

# EDM WasteX Interface v1.08

## Description

SIMPLE EDI FOR WASTE SHIPMENTS

27 Apr 2023

© 2023 Environment Agency Austria (EAA)

Commissioned by the Austrian Federal Ministry of Environment (BMK)

# Table of Contents

1	Introduction .....	3
1.1	PDF Description and HTML Annex .....	3
1.2	Notation.....	3
1.3	Background.....	4
1.4	Purpose of the EDM WasteX web service interface .....	6
1.5	Purpose of the interface description document.....	6
1.6	Intended audience.....	6
1.7	Contact .....	6
2	Interface description.....	7
2.1	WasteX “Basic” EDI protocol .....	7
2.2	Client/service .....	7
2.3	Web service environments and endpoints .....	7
2.4	Payload – user message vs. signal message .....	8
2.5	Unidirectional vs. bidirectional usage .....	8
2.6	Web service operations .....	10
2.7	Polling.....	11
2.8	Codelists.....	13
2.9	Party identification.....	15
2.10	EDM WasteX EDI practicalities .....	17
2.11	Preparatory step 1: Client EDI node registration .....	29
2.12	Preparatory step 2: Enabling access by/for parties .....	30
2.13	Preparatory step 3: Party IDs in the notification .....	30
2.14	Authentication.....	31
2.15	EDM WasteX interface constraints.....	35
2.16	EDM WasteX v1.08 limitations .....	35
3	Terms and definitions.....	36
4	Acronyms.....	39
5	References.....	42
6	Attribution.....	44
7	Team and contributions .....	45

# 1 Introduction

## 1.1 PDF Description and HTML Annex

The interface description consists of two parts:

1. This PDF document
2. An Annex in HTML format

The PDF description provides an introduction and details such as on authentication.

The [HTML] Annex contains several auto-generated contents, such as XML format descriptions generated from annotated [XSD] files and web service descriptions generated from annotated [WSDL] files.

The Ministry of Environment (BMK) publishes this specification for download at [edm.gv.at](http://edm.gv.at).  
Online access to the latest version of the specification is available at <https://test.umweltbundesamt.at/dataharmonisation/spec/wastex/index.html>.

## 1.2 Notation

The text in this specification document contains the following:

1. **Terms** with a specific meaning in the context of the WasteX specification  
See Section 3 on page 36
2. **Acronyms**  
See Section 4 on page 39
3. **References**, written in [square brackets]  
See Section 5 on page 42

The key words MUST, MUST NOT, REQUIRED, SHALL, SHALL NOT, SHOULD, SHOULD NOT, RECOMMENDED, MAY, and OPTIONAL in this specification are to be interpreted as described in [RFC2119].

## 1.3 Background

### 1.3.1 Waste shipment business processes

Transboundary movements of waste, also called shipments of waste, carry the risk of adverse effects to human health and the environment.

The [BC] is an international treaty aimed at **protecting human health and the environment** against such adverse effects by strict control. The [WSR] is an EU Regulation with the same goal. The [WSR] contains additional control provisions defined for the EU on top of the [BC] provisions.

Among the core provisions of the [BC] and the [WSR] are the following business processes involving information interchanges between EOs and CAs (B2A):

<b>Waste shipments requiring prior written notification and consent</b>	
<p><b>BP_AMBER_CONSENT</b> Notification and consent process</p>	<p>Steps such as the following:</p> <ul style="list-style-type: none"> <li>• EO submits notification to CA, Art. 4 [WSR]</li> <li>• (Optional): CA requests additional information from EO, Art. 4(3) [WSR]</li> <li>• CA acknowledges receipt of a properly <i>carried out</i> notification to EO, Art. 7(1) [WSR]</li> <li>• CA acknowledges receipt of a properly <i>completed</i> notification to EO, Art. 8(2) [WSR]</li> <li>• CA issues consent, consent with conditions or objection to EO, Art. 9 [WSR]</li> </ul>
<p><b>BP_AMBER_INFO</b> Information process for each individual shipment under a consent ("Movement document" process)</p>	<p>Steps such as the following:</p> <ul style="list-style-type: none"> <li>• MA: EO announces shipment to CAs and other EOs, Art. 16(b) [WSR]</li> <li>• (Optional): EO provides consent and individual shipment information to CAs upon inspection and customs controls, Art. 16(c) [WSR]</li> <li>• CoWR: EO confirms receipt of waste to CAs and other EOs, Art. 16(d) [WSR]</li> <li>• CoWT: EO confirms completion of the treatment of shipped waste to CAs and other EOs, Art. 16(e) [WSR]</li> </ul>
<b>Waste shipments not requiring consent – "general information requirements"</b>	
<p><b>BP_GREEN_INFO</b> Information process for an individual shipment not requiring consent ("Consignment information" process)</p>	<p>Steps such as the following:</p> <ul style="list-style-type: none"> <li>• EO prepares information to accompany the shipment of waste, Art. 18(1) [WSR]</li> <li>• EO provides prepared information to CA upon request, Art. 18(2) [WSR]</li> </ul>

### 1.3.2 Electronic waste shipment information handling in Austria

**Austria** has a single designated **CA for shipments of waste**, the **Ministry of Environment (BMK)**, department V/1. Since 2006, the Austrian CA uses **[EDM]** to manage waste shipment data and processes. **[EDM]** is an interconnected system of IT applications and databases by the Austrian public administration to support environment-related business processes, and is part of the Austrian e-Government services.

EOs can exchange structured electronic waste shipment information with the Austrian CA via the **[EDM] web application**, and/or via an **[EDM] EDI web service**.

Since the end of 2017, **[EDM]** has an **EDI interconnection with [VeVA-Online]**, the e-Government solution used by the **CAs of Switzerland**. **[EDM]** and **[VeVA-Online]** exchange waste shipment information by using the **[EUDIN]** specifications and infrastructure, in particular the *[EUDIN] Message Broker*.

Since 2006, the Austrian CA enters all waste shipment data that it receives in other ways than structured electronic data – such as by e-mail, fax, or post – as structured data into the **[EDM]**.

Until 2021, structured electronic data exchange between Austrian EOs and the Austrian CA had been an option for EOs, but not an obligation.

### 1.3.3 Amended legal requirements as of 2022

With **[AWG 2021]**, the **2021 Circular Economy amendment to the Austrian Federal Act on sustainable waste management**, Austrian **EOs become obliged to exchange waste shipment information as structured electronic data** with the Austrian CA, i.e., to use the **[EDM]** web application or to send waste shipment data via EDI, starting from March 2022.

As a result, the demand for waste shipment B2A EDI increases in Austria. **The Austrian CA meets this demand by offering the new [EDM WasteX] EDI web service interface.**

## 1.4 Purpose of the EDM WasteX web service interface

The BMK provides the [EDM WasteX] EDI web service interface as a means for EOs to exchange information with the BMK pursuant to the **BP\_AMBER\_INFO** processes (see 1.3.1).

The following are the most important B2A transmissions that the web service supports:

1. **MovementAnnouncement**

Prior information regarding the actual start of a shipment, pursuant to Art. 16(b) [WSR]

2. **ConfirmationOfWasteReceipt:**

Confirmation of receipt of the waste by the facility, pursuant to Art. 16(d) or Art. 15(c) [WSR]

3. **ConfirmationOfWasteTreatment**

Confirmation of the completion of recovery or disposal of waste by the facility, pursuant to Art. 16(e) or Art. 15(d) [WSR]

From a technical point of view, the [EDM WasteX] interface enables EDI between the following two EDI nodes:

1. EDI node used by an EO, such as ERP (enterprise resource planning) solution used by an EO
2. [EDM], the EDI node used by the BMK, i.e., by the Austrian CA

The Austrian CA offers EDI over the [EDM WasteX] web service interface to EOs as an alternative to using the online forms of the [EDM] waste shipment web application (“eVerbringung”).

## 1.5 Purpose of the interface description document

This description document provides guidance on the establishment, implementation and operation of a connection to the [EDM WasteX] interface. It complements the [WasteX] specification, such as by describing specifics of the [EDM WasteX] interface.

## 1.6 Intended audience

The main intended audience of this specification is **IT personnel**, i.e., IT business analysts, developers, and testers, working for software/solution/service providers that implement an EDI connection to the [EDM WasteX] interface, or consider doing so.

In addition, this description targets **waste management domain experts**, and **decision makers at software/solution/service providers** in the field of waste management.

## 1.7 Contact

For enquiries about [EDM WasteX] refer to [edm-helpdesk@umweltbundesamt.at](mailto:edm-helpdesk@umweltbundesamt.at)

## 2 Interface description

### 2.1 WasteX “Basic” EDI protocol

The [EDM WasteX] web service interface **complies with** the [WasteX] “Basic” EDI protocol.

The authors of the [WasteX] “Basic” EDI protocol pursued the following goals:

1. Definition of an edm.gv.at data interface for business-to-authority data transmissions to the Austrian competent authority for shipments of waste
2. Reusability and reimplementability. For example, other competent authorities for shipments of waste can easily offer an identical data exchange interface

### 2.2 Client/service

The [WasteX] “Basic” EDI protocol enables EDI between nodes A and B for the following configurations:

	A	B
1	client only	web service only
2	web service only	client only
3	web service & client	web service & client

Note that for the time being the [EDM] EDI node supports “web service only” functionality.

This has the following consequences on preparing software for EDI with [EDM]:

1. Client functionality is sufficient – the software does not need to feature web service functionality
2. For the time being: Web service functionality is not sufficient - the software needs to feature client functionality
3. For the time being: If the software features both client and web service functionality, only the client functionality is relevant to the EDI connection with [EDM], as [EDM] cannot use the connected EDI node’s web service functionality

### 2.3 Web service environments and endpoints

To external developers, there are two [EDM WasteX] environments and endpoints:

1. Production environment:  
[https://secure.umweltbundesamt.at/wastex/wastex\\_webservice](https://secure.umweltbundesamt.at/wastex/wastex_webservice)  
[https://secure.umweltbundesamt.at/wastex/wastex\\_webservice/wastex.wsdl](https://secure.umweltbundesamt.at/wastex/wastex_webservice/wastex.wsdl)
2. Pre-production environment (stage):  
[https://vprod.umweltbundesamt.at/wastex/wastex\\_webservice](https://vprod.umweltbundesamt.at/wastex/wastex_webservice)  
[https://vprod.umweltbundesamt.at/wastex/wastex\\_webservice/wastex.wsdl](https://vprod.umweltbundesamt.at/wastex/wastex_webservice/wastex.wsdl)

The pre-production environment is for testing. It resembles the production environment, and features database contents similar to those of the production environment.

## 2.4 Payload – user message vs. signal message

Pursuant to the [WasteX] message format specification, the [EDM WasteX] interface distinguishes two types of payload:

### 1. User message

A user message contains waste shipment related data, such as

- **MA** (movement announcement, Art. 16(b) [WSR])
- **CoWR** (confirmation of waste receipt, Art. 16(d) or Art. 15(c) [WSR])
- **CoWT** (confirmation of the completion of waste treatment, Art. 16(e) or Art. 15(d) [WSR])

Note that each user message contains a **single** MA, CoWR, CoWT content (see also 2.10.8 on bulk transmissions). For example:

- A user message cannot contain more than one MA, i.e., it can contain the shipment announcement for a single combination of notification number and sequential number, but not for multiple such combinations (applies similarly to CoWR, CoWT, etc.)
- A user message can contain only one of MA, CoWR, CoWT, but not a combination of these contents

### 2. Signal message

A signal message has a supporting role in EDI, and carries information on the following:

- information of whether or not a received user message was accepted or rejected at the receiving EDI node
- report resulting from the receiving EDI node's formal validation of the received user message

## 2.5 Unidirectional vs. bidirectional usage

### 2.5.1 Introduction

Those that implement a connection to [EDM WasteX] can choose from one of the following two modes of connection:

#### 1. Unidirectional connection

This refers to a type of connection with which waste shipment payload flows in one direction only, namely from the EO to the CA, i.e., from the EO's EDI node to the [EDM]

#### 2. Bidirectional connection

This refers to a type of connection with which waste shipment payload flows in both directions, from EO to CA and vice versa, i.e., to the [EDM] and from the [EDM]

## 2.5.2 Guidance on choosing the appropriate connection type

Clearly, user message transmissions to the [EDM], realizing EO to CA sharing of waste shipment information, are the main use case for [EDM WasteX].

User message transmissions in the opposite direction – from the [EDM] to the EO's EDI node – can occur upon the following occasions:

1. Entering or editing of waste shipment information in the online forms of the [EDM] "eVerbringung" web application, either by an EO or by the CA
2. New or updated waste shipment information that the Austrian CA receives via [EUDIN] A2A information exchange. For the time being, it is only the Swiss CAs that conducts [EUDIN] A2A information exchange with the Austrian CA

As a result, there are **two potential cases for** choosing the **bidirectional connection** type:

1. **Synchronisation of one's own waste shipment information:** For the waste shipment data that an EO submits to the Austrian CA, ensure that this data is and remains identical in the two EDI nodes, (a) in the EO's EDI node and (b) in the [EDM]. In particular: Ensure that any entering or editing of the EO's waste shipment data in the [EDM] online forms does not go unnoticed in the EO's EDI node
2. **Access to other parties' waste shipment information:** Pursuant to the [WSR], EOs share waste shipment information not only with CAs. Example: Pursuant to Art. 16(d) [WSR], treatment facilities disseminate confirmations of waste receipt not only with CAs, but also with the notifier. In this example, the bidirectional connection type enables Austrian notifiers to access confirmations of waste receipt that the Austrian CA receives from Switzerland via [EUDIN]. Note that the default for [WasteX] EDI and the [WasteX] protocol is for [WasteX] EDI nodes not to conduct forwarding. As a result, if the Austrian CA receives a [WasteX] CoWT transmission from a German treatment party, the [EDM] [WasteX] EDI node does not forward the message to the Austrian notifier via the [EDM WasteX] web service.

The bidirectional type of connection is advisable if any of the two cases, synchronisation or access, is relevant to the users of a software solution.

## 2.6 Web service operations

The web service features three operations – see the HTML Annex for more details:

### 1. ShareMessage

Serves two purposes:

- a. **Sharing waste shipment contents with the CA** by sharing a *user message* with the [EDM]
- b. Sharing a *signal message* with the [EDM], in reaction to receiving a user message from the [EDM]

Remarks:

- Note that clients use (b) only in bidirectional connections to the [EDM] (see 2.5). Clients that implement a unidirectional connection do not use (b)
- Note the **asynchronous** behaviour for (a):
  - When the client shares a *user message* with the [EDM], [EDM] reacts with sharing a *signal message* with the client (unless upon exceptions, which may cause the [EDM] to react with a fault)
  - The signal message response is **asynchronous** in the following sense:
    - The *ShareMessage* operation invocation output/response does not contain the signal message
    - The client needs to conduct polling, by using the *QueryUpdate* operation, for getting access to the [EDM]’s signal message response to the user message transmission. This allows for time decoupling. For example, it may take the [EDM] 7 minutes to create and disseminate a signal message in response to a user message M. If a client invokes *QueryUpdate* 5, 10, 15, 20, ... minutes after sharing M with the [EDM], it will be the client’s *QueryUpdate* invocation after 10 minutes that yields the signal message for the sharing of M, whereas the invocation after 5 minutes may yield an empty result

### 2. InitSync

Polling-related operation. Serves the purpose of retrieving a “starting point” for polling based on a timestamp

### 3. QueryUpdate

Polling operation. Called by the client in regular intervals, such as every 5 minutes, in order to receive “news/updates” from the [EDM]. There are two types of updates that [EDM] disseminates:

- a. Signal message  
Disseminated by the [EDM] in response to a client sharing a user message with the [EDM] via invoking the *ShareMessage* operation (see 1.a)
- b. User message  
Disseminated by the [EDM] in cases described under 2.5.2.  
Note: Clients that implement unidirectional usage (see 2.5) ignore any [EDM] dissemination of user messages, i.e., ignore any user message content in *QueryUpdate* operation output

## 2.7 Polling

### 2.7.1 Function of polling

User message and signal message transmission from [EDM] to the [EDM WasteX] client depend on polling, i.e., on the client's repeated invoking of the *QueryUpdate* operation. See section 3 for a definition of polling.

This function of polling depends on the type of connection to the [EDM WasteX] interface (see 2.5):

- In a **bidirectional connection**, the client is interested in both user messages and signal messages that the [EDM] disseminates, and uses polling to access these messages
- In a **unidirectional connection**, the client is only interested in the signal messages made available by the [EDM], i.e., in the [EDM]'s reaction to the client's transmission of a user message. Note that the [EDM WasteX] web service interface does not distinguish bidirectional and unidirectional connections. It is rather the client that makes the connection a unidirectional connection by ignoring all user messages that [EDM WasteX] disseminates to the client

### 2.7.2 Mode of polling

The default mode for polling is **continuous uninterrupted polling**. Example: The client invokes the *QueryUpdate* operation every 5 minutes, 24 hours a day, 7 days a week.

For bidirectional connections, continuous uninterrupted polling is a strict necessity, as polling interruptions would lead to delays in disseminating waste shipment payload.

For unidirectional connections, there is no strict functional necessity of continuous uninterrupted polling: For example, the client could interrupt polling whenever it has received all signal message responses to previous transmissions. However, the [EDM WasteX] authors strongly recommend continuous uninterrupted polling even for unidirectional connections:

- As explained in 2.7.1, the [EDM WasteX] web service does not distinguish between bidirectional and unidirectional connections. Polling interruptions can therefore lead to the "cluttered" (unevenly distributed, highly accumulated) dissemination of user messages, and therefore negatively impact EDI performance
- A client logic of continuous uninterrupted polling is more simple and robust than a client logic of "on demand polling"

See the constraints in 2.13 for the maximum polling frequency supported by [EDM WasteX].

### 2.7.3 Client side state management

The querying of updates from the [EDM WasteX] web service involves the management of state: "State" refers to distinguishing between the following:

1. Messages (user messages and signal messages) that the client already "knows", that the [EDM WasteX] has already previously disseminated to the client, and that [EDM WasteX] therefore shall not provide to the client in the *QueryUpdate* output
2. Messages that the client does not yet "know", and that [EDM WasteX] shall therefore include in the *QueryUpdate* output

The following applies to the state management:

- The [EDM WasteX] web service does not manage the state
- It is the client that manages the state
- It is the "UUID of the latest update available/known to the client" by which the client manages its state
- The client passes this UUID to the [EDM WasteX] webservice upon each *QueryUpdate* invocation
- Each time the client receives non-empty output from a *QueryUpdate* invocation, i.e., each time it receives one or more updates (*QueryUpdateResponse/Update*), it uses the *UpdateUUID* from the last *QueryUpdateResponse/Update* element as its new state, i.e., as its new "UUID of the latest update available/known to the client". Put briefly and simplified: In a series of *QueryUpdate* invocations, the client must use the output from one invocation as the input to the next invocation
- The previous point leaves one question open: Which "state UUID" shall the client use for the first in a series of *QueryUpdate* invocations, upon initializing or re-initializing the querying of updates? The *InitSync* operation serves this purpose. It provides the client with an "initial state UUID", based on a timestamp that the client passes to the web service. By using this UUID from the *InitSync* output in the first of a series of *QueryUpdate* invocations, the client will receive all updates available from the web service that are newer than the timestamp passed to *InitSync*
- Pursuant to the [WasteX] "Basic" specification [EDM WasteX] must keep updates for at least 4 weeks. The timestamp a client passes to *InitSync* shall lie anywhere between (*current time – 4 weeks*) and *current time*

## 2.8 Codelists

### 2.8.1 Introduction

For several data elements, the [WasteX] specification uses the codification of information and defines sets of permitted values. Examples:

- Values *ACCEPTED*, *REJECTED* in *WasteMovementSignalMessage/SignalID*
- Values *1*(for drum), *2*(for wooden barrel), ..., *9*(for other) in *MovementAnnouncement/PackagingTypeID*
- Values *040*(for Austria), *276*(for Germany), *380*(for Italy), ... in *NationalRoute/CountryID*

For several of these data elements, the [WasteX] specification refers to EUDIN codelists with their respective 4-digit identifier. Examples:

- Reference to codelist 3862 *countries* for data element *NationalRoute/CountryID*
- Reference to codelist 6524 *packaging types* for data element *MovementAnnouncement/PackagingTypeID*

### 2.8.2 Codelist illustration

The following table illustrates the first few entries of 6524 *packaging types*:

Code	Name
1	Drum
2	Wooden Barrel
3	Jerrican
...	...

### 2.8.3 Codelist characteristics

As 2.8.3 illustrates, the following are characteristics of a codelist:

- A table within which each row represents attributes of an element from a set
- One or more code/ID columns
- Zero or more columns with natural language texts (names, descriptions) or other types of information (such as population sizes or other numerical information)

## 2.8.4 Codelist uses in IT

The following two are main uses of codelists in IT:

1. **Lookup of characteristics:** Storage and exchange of information uses elements' codes/IDs only. Where software needs to access elements' attributes, such as name or description, it does so by looking the information up in the codelist (in: code; out: attribute)

One specific case of lookup: **Translation between codified information and human-readable natural language text:** Software solutions use lookup in codelists for a variety of purposes. One very frequently occurring purpose is the "translation" of codified information into human-readable natural language texts. For example, the use of codified information, such as *packaging type codes 1 to 9*, is common and advantageous in data electronic storage, access and exchange. The usability of user interfaces however typically benefits from using more than just codified information, especially human-readable natural-language texts, such as names and descriptions. Software solutions use codelists to "translate" between the codes/IDs used in internal representation and the natural language texts for user interfaces

2. **Defining sets of permitted values:** A codelist may define the set of elements from which one element must be selected in a given context

The use of codified information and codelists is also specifically important in EDI across natural language regions. For example, in a scenario in which the information sender speaks Italian and the information receiver speaks German, the use of codified information and codelists can help as follows:

- As the codified information is independent from natural languages, it is the same all across the world. There is **no need for "translations", for applying changes to the data, no matter where in the world the senders and receivers are located**
- The "translations" between codified information and human-readable natural language text contained in a variety of software solutions can have "natural language translations" as an intended side effect. For example, the Italian sender's software may use the following left-hand side packaging types codelist, whereas the German receiver's software may use the following right-hand side packaging types codelist, resulting in said translations:

Code	Name
1	Fusto
2	Barile di legno
3	Tanica
...	...

Code	Name
1	Trommel/Fass
2	Holzfass
3	Kanister
...	

## 2.8.5 Natural language content – supported languages

The EAA publishes all codelists referenced and used by the [WasteX] specification with natural language content (names, descriptions, etc.) in two languages, **English** and **German**.

The EAA is open to adding content in **additional languages on demand**.

## 2.8.6 EDM codelist web service

EDI nodes can access all codelists relevant for [WasteX] EDI from an [EDM] codelist web service. Software developers MAY choose to implement daily checks for codelist updates via this web service, for EDI nodes update local copies of codelists fully automated.

Software developers interested in using the [EDM] codelist webservice shall get in touch with the contact named in 1.7.

## 2.8.7 Further details

Note that corresponding codelists exist only for some of the enumerations statically defined by [WasteX], but not for all of these enumerations. For example, there is no corresponding codelist for the *ACCEPTED/REJECTED* enumeration from the *SignalID* element.

**The HTML Annex to this description lists all codelists that the [WasteX] specification uses, and contains links to the codelist contents.**

## 2.9 Party identification

### 2.9.1 [WasteX] party identification scheme

#### 2.9.1.1 Introduction

Some of the payload contains the **identification of parties**.

Example: The *MovementAnnouncement* – prior information regarding the actual start of a shipment, pursuant to Art. 16(b) [WSR] – contains, pursuant to the [WSR], information on the following parties:

- **Carriers** for the particular shipment
- **Waste producers** that are the origin of the particular shipment's waste

The **[WasteX]** specification has the following provisions on the identification of parties:

1. The messages contain a **single party identifier for each party**, but do not contain party name and party address, as the recipient/CA already “knows” the latter from notification/consent
2. A party's complete identification consists of **two parts**
  - a. **Party identifier**
  - b. **Register identifier**, identifying the registration process from which the party identifier results, i.e., identifying the type of party identifier

Example from *example\_A\_MovementAnnouncement.xml*:

```
<wxm:CarrierPartyID registerID="GLN.AT">9110016663449</wxm:CarrierPartyID>
```

In this example, the party identifier is *9110016663449*, and the register identifier is *GLN.AT*

3. The party identification, i.e, the use of party identifier and register identifier combinations, must follow a **scheme** defined by the [WasteX] specification

### 2.9.1.2 Characteristics

- The scheme’s **purpose** is the **uniform identification** of all parties across different countries and CAs, and the **use of existing registration processes**, instead of creating new registration requirements
- The scheme defines that the **country of the EO’s registered office determines the type of identifier** to use for that EO in [WasteX] EDI
- Two tables (lists) are the basis for the [WasteX] party identification scheme:
  1. A list of **registers** for party registration and public lookup of these registrations  
Example entries:

Type	Public Lookup	Notes / Example ID
<b>GLN.AT</b>	<a href="https://firmen.wko.at/">https://firmen.wko.at/</a> <a href="https://gepir.gs1.org/">https://gepir.gs1.org/</a>	<b>“Public administration GLN” of the Austrian public administration</b>  Example ID: 9110015414875
<b>UID.CH</b>	<a href="https://www.uid.admin.ch/Search.aspx">https://www.uid.admin.ch/Search.aspx</a>	<b>Swiss “Unternehmens-Identifikationsnummer”</b> (company identification number)  Example ID: CHE-105.874.886

2. A **“mapping” from countries to registers**, determining the register (type of identifier) to use in [WasteX] for EOs whose registered office is in that country  
Example entries:

Registered office country	EO ID register
<b>AT</b>	GLN.AT
<b>CH</b>	UID.CH
<b>GB</b>	VAT.GB
<b>LI</b>	FL.LI
<b>DE</b>	VAT.EU
<b>SK</b>	VAT.EU

### 2.9.1.3 On-demand extensibility

The following applies to the [WasteX] party (EO) identification scheme:

- **Initially** the scheme contains definitions of the type of EO identifiers to use **for just a few countries** of registered office
- The scheme is **open for adding additional definitions on demand**, depending on which countries of registered office actually occur for EOs involved in [WasteX] EDI relevant shipments

[WasteX] realizes the on-demand-extensibility as follows:

- Both of the following are **codelists that [WasteX] uses dynamically**:
  1. **List of registers for party registration – codelist 2237**
  2. **Mapping from countries to registers – codelist 1387**
- Adjusting [WasteX] EDI nodes to extensions of these codelists, and thus to extensions of the [WasteX] party identification scheme, is and must be fast – a matter of hours or at most days – and cost efficient

#### 2.9.1.4 Prerequisite: Use of [WasteX] party IDs in notification/consent

In most cases, there is the following **prerequisite** for using party IDs from the [WasteX] party identification scheme in [WasteX] EDI: Prior to [WasteX] EDI, the data transmission recipient must have already used and stored the respective party IDs in the notification/consent data stored at the receiving EDI node. See 2.13 for more details.

#### 2.9.2 [EDM] specific legacy party identification scheme

[EDM WasteX] supports a legacy identification scheme as an alternative to the [WasteX] party identification scheme introduced in 2.9.1:

- For all parties, the ID must be the Global Location Number (GLN) assigned to the party as primary identifier in the [EDM]
- For all parties, the register ID must be set to *GLN.AT*

For each [WasteX] user message, party IDs **MUST** be from one party identification scheme only. [EDM WasteX] does not support the use of party IDs from more than one identification scheme within a single user message.

The [EDM] specific legacy party identification scheme is available for EDI with the Austrian CA, but not for EDI with other CAs. The [EDM WasteX] authors recommend using party identifiers from the [WasteX] party identification scheme for interoperability across countries and CAs.

## 2.10 EDM WasteX EDI practicalities

### 2.10.1 Service level

The EAA operates [EDM WasteX] under the service level conditions of the [EDM]. See [edm.gv.at](http://edm.gv.at), "*Technische und organisatorische Spezifikationen*", "*Allgemeine Betriebsbedingungen*". For example, [EDM WasteX] can become temporarily unavailable due to scheduled maintenance, such as upgrades, on Tuesdays and Thursdays from 18:00 local time (CET/CEST).

## 2.10.2 Recommended reactions to certain [EDM WasteX] behaviour

Behaviour	Result	Description / Example / Handling
<b>Timeout / No Response / Network Errors</b>	<b>Failure</b> Regular	<p>This counts as an <b>outage</b> of the [EDM WasteX] WS, unless caused by limited connectivity at the WS client side.</p> <p>Can occur under regular circumstances, such as during deploy of a new [EDM WasteX] version.</p> <p><b>Reaction by software (sender side):</b> (see 2.10.5)</p> <ul style="list-style-type: none"> <li>• Automatic <b>transmission retries</b></li> <li>• If the WS is not back in an available state within a certain duration since first becoming unavailable, the WS client can automatically send <b>emails</b> to the Austrian CA</li> <li>• Display transmission status information to users (pending EDI transmissions, sent emails, etc.)</li> <li>• If this occurs in bulk transmissions: end/interrupt bulk transmission</li> </ul> <p><b>Reaction by users (sender side):</b></p> <ul style="list-style-type: none"> <li>• Usually no reaction required, other than checking back that EDI transmissions automatically resume after a while</li> </ul> <p><b>Reactions by IT support (sender side):</b> None required</p>
<b>SOAP Fault</b>	<b>Failure</b> Exception	<p>The WS reacts with a SOAP Fault message, instead of a SOAP response message</p> <p>Examples:</p> <ul style="list-style-type: none"> <li>• May occur in response to a malformed (non XML Schema compliant) request</li> <li>• May occur in response to failed WS client EDI node authentication</li> </ul> <p><b>Reaction by software (sender side):</b></p> <ul style="list-style-type: none"> <li>• Display information to users that a technical problem requires IT support</li> <li>• If this occurs in bulk transmissions: end/interrupt bulk transmission</li> </ul> <p><b>Reaction by users (sender side):</b></p> <ul style="list-style-type: none"> <li>• Pass issue to sender side IT support</li> </ul> <p><b>Reactions by IT support (sender side):</b></p> <ul style="list-style-type: none"> <li>• Resolve issue, for example with a hotfix</li> <li>• If needed, involve [EDM WasteX] IT support</li> </ul>

<p><b>Missing Signal</b></p>	<p><b>Unknown Exception</b></p>	<p>The WS reacts with a SOAP message (no SOAP Fault). The client does not receive a signal message for one or more transmission (for example, hours have passed since transmission, and there's still no signal message)</p> <p><b>Reaction by software (sender side):</b></p> <ul style="list-style-type: none"> <li>• Automatic transmission retries (see 2.10.2)</li> <li>• Display transmission status information to users</li> <li>• If signal remains missing after retries: Display information to users that a technical problem requires IT support</li> <li>• If this occurs in bulk transmissions: Does not affect bulk transmission – bulk transmission can continue</li> </ul> <p><b>Reaction by users (sender side):</b></p> <ul style="list-style-type: none"> <li>• If signal remains missing after retries: Pass issue to sender side IT support</li> </ul> <p><b>Reactions by IT support (sender side):</b></p> <ul style="list-style-type: none"> <li>• Involve [EDM WasteX] IT support, resolve issue</li> </ul>
<p><b>Rejection Signal</b></p>	<p><b>Failure Regular</b></p>	<p>The WS reacts with a signal that indicates automatic rejection</p> <p><b>Reaction by software (sender side):</b></p> <ul style="list-style-type: none"> <li>• Display transmission status information to users, including <i>SignalID</i>, <i>ReceiverFailureIndicator</i> and <i>SignalDescription</i>, plus formal validation report if available</li> <li>• Alert users that the transmission was unsuccessful and they <u>need</u> to take steps (interact with the software) to resolve the cause and <u>initiate a retransmission attempt</u></li> <li>• Bulk transmissions: Can continue</li> </ul> <p><b>Reaction by users (sender side):</b></p> <ul style="list-style-type: none"> <li>• Based on the information received in the signal, resolve the cause of transmission failure and initiate a retransmission attempt</li> <li>• If needed, <b>involve receiver side users</b>, for example, if notification/consent information is missing or wrong at the receiver side</li> </ul> <p><b>Reactions by IT support (sender side):</b> None required</p>

<p><b>Acceptance Signal with INFO or WARNING formal validation result</b></p>	<p><b>Success</b> Regular</p>	<p>The WS reacts with a signal that indicates acceptance and contains an INFO or WARNING overall result of formal validation</p> <p><b>Reaction by software (sender side):</b></p> <ul style="list-style-type: none"> <li>• Display transmission status information to users</li> <li>• Alert users that software has detected inconsistencies or violations of provisions in transmitted data, and that it is therefore <u>advisable</u> for users to review the data, and – if necessary – to update and re-transmit the data</li> <li>• Bulk transmissions: can continue</li> </ul> <p><b>Reaction by users (sender side):</b></p> <ul style="list-style-type: none"> <li>• Review the data transmission based on the information received in the signal, and, if needed, add/edit data and initiate a re-transmission</li> <li>• If needed, involve receiver side users. For example, in the review of the data transmission, it may turn out that wrong or missing data at the receiver side caused the INFO or WARNING, such as a typo in the consent information. In this case, first a user at the receiver side fixes the receiver side data, and then a user at the sender side initiates a re-transmission</li> </ul> <p><b>Reactions by IT support (sender side):</b></p> <ul style="list-style-type: none"> <li>• None required</li> </ul>
<p><b>Acceptance Signal with OK formal validation result</b></p>	<p><b>Success</b> Regular</p>	<p>The WS reacts with a signal that indicates acceptance and contains an OK result of formal validation</p> <p><b>Reaction by software (sender side):</b></p> <ul style="list-style-type: none"> <li>• Display transmission status information to users</li> <li>• Bulk transmissions: can continue</li> </ul> <p><b>Reaction by users (sender side):</b></p> <ul style="list-style-type: none"> <li>• None required</li> </ul> <p><b>Reactions by IT support (sender side):</b></p> <ul style="list-style-type: none"> <li>• None required</li> </ul>

### 2.10.3 Signal message sender and receiver side indication

Signal messages that signal rejection contain the *ReceiverFailureIndicator* indication. WS clients SHALL display this information to users as part of the signal content.

The meaning of the indicator is as follows:

- *true* means that resolving the (transmission) rejection is certain to require receiver side involvement, i.e., that it is certain the sender side cannot resolve the issue on its own
- *false* means the following:
  - The issue may be resolvable by the sender side on its own, i.e., without involving the receiver side
  - It is however not certain that the issue is resolvable without involving the receiver side

The purpose of the indicator, and the expected sender side reactions:

- *true* makes it clear to sender side users from the very beginning
  - that they **will need support from the receiver side for resolving the transmission issue**
  - that there is no point to attempting a fix of the transmission issue without involving the receiver side

Reaction expected from users at the sender side: Get in touch with the receiver side in an attempt to fix the cause for transmission rejection

- *false*, on the other hand, tells sender side users that they **may be able to resolve the transmission issue without involving the receiver side**

Reaction expected from users at the sender side:

- Try resolving the transmission issue without involving the receiver side
- Only if it turns out that the sender side cannot resolve the transmission issue on its own, the sender side gets in touch with the receiver side for resolving the issue

Examples:

**1. At the [EDM WasteX] WS an interaction with the database fails**

Handling by the [EDM WasteX] WS: *ReceiverFailureIndicator* set to *true*

Handling by sender side users: Get in touch with the receiver side for fixing the issue

**2. The transmitted user message contains a wrong notification number (e.g. typo)**

Handling by the [EDM WasteX] WS: This will trigger validation rule R780:

**R780**                      **ERROR**                      The movement announcement message refers to notification number »IT 347203«. A notification with that number does not exist in the database at »EDM.GV.AT«.

The WS sets *ReceiverFailureIndicator* to *false*

Handling by sender side users: Check the notification number. Upon detecting the typo, the sender side user corrects the notification number and triggers re-transmission

**3. The receiver has not yet stored information about a particular notification in its database**

Handling by the [EDM WasteX] WS: The same as in the previous point, point 2, i.e., triggering of validation rule R780

Handling by sender side users: Check the notification number. The sender side user finds that the transmitted notification number is correct. Consequently, the sender gets in touch with the receiver side, so that the receiver side can take any necessary actions, such as entering data on the particular notification. Then the sender triggers re-transmission

Examples 2 and 3 above illustrate cases in which software cannot determine whether resolving an issue is up to senders or to receivers. Especially when detecting inconsistencies between newly transmitted data and data already available at the receiver side, it may take senders, receivers or both to fix the issue.

#### 2.10.4 Sender and receiver side responsibilities

2.10.2 and 2.10.3 illustrate sender and receiver side responsibilities upon failed [EDM WasteX] transmissions:

1. It is a **sender side responsibility to initiate resolving failed transmissions**. The sender side shall not rely on the receiver side to initiate any fixing steps. Depending on the type of transmission issue, the sender side may however wait, e.g. for some hours, to see if the receiver side has fixed the issue by then (see third list item), prior to taking any steps for resolving the issue
2. The **sender side may need to involve the receiver side** in resolving failed transmissions, such as for entering or correcting data at the receiver side (user level), or for re-establishing regular operation of the web service (technical operation level)
3. Even though the sender side cannot and shall not rely on the receiver side to initiate actions for resolving transmission failures, the **receiver side does** initiate steps for resolving causes of certain transmission failures – possibly in parallel to the sender side. In particular, the EAA operates an automatic monitoring of the operation of [EDM WasteX], and takes steps for re-establishing regular operation when the monitoring indicates issues. This happens independent from the reporting of issues by the sender side

### 2.10.5 Handling of [EDM WasteX] outages

When a client EDI node attempts transmissions to [EDM WasteX], the web service may not be available (see first item in the table at 2.10.2). For example, a timeout may occur because of planned maintenance of the [EDM WasteX] web service.

Note that this section applies to neither of the following (see items other than the first item at 2.10.2):

- regular reactions to web service interactions (regular SOAP response messages)
- SOAP Fault reaction to a web service interaction
- receipt of signal messages indicating automated rejection

Recommended handling of [EDM WasteX] outages by WS client EDI nodes:

1. **Attempt automatic re-transmissions**, rather than requiring users to re-trigger transmissions manually, for transmissions that the client EDI cannot conduct due to an [EDM WasteX] outage. See 2.10.6.3 for more details
2. MAs, MA cancellations and CaCs may require transmission within a short period (a few hours, the same day) for meeting the legal requirements. In a scenario in which the [EDM WasteX] outage lasts longer than 60 minutes, EDI nodes can **email the information to [wastex@bmk.gv.at](mailto:wastex@bmk.gv.at)** automatically (without user interaction). The information content shall be contained in the email body as natural language plaintext. The email must not contain attachments. In particular, the EDI node must not send PDF forms by email to [wastex@bmk.gv.at](mailto:wastex@bmk.gv.at)
3. The above applies to MAs, MA cancellations and CaCs. EDI nodes must not send emails to [wastex@bmk.gv.at](mailto:wastex@bmk.gv.at) for any other message types and their contents (CoWR, CoWT, etc.)
4. The purpose of emails to [wastex@bmk.gv.at](mailto:wastex@bmk.gv.at) is transmitting in time. Emails to [wastex@bmk.gv.at](mailto:wastex@bmk.gv.at) do **not replace** electronic structured data transmissions, such as those over [EDM WasteX]. Instead, when the [EDM WasteX] service returns into an operational state, the client EDI node shall conduct all previously delayed transmissions via [EDM WasteX] (as described in the first point), independent of whether or not it has sent emails to [wastex@bmk.gv.at](mailto:wastex@bmk.gv.at) for the respective transmissions
5. Note that the timing-related [WasteX] **formal validation rules offer “timing tolerance”** of a few days in order to support recovery of EDI transmission after transmission failures. For example, the formal validation rules accept MA transmissions even after the start date. This means that even with outages that last several days, the sender side can comply with all requirements by doing the following:
  - a. Send emails while the outage lasts (see points 2, 3 and 4) – comply with timing requirements
  - b. Subsequent EDI transmission once the outage ends – comply with structured data transmission requirements

## 2.10.6 Transmissions retries

### 2.10.6.1 Introduction

Transmission attempts to [EDM WasteX] can fail for a variety of reasons. See 2.10.2 for a classification of possible reasons.

The following applies to failed transmissions and attempting re-transmission:

- If the transmission fails, then the **recipient (Austrian CA) does not get to know about the failed transmission attempt and its contents**
- In order for the sending EO to provide the information from the failed transmission to the Austrian CA, the **EO will have to re-conduct (retry) the transmission**
- Typically, EOs will conduct transmission **retries** over the same **information exchange channel** as the initial failed transmission. In particular, if EDI transmissions to [EDM] via [EDM WasteX] fail, then the EOs will typically conduct the retries again as [EDM WasteX] transmissions. The EO can however switch to another channel of structured electronic data exchange if the EO wishes so. In particular, an EO can decide to enter data from a failed transmission via the [EDM] web forms
- As 2.10.2 illustrates, **in most cases sender side users will have to take certain actions before attempting re-transmission**. For example, sender side user may correct data, or get in touch with sender side IT support or the receiver side in order to eliminate the cause of transmission failure. In such cases, there is no point to attempting re-transmission as long as the EO has not yet conducted said actions for eliminating the cause of transmission failure
- The main exception to the previous point is **[EDM WasteX] outages** (see 2.10.6 and the first item in the table of 2.10.2). In this case, it makes sense for the sender to retry transmission at a later point in time, without taking any other actions targeted at fixing some transmission error before retrying transmission. See 2.10.5 for more on the handling of [EDM WasteX] outages

### 2.10.6.2 Recommended EDI node functionality

EDI nodes, in particular clients to the [EDM WasteX] WS, can support a smooth data interchange by featuring the following functionality:

- The EDI node **keeps track of transmission attempts and transmission status**
- **Users can access lists/pools of "required/recommended actions"**, including
  - Lists/pools of attempted transmissions that failed and have not yet been followed by successful re-transmissions (typically require action, in some cases urgent action, in particular for announcements close to the announcement deadline)
  - Lists/pools of attempted transmissions that succeeded but yielded INFO or WARNING in formal validation (see 2.10.7, recommended for inspection/verification by sender side users)
- **Possibility for users to trigger re-transmissions**
  - Explicitly and/or implicitly (for example, completing the correction of a data instance may automatically trigger a re-transmission)
  - Possibility for certain users to trigger **bulk re-transmissions**
    - Example 1: Possibility to trigger re-transmission for all failed transmissions for a certain notification. For illustration, see example 4 in 2.10.3: After the Austrian CA has entered a previously missing notification/consent into its database, the sender may want to "bulk re-transmit" all previously failed transmissions belonging to that notification
    - Example 2: After the receiver side has fixed an outage of [EDM WasteX], possibility to trigger re-transmission for all transmissions that previously failed because of the outage
- **Automatic re-transmission attempts** by the EDI node for transmissions that fail due to an [EDM WasteX] outage. See 2.10.6.3 for more details
- **Automatic e-mails to the CA** during an [EDM WasteX] outage for meeting transmission deadlines. See 2.10.5 for more details

### 2.10.6.3 Automatic retries

For transmissions that fail because of an [EDM WasteX] outage, WS client EDI nodes can implement automatic transmission retries.

As 2.10.2 illustrates, EDI nodes must not automatically retry transmissions that fail for reasons other than [EDM WasteX] outage.

The recommended schedule for an EDI node to re-attempt transmission is as follows:

- 20 minutes after the initial transmission attempt
- 40 minutes after the initial transmission attempt
- 60 minutes after the initial transmission attempt
- after that, once per day, for instance late in the day according to local time

Note that in 2.10.6.2 there is also the recommendation to enable users to trigger "bulk re-transmissions".

## 2.10.7 Transmission triggers

Trigger	Description
TRG_AUTO_DATA	<p>This refers to an EDI node <u>automatically</u> triggering a message transmission as soon as new or updated data of sufficient maturity/finality becomes available at that EDI node, such as resulting from a user entering or editing data in a GUI.</p> <p>This does <u>not</u> refer to a scenario in which a user explicitly triggers the EDI transmission of data, such as by pressing a “send” button.</p>
TRG_AUTO_TIMER	<p>This refers to an EDI node <u>automatically</u> triggering a message transmission as soon as the EDI node detects reaching or exceeding a certain point in time.</p>
TRG_MANUAL	<p>This refers to a software user <u>explicitly</u> triggering the transmission of a message by conducting a user interaction, such as pushing a “send” button.</p> <p>This does <u>not</u> refer to a scenario in which a user interaction, such as the entering (completion/saving) of data, implicitly triggers an EDI transmission.</p>

Note that pursuant to 2.10.6.2 EDI nodes shall support TRG\_MANUAL for all types of user messages as a means for (certain) users to handle transmission issues.

For announcements, in particular MAs, EDI nodes can support TRG\_AUTO\_DATA, TRG\_AUTO\_TIMER or both (in addition to TRG\_MANUAL, as explained above). When supporting TRG\_AUTO\_DATA and TRG\_AUTO\_TIMER, EDI nodes may leave it to users to select the type of trigger, such as through a per-notification/consent configuration.

TRG\_AUTO\_TIMER for announcements refers to an automatic triggering of transmissions when reaching a certain point in time, in particular when getting close to the transmission deadline. EDI nodes will typically need to support users with data completion in time, such as through reminders. With regard to transmission triggering, it is important to take into account that the following scenario would result in a violation of the transmission deadline:

- The EDI node conducts a transmission attempt at the day of the deadline, but the transmission fails
- Only the re-transmission attempt at the following day turns out successful

For reports, such as CoWR and CoWT, EDI nodes shall support TRG\_AUTO\_DATA but not TRG\_AUTO\_TIMER, as there is no point in delaying the transmission when the complete set of data is already available.

### 2.10.8 Bulk transmissions

A client to the [EDM WasteX] WS may have multiple messages “waiting” for transmission. This specification calls an EDI node’s operation on a queue/backlog of messages waiting for transmission **bulk transmission**. I.e., the term refers to the (intended) transmission of a potentially larger number of messages within a short period of time.

The **[WasteX] and [EDM WasteX]** specification design focusses on real time data interchange, i.e., the exchange of data at when it is “finalised” and ready for transmission, independent of when that is (in the morning, in the night, etc.). This means that the **specifications do not focus on bulk transmissions**. As a result, the specification authors recommend keeping bulk transmissions to a minimum, and only using bulk transmissions when necessary.

The following two are scenarios in which in which bulk transmissions can become necessary in [WasteX] data interchange:

1. Late transmissions for “catching up” after an interruption of the data exchange, such as after an [EDM WasteX] outage (see 2.10.5)
2. Use of TRG\_AUTO\_TIMER for announcement transmissions, (see 2.10.7), such as transmissions of announcements once per day, immediately before the transmission deadline

For the implementation of bulk transmissions, it is important to take into account the **“number of transactions per unit of time” limits** described in 2.14. Depending on the number of messages to transmit, it may become necessary to spread the transmissions out to multiple hours of the day (transmission of  $n$  messages per hour, for each hour until all messages waiting for transmission have been sent), or even to multiple days.

### 2.10.9 Updates / corrections

An EDI node can update/correct contents from a previously transmitted user message by re-transmitting that user message with updated/corrected contents. [EDM WasteX] treats a received transmission as an update if its database already contains entries for the respective IDs, in particular for the respective combination of notification number and sequential number.

Note that there can be limits to the update ability. For example, validation rule R100 rejects transmissions, including updates, for cancelled shipments.

### 2.10.10 Formal validation

The Schematron-based set of formal validation rules is an integral part of the [WasteX] specification. The violation of ERROR-rules leads to the automatic rejection of transmissions, and therefore typically to a necessity of urgent reactions by the EO.

Signal messages contain the results of formal validation in human-readable text, for the time being in English language only.

For the implementation of an EDI connection to [EDM WasteX] it is crucial to make the human-readable validation results accessible to software users, so that users can react, such as by correcting data and subsequently triggering an EDI (re-)transmission. For each entry of the formal validation protocol, software users shall have access to all the following via their software's GUI:

1. Natural language validation result text
2. Validation rule ID
3. INFO/WARNING/ERROR distinction

## 2.11 Preparatory step 1: Client EDI node registration

A client must use EDI node identifier (*Client-ID*) and EDI node secret (*Client-Secret*) with each [EDM WasteX] web service interaction. Section 2.14 on authentication describes the ways in which an EDI node can use identifier and secret with web service interactions.

Prior to establishing a connection to [EDM WasteX], operators or administrative users of the EDI node that becomes a client to [EDM WasteX] must "register" the EDI node with the [EDM], i.e., they must request EDI node identifier and EDI node secret from [EDM] administrators.

Please refer to the contact in 1.7 for requesting such credentials.

Note that there are different EDI node credentials for the different environments (see 2.3).

[EDM] administrators hand out credentials for the production environment only after successful tests in a test (pre-production) environment.

## 2.12 Preparatory step 2: Enabling access by/for parties

Conducting EDI over the [EDM WasteX] web service requires a second preparatory step. EOs that want to use EDI transmissions to the Austrian CA need to “activate” waste shipment EDI transmissions for a certain (registered) EDI node. Such activation is the prerequisite for the EDI node to transmit and access waste shipment information “on behalf” of the respective EO:

- In *ShareMessage* interactions, there is an identification of EDI node and message creation party. [EDM WasteX] accepts the interaction only if the EO identified as message creation party has “activated” waste shipment EDI
- In *QueryUpdate* interactions, [EDM WasteX] selects updates for an EDI node based on contents (such as notifier or treatment party from the notification/consent) and “activation” of EDI nodes. See 2.5.2 for an explanation on the contents the [EDM] makes accessible as user messages via [EDM WasteX]. In its current version, [EDM WasteX] limits such access to notifiers, consignees and waste treatment parties, excluding subsequent treatment parties

For the time being, EOs shall get in touch with the contact in 1.7 for such activation.

The BMK intends to offer self-administration of activation/de-activation in the EDM user interface in the future.

EOs interested in using [EDM] side administration for restricting an EDI node’s [EDM WasteX] access to specific notifications shall get in touch with the contact in 1.7. Note that the authors of [EDM WasteX] advise against such [EDM] side notification-level administration due to the high efforts for EOs.

## 2.13 Preparatory step 3: Party IDs in the notification

Compare 2.9.1.1:

- [WasteX] messages contain party identifiers, but do not contain information such as party name and party address
- [WasteX] relies on recipients (and recipient EDI nodes) already “knowing” information such as party name and party address

For EDI transmissions to the Austrian CA via [EDM WasteX], the ability to use EO identifiers from the [WasteX] party identification scheme requires the following preparatory step:

**The notifier must provide party identifiers from the [WasteX] party identification scheme for all EOs named in the notification (notifier, consignee, waste producers, waste treatment parties, carriers) within the submission of the notification to CAs.**

As an exceptional workflow, the CA may also accept the submission of these identifiers from a party other than the notifier (in particular consignee or treatment party) and separate from the notification submission.

## 2.14 Authentication

### 2.14.1 Introduction

The [EDM WasteX] web service supports two types of authentication, which the subsequent sections describe in more detail:

1. HTTP Basic Authentication
2. HMAC Authentication

Upon implementing a connection with the [EDM WasteX] web service, software developers and designers can freely choose between these two options.

The operators of the [EDM WasteX] web service RECOMMEND using the HMAC authentication option, as security tests show that the HMAC variant provides a significantly higher level of security.

### 2.14.2 HTTP Basic Authentication

#### 2.14.2.1 Description

With this authentication option, the web service client uses the credentials from registration, *Client-ID* and *Client-Secret*, as follows:

- For each interaction with the [EDM WasteX] web service, the client transmits an [HTTP] *Authorization* Header. Note that the following is the generic pattern for [HTTP] *Authorization* Headers:  

```
Authorization: <auth-scheme> <authorization-parameters>
```
- The client uses value »Basic« for *auth-scheme*. I.e., for each interaction with the web service, the client transmits a header with this pattern:  

```
Authorization: Basic <credentials>
```
- The client constructs the *credentials* value as follows:
  1. Concatenate the following 3 character strings:
    - a. *Client-ID*
    - b. The character string consisting of a single character, »:« (colon, 0x3A)
    - c. *Client-Secret*
  2. Calculate the byte sequence resulting from applying [UTF-8] character encoding to the character sequence of step 1
  3. Apply [Base64] encoding to the byte sequence from step 2. Use the following variant of [Base64] encoding:
    - Variant with characters »+« and »/« for index values 62 and 63
    - Variant with *padding*. This variant ensures that the length of the resulting character sequence is a multiple of 4. If needed, the encoding appends one or two »=« padding characters (0x3D) to the result for complying with this length constraint

### 2.14.2.2 Example

This example assumes that the registration of the web service client provided the following client credentials:

```
Client-ID: WASTE_IT.CLOUD
Client-Secret: 6H!uU3w*hSyuzj
```

[Base64] encoding of the [UTF-8] byte sequence for »WASTE\_IT.CLOUD:6H!uU3w\*hSyuzj« yields:

```
»V0FTVEVfSVQuQ0xPVUQ6Nkg hdVUzdyp oU3l1emo=«
```

In this example, the web service client uses the following [HTTP] Header in each of the interactions with the web service:

```
Authorization: Basic V0FTVEVfSVQuQ0xPVUQ6Nkg hdVUzdyp oU3l1emo=
```

### 2.14.3 HMAC Authentication

#### 2.14.3.1 Description

With this authentication method, the web service client uses the credentials from registration, *Client-ID* and *Client-Secret*, as follows:

- The web service client transmits an [HTTP] *Authorization* Header with each web service interaction. Note that the following is the generic pattern for [HTTP] *Authorization* Headers:

```
Authorization: <auth-scheme> <authorization-parameters>
```

- The web service client uses a custom *Authorization* scheme with »EDM1« *auth-scheme* value and the following pattern:

```
Authorization: EDM1 clientID="<client-id>", clientHMAC="<hmac>"
```

- The client uses the *Client-ID* character string from client registration is as *client-id* value

- The web service client calculates the *hmac* value as follows:
  1. Compose the `stringToSign` character string as follows:

For the *ShareMessage* operation with user messages:

```
stringToSign = transactionUUID + "\n" +
              messageUUID + "\n" +
              messageCreationPartyID
```

For the *ShareMessage* operation with signal messages:

```
stringToSign = transactionUUID + "\n" + "\n"
```

For the *InitSync* operation:

```
stringToSign = startFromUTCTimestamp
```

For the *QueryUpdate* operation:

```
stringToSign = updateRangeStartUUID
```

The following applies to composing the string:

- The web service client must use/insert the character strings exactly as they appear in the XML input to the operation. For instance, for [UUID] values the client must use the canonical representation from the XML input
- The web service client must use the XML element content only, and ignore any XML attributes. In particular, the web service client must ignore the *registerID* attribute for *MessageCreationPartyID* content
- The above character strings represent the following operation input values:

```
transactionUUID
```

Content of *ShareMessageRequest/TransactionUUID*

```
messageUUID
```

Content of *ShareMessageRequest/UserMessage/*  
*WasteMovementUserMessage/MessageUUID*

```
messageCreationPartyID
```

Content of *ShareMessageRequest/UserMessage/*  
*WasteMovementUserMessage/MessageCreationPartyID*

```
startFromUTCTimestamp
```

Content of *InitSyncRequest/StartFromUTCTimestamp* without  
time-zone indication

Example: If the content of *StartFromUTCTimestamp* is

```
2021-04-19T14:35:43.699370Z, then startFromUTCTimestamp is
2021-04-19T14:35:43.699370
```

```
updateRangeStartUUID
```

Content of *QueryUpdateRequest/UpdateRangeStartUUID*

2. Calculate an HMAC-SHA-256 value (byte sequence) from the following two inputs:
  - a. *data*: byte sequence resulting from [UTF-8] encoding of the `stringToSign` character sequence
  - b. *key*: byte sequence resulting from the [UTF-8] encoding of the `Client-Secret` character sequence
3. Apply [Base64] encoding to the byte sequence from point 2, using the following variant of [Base64] encoding:
  - Variant with characters »+« and »/« for index values 62 and 63
  - Variant with *padding*. This variant ensures that the length of the resulting character sequence is a multiple of 4. If needed, the encoding appends one or two »=« padding characters (0x3D) to the result for complying with this length constraint

### 2.14.3.2 Example

For this example, the client credentials are the same as in the previous HTTP Basic authentication example:

```
Client-ID: WASTE_IT.CLOUD
Client-Secret: 6H!uU3w*hSyuzj
```

This example illustrates the HMAC authentication for a `ShareMessage` operation with the following input (excerpt):

```
<soap12:Envelope>
  <soap12:Body>
    <wxt:ShareMessageRequest>
      <wxt:ClientInterfaceVersionID>1.08</wxt:ClientInterfaceVersionID>
      <wxt:ClientVersionID>3.3.2-1</wxt:ClientVersionID>
      <wxt:TransactionUUID>825afc6d-2d22-4f78-a0bf-fef1dd45d5eb</wxt:TransactionUUID>
      <wxt:UserMessage>
        <wxm:WasteMovementUserMessage>
          <wxm:MessageUUID>2ddea4a4-2ede-4734-967d-5d9ae904a02e</wxm:MessageUUID>
          <wxm:MessageCreationUTCTimestamp>2023-03-10T08:34:25.738661Z</wxm:MessageCreationUTCTimestamp>
          <wxm:MessageCreationDataFormatVersionID>1.08</wxm:MessageCreationDataFormatVersionID>
          <wxm:MessageCreationPartyID registerID="VAT.EU">IT02795790241</wxm:MessageCreationPartyID>
          <wxm:MovementAnnouncement>
```

For the `ShareMessage` operation with this input the `stringToSign` is:

```
"825afc6d-2d22-4f78-a0bf-fef1dd45d5eb" + "\n" +
"2ddea4a4-2ede-4734-967d-5d9ae904a02e" + "\n" +
"IT02795790241"
```

Applying HMAC-SHA256 to the [UTF-8] encodings of `stringToSign` (*data*) and `Client-Secret` »6H!uU3w\*hSyuzj« (*key*) results in the following byte sequence (represented here in hexadecimal notation):

```
»faea27f8df59bd2fe1cf971f9b7d0d30d017438eb11239e5ef73821ce5af754d«
```

Applying [Base64] encoding to this value yields:

```
»+uon+N9ZvS/hz5cfm30NMNAXQ46xEjn173OCHOWvdU0=«
```

Thus, the web service client invokes the `ShareMessage` operation for the above-mentioned `ShareMessage` input with the following [HTTP] Header:

```
Authorization: EDM1 clientID="WASTE_IT.CLOUD",
clientHMAC="+uon+N9ZvS/hz5cfm30NMNAXQ46xEjn173OCHOWvdU0="
```

## 2.15 EDM WasteX interface constraints

The maximum polling frequency supported by [EDM WasteX] is every 2 minutes. The recommended polling frequency is between 2 and 30 minutes. This applies to the use of the *QueryUpdate* operation.

For *ShareMessage* and *InitSync*, the maximum supported number of invocations per client and duration is as follows:

	per hour	per day
ShareMessage	5.000	20.000
InitSync	10	10

## 2.16 EDM WasteX v1.08 limitations

### 2.16.1 Introduction

There is a limitation of the [EDM WasteX] v1.08 interface compared to the interface specification.

This section details the limitation.

### 2.16.2 Unprocessed payload

There are contents that [EDM WasteX] v1.08 cannot process. The behaviour of [EDM WasteX] with respect to such contents is as follows:

- [EDM WasteX] accepts user message transmissions with these contents. Transmitting such contents to [EDM WasteX] does not trigger automatic rejections
- [EDM WasteX] does however automatically issue INFO entries in the signal message's formal validation report when it encounters such contents, in order to raise the sending EO's awareness that the Austrian CA does not currently have access to these contents via the [EDM] user interface
- Apart from the INFO reaction described in the previous point, [EDM WasteX] "ignores" these contents for the time being, in the sense of skipping structured storage in [EDM]'s database. As a result, [EDM] users cannot access these contents via the [EDM] user interface

[EDM WasteX] v1.08 ignores the following contents, in the sense described above:

- **Carrier confirmations** (*CarrierConfirmation* user message element): [EDM WasteX] ignores messages of type *carrier confirmation* as a whole

### 3 Terms and definitions

allow list	A mechanism which explicitly allows some identified entities to access a particular privilege that is denied by default
asynchronous web service operation	A web service operation designed so that a client has to wait for results, and distinguishing (a) initiating processing and (b) delivery of ultimate results; See also ↗ synchronous web service operation
business process	A set of activities a party (EO or CA) can execute to achieve a desired result in pursuit of a specified objective; Source: adapted from ISO 15704:2019(en)
carrier	A party that physically transports goods from one place to another; Source: [GSI XML] glossary; See also: ↗ economic operator
competent authority (CA)	Authority/authorities designated by a country to be responsible for receiving ↗ waste shipment ↗ notifications and any information related to it, and for responding to such a notification; Source: [BC] and [WSR]
consignee	The EO under the jurisdiction of the country of destination to whom waste is shipped for ↗ recovery or ↗ disposal; Source: adapted from [WSR]
database	Collection of machine-readable information organized so that it can be easily accessed, managed and updated; source: ISO 5127:2017(en) Information and documentation – Foundation and vocabulary
economic operator (EO)	Legal or natural person active on the market, i.e., offering the execution of work, the supply of products or the provision of services on the market; Source: EU Directive 2014/24 on public procurement  Note: Used for all parties involved in waste shipments except CAs, for example ↗ notifier, ↗ carrier, ↗ consignee, etc.

EDI node	<p>A sending and/or receiving point in ↗ electronic data interchange (EDI).</p> <p>Note 1: An EDI node is associated with:</p> <ul style="list-style-type: none"> <li>• one ↗ database that is the source of sent ↗ messages and/or the destination of received ↗ messages</li> <li>• one software product/service as marketed to customers/users</li> <li>• EDI functionality, in particular the capability of <ul style="list-style-type: none"> <li>○ reading from the ↗ database and creating/sending ↗ messages, and/or</li> <li>○ receiving/parsing/validating ↗ messages and writing the received contents to the ↗ database associated with the EDI node</li> </ul> </li> </ul> <p>Note 2: The software product/service with which an EDI node is associated can be the same for different EDI nodes</p> <p>Fictitious example: Waste management software “AlphaWaste” is offered in two variants:</p> <ol style="list-style-type: none"> <li>1. Cloud – software/database hosted by a third party, such as the “AlphaWaste” software/service provider, and used by the customer over the Internet</li> <li>2. On-premises – software/database hosted/operated on customer’s IT infrastructure</li> </ol> <p>“AlphaWaste” customer A may use the cloud variant, and customer B the on-premises variant. EDI can take place between these two “AlphaWaste” EDI nodes</p>
electronic data interchange (EDI)	<p>Structured way of transmitting data held electronically from ↗ database to ↗ database, usually using telecommunications networks; Source: ISO/IEC/IEEE 24765:2017(en) Systems and software engineering – Vocabulary; See also: ↗ message, ↗ EDI node</p>
facility	<p>Note: “facility” is a term that is widely used in the [BC] and the [WSR]. For reasons of harmonization and unambiguity, the [WasteX] specification use the terms “waste treatment party” and “waste treatment site” instead; See also: ↗ treatment, ↗ site, ↗ economic operator</p>
interim treatment	<p>An operation such as accumulation, temporary storage or repackaging applied prior to submission to a subsequent ↗ treatment of waste; Source: adapted from [WSR]</p>
message	<p>Collection of information sent through an information channel as a single logical entity; Source: adapted from ISO/IEC/IEEE 21451-7:2011(en) and ISO/TS 18234-6:2006(en)</p> <p>See also: ↗ electronic data interchange</p>
message format	<p>↗ data format for certain types of ↗ messages</p>

polling	A client-server interaction pattern whereby a client <u>periodically</u> sends a request to a server in order to get to know whether or not something new and relevant to the client has become available
shipment of waste	Transport of waste destined for recovery or disposal in another country; Source: [WSR]
signal message	<p>A ↗ message with a supporting role in establishing message exchange patterns, non-repudiation and reliability; source: [AS4]</p> <p>See also: ↗ user message, ↗ electronic data interchange</p> <p>Note: A WasteX signal message carries the following information:</p> <ul style="list-style-type: none"> <li>• information of whether or not a received ↗ user message was accepted or rejected at the receiving ↗ EDI node</li> <li>• report resulting from the formal validation of the received ↗ user message at the receiving ↗ EDI node</li> </ul>
site	A location at which a waste ↗ treatment operation is carried out, or at which waste ↗ production occurs
synchronous web service operation	A web service operation designed to deliver results to the client immediately, <u>without</u> distinguishing (a) initiating processing and (b) delivery of ultimate results; See also ↗ asynchronous web service operation
treatment (of waste)	Recovery or disposal operations, including preparation prior to recovery or disposal; Source: EU Waste Directive 2008/98
user	A (natural) person who interacts with a system, product or service; Source: ISO 26800:2011 Ergonomics
user message	<p>A ↗ message that contains business payload to be exchanged between two ↗ EDI nodes; Source: adapted from [AS4]</p> <p>See also: ↗ signal message</p> <p>Note: In the context of the WasteX specification business payload is waste shipment related information, such as the mass of waste</p>
waste shipment	See ↗ shipment of waste
[XSLT] processor	<p>A software (module/library) capable of applying [XSLT] to [XML] instances. An [XSLT] processor takes one or more [XML] instances plus one or more [XSLT] stylesheets as input and processes them to generate an output</p> <p>Example: [Saxon]</p>

## 4 Acronyms

API	application programming interface
AT	Austria
B2A	business-to-administration business processes and data interchanges involving both EOs and CAs
B2B	business-to-business business processes and data interchanges involving EOs only, i.e., not involving CAs
BAFU	Swiss Federal Office for the Environment (Bundesamt für Umwelt), the main CA for waste shipments in Switzerland ( <a href="https://www.bafu.admin.ch/">https://www.bafu.admin.ch/</a> )
BMK	Austrian Federal Ministry for Climate Action, Environment, Energy, Mobility, Innovation and Technology, the CA for waste shipments in Austria ( <a href="https://www.bmk.gv.at/en.html">https://www.bmk.gv.at/en.html</a> )
BP_AMBER_CONSENT	The business process involving notification to CAs and the intended result of which is consent by CAs to intended shipments of waste  Note 1: The word “amber” is from the domain terminology and refers to shipments that require prior written notification and consent  Note 2: In the domain also referred to as the “notification process”
BP_AMBER_INFO	The business process of EOs sharing information with CAs and other EOs on individual shipments of waste under a consent, i.e., shipments of waste requiring prior notification and consent  Note 1: Includes announcing a shipment of waste, confirming the receipt of waste, and confirming the completion of treatment of shipped waste  Note 2: The word “amber” is from the domain terminology and refers to shipments that require prior written notification and consent  Note 3: In the domain also referred to as the “movement document process”
BP_GREEN_INFO	The business process of EOs preparing information on an individual shipment of waste <u>not</u> requiring prior notification and consent, and of providing this information to CAs upon request  Note 1: The word “green” is from the domain terminology and refers to shipments that do <u>not</u> require prior written notification and consent  Note 2: In the domain also referred to as the “consignment information” process

CA	competent authority (for shipments of waste, if not otherwise specified)  Note: In a few sections in this specification, CA stands for Canada. This will be clear from the context
CaC	carrier's confirmation of taking charge of a shipment of waste, in accordance with Article 16 of the WSR, as well as Annex IB block 8 and Annex IC paragraphs 6 and 32; see also MA, CoWR and CoWT
CoWR	confirmation of waste receipt; one of the per-shipment B2A information exchanges mandated by [BC] and [WSR]; see also MA, CaC and CoWT
CoWT	confirmation of the completion of the treatment of shipped waste; one of the per-shipment B2A information exchanges mandated by [BC] and [WSR]; see also MA and CoWR
EAA	Environment Agency Austria (Umweltbundesamt, <a href="https://www.umweltbundesamt.at/en/">https://www.umweltbundesamt.at/en/</a> )
EC	European Commission ( <a href="https://ec.europa.eu/">https://ec.europa.eu/</a> )
EC DG ENV	European Commission, Directorate-General for Environment ( <a href="https://ec.europa.eu/info/departments/environment_en">https://ec.europa.eu/info/departments/environment_en</a> )
EDI	electronic data interchange
EO	economic operator
EU	European Union
GLN	Global Location Number, an identification scheme by GS1
GS1	A neutral, global collaboration platform that brings industry leaders, government, regulators, academia, and associations together to develop standards-based solutions to address the challenges of data exchange ( <a href="https://www.gs1.org/">https://www.gs1.org/</a> )
GUI	graphical user interface
IEC	International Electrotechnical Commission, an international standards organization that prepares and publishes international standards for all electrical, electronic and related technologies ( <a href="https://www.iec.ch/">https://www.iec.ch/</a> )
IETF	Internet Engineering Task Force, a large open international community of network designers, operators, vendors, and researchers concerned with the evolution of the Internet architecture and the smooth operation of the Internet, which develops and promotes voluntary Internet standards ( <a href="https://www.ietf.org/">https://www.ietf.org/</a> )
ISO	International Organization for Standardization, an international standard-setting body composed of representatives from various national standards organizations ( <a href="https://www.iso.org/">https://www.iso.org/</a> )

MA	movement announcement; one of the per-shipment B2A information exchanges mandated by [BC] and [WSR]; see also CoWR and CoWT
MS	Member state of the European Union
OASIS	Organization for the Advancement of Structured Information Standards, a non-profit international standards body offering projects – including open source projects – a path to standardization and de jure approval for reference in international policy and procurement ( <a href="https://www.oasis-open.org/">https://www.oasis-open.org/</a> )
RDBMS	Relational Database Management System
RFC	Request For Comments, publication by the IETF or associated bodies. Some RFCs evolve into Internet Standards
SVRL	[Schematron] Validation Report Language, a data format used for expressing the results of [Schematron] validation
UN/CEFACT	United Nations Centre for Trade Facilitation and Electronic Business, an organisation open to experts nominated by UN Member States and UN recognized organisations, and working on harmonizing, standardizing and automating the exchange of information ( <a href="https://unece.org/trade/uncefact">https://unece.org/trade/uncefact</a> )
W3C	World Wide Web Consortium, an international community that develops open standards to ensure the long-term growth of the web ( <a href="https://www.w3.org/">https://www.w3.org/</a> )
WS	web service

## 5 References

- [[AWG 2021](#)] 2021 Circular Economy amendment to the Austrian Federal Act on sustainable waste management (Abfallwirtschaftsgesetz). Contains an obligation for EOs to exchange waste shipment data as structured electronic data with the Austrian CA
- [[BC](#)] Basel Convention on the control of transboundary movements of hazardous wastes and their disposal
- [[EDM](#)] “Electronic Data Management”  
Interconnected system of IT applications and databases by the Austrian public administration to support environment-related business processes, such as reporting under reporting obligations. Part of the Austrian e-Government services offered under <https://www.usp.gv.at/en/index.html>.  
The Austrian CA for waste shipments uses [EDM] for the management of waste shipment processes and data
- [[EUDIN](#)] “European Data Interchange for Waste Notification Systems”  
Refers to all of the following:
1. A multi-national initiative, established in 2005, on waste shipment EDI
  2. The set of EDI specifications published by the EUDIN initiative
  3. An IT infrastructure for supporting waste shipment EDI, in particular the EUDIN Message Broker (illustrated in on page 5)
- [[IMSOC](#)] Integrated Management System for Official Controls  
This refers to:
1. A concept to allow EU/EC systems and MS systems to exchange information and share features, as laid down in the IMSOC Commission Implementing Regulation (EU) 2019/1715
  2. IT components and systems integrated following the IMSOC concept and IMSOC regulation
- [[RFC2119](#)] Key words to indicate requirement levels, IETF Best Current Practices
- [[Schematron](#)] 2016 Information technology, Schema Definition Languages (DSDL), Part 3: Rule-based validation, ISO/IEC 19757-3 standard
- [[VeVA-Online](#)] e-Government solution used by the Swiss CAs for managing waste shipment processes and data
- [[WasteX](#)] A modular set of waste shipment EDI specifications, developed by the EAA and commissioned by the BMK, the Austrian CA for shipments of waste  
The EAA developed the specification for use in [EDM WasteX] and for re-use across multiple waste shipment EDI nodes

<a href="#">[EDM WasteX]</a>	An EDI web service operated by the BMK, the Austrian CA for shipments of waste, for B2A EDI between EDI nodes used by EOs and the EDI node used by the Austrian CA, [EDM] The web service implements the [WasteX] “Basic” EDI protocol.
<a href="#">[WSDL] 2.0</a>	Web Services Description Language, an [XML] language for describing web services, W3C Recommendation
<a href="#">[WSR]</a>	EU Waste Shipment Regulation 1013/2006
<a href="#">[WSR 2021 EC proposal]</a>	EU Waste Shipment Regulation, EC proposal Nov 2021
<a href="#">[XML] 1.0</a>	Extensible Markup Language, W3C Recommendation
<a href="#">[XPath] 2.0</a>	[XML] Path Language, W3C Recommendation
<a href="#">[XQuery] 2.0</a>	[XML] Query Language, W3C Recommendation
<a href="#">[XSD] 1.0</a>	[XML] Schema Definition Language, W3C Recommendation
<a href="#">[XSLT] 2.0</a>	Extensible Stylesheet Language Transformations, W3C Recommendation

## 6 Attribution

Oxygen-Light.ttf: Copyright (c) 2012, Vernon Adams (vern@newtypography.co.uk), with Reserved Font Names 'Oxygen'; published under [Open Font License 1.1](#)

Schematron: Copyright (c) 2004-2010 Rick Jelliffe and Academia Sinica Computing Centre, Taiwan; published under the [MIT license](#)

SchXslt: Copyright (c) 2018–2021 by David Maus <dmaus@dmaus.name> and released under the terms of the [MIT license](#)

svrl.xsd: modified from the schema defined in ISO/IEC 19757-3, Copyright (c) ISO/IEC 2015

## 7 Team and contributions

Name	Organisation	Role
Norbert Pfaffinger	EAA	Editor IT Business Analysis, Data Modelling & Harmonization
Martin Kaufmann	EAA	Review IT Business Analysis
Martin Schöpf	EAA	Review IT Development
Ivan Igic	EAA	Review Test
Robert Toth	EAA	Project Management
Juliana Kund	EAA	Project Management
Gernot Lorenz	BMK	Review, Approval, Project Lead
Claudia Neurauter	BMK	Review, Approval, Project Lead