarding this transport),
- under which endpoint they can be queried.
Request: getDGTDocumentServiceEndpointRequest
Response: getDGTDocumentServiceEndpointResponse

Figure 12: getDGTDocumentServiceEndpointRequest - Element
getDGTDocumentServiceEndpointRequest: Contains to which transport data is requested. Either by the help of meta data regarding different transport modes or optionally by indicating the transportID.

Figure 13: getDocumentServiceEndpointResponse – Element

getDocumentServiceEndpointResponse: Contains the communication endpoint of the TP2 system, where the dangerous goods documents can be requested, the transportID, a result code and the message of the response.

6.2.7 getTrustedCertificates
This is a method, which may be invoked by CCCs, TP2 systems and further external systems in order to draw a list of supported/trusted certificates, which shall be accepted for authentication.

Request: none
Response: getTrustedCertificatesResponse
getTrustedCertificatesResponse: liefert eine Liste von vertrauenswürdigen X509v3-Zertifikaten, die auf TP1 registriert sind, als DER-codierte Binär-Objekte.

6.2.8 getDGTDocument

This is a method, which enables a CCC or an authority to recall the dangerous goods data without any direct connection to the TP2 system (proxy mode), if they are restricted in using the internet. Therefore only a connection to one TP1 is required.

Request: getDGTDocumentRequest
Response: getDGTDocumentResponse
Figure 15: getDGTDocumentRequest - Element (Proxy-Mode)
getDGTDocumentRequest: Contains the information

- for which transport (transportID) the documents are recalled,
- who is recalling it (requester),
- the signature [xmlDsig] of the CCC or the authority system, which requested the retrieval of the documents,
- restrictions, which data shall be provided. The dangerous goods data from the correspondent schema is grouped into the different data types. These can be requested separately (dgFolder, dgLoadInformationDynamic, dgTransportInformationDynamic) or altogether, if no data type is indicated.
- the endpoint URL of the TP2 system, which provides the data (in case this information is available and does not to be determined by the TP1 system)
- the certificate of the CCC or the authority, on behalf of which TP2 is recalled.
getDGTDocumentResponse: Provides information about the dangerous goods data schema as well as the status of the recall as numeric code and plain text.
6.3 WSDL TP2

6.3.1 Overview

This WSDL embraces all methods, which are provided to other systems by a TP2 system.

6.3.2 getDGTDocument

This is a method, which enables a CCC or an authority to recall the dangerous goods data from a TP2 system (redirect mode). If they are restricted in using the internet, TP1’s proxy mode service is recommended (see 6.2.8).

Request: getDGTDocumentRequest
Response: getDGTDocumentResponse
Figure 18: getDGTDocumentRequest - Element (Redirect-Mode)
getDGTDocumentRequest: Contains the information

- for which transport (transportID) the documents are recalled,
- who is recalling it (requester),
- the signature [xmlDsig] of the CCC or the authority, which requested the retrieval of the documents,
- restrictions, which data shall be provided. The dangerous goods data from the correspondent schema is grouped into the different data types. These can be requested separately (dgFolder, dgLoadInformationDynamic, dgTransportInformationDynamic) or altogether, if no data type is indicated.

Figure 19: getDGTDocumentResponse - Element (Redirect-Mode)
getDGTDocumentResponse: Provides information about the dangerous goods data schema as well as the status of the recall as numeric code and plain text.

### 6.3.3 getStatus

This method provokes a self-disclosure of the TP2 system.

Request: none
Response: getStatusResponse

![Diagram of getStatusResponse - Element](image)

Figure 20: getStatusResponse - Element

getStatusResponse: The TP2 System has to provide the version of the WSDL, the version of the DGTSchema, a system identifier and the server certificate.
7. References

[WSDL] Web Services Description Language (WSDL) Version 2.0; W3C Recommendation 26 June 2007; http://www.w3.org/TR/soap

[SOAP] SOAP Version 1.2; W3C Recommendation 27 April 2007; http://www.w3.org/TR/soap


