GUIDE August 2022, version 1.3

DMS System Guide

Functional guide to DMS

UDVIKLINGS OG FORENKLINGS STYRELSEN

Table of contents

Introduction

1

This guide describes the functionality of the new Declaration Management System – DMS. The target group for this system guide is developers responsible for developing system-to-system (S2S) integration from own customs clearance system to DMS System-to-System.

The aim of this document is to guide an understanding of the technical setup around a system-to-system integration. While the DMS Connectivity guide on the Danish Customs and Tax Administration GitHub explains how to establish connection to the AS4 gateway, this system guide explains the message flows and the functions that can be carried out with DMS. In specific, it explains the notifications in detail.

The guide will refer to two agencies under the Danish Ministry of Taxation: the IT and Development Agency (Udviklings- og Forenklingsstyrelsen) and the Danish Customs Agency (Toldstyrelsen). The guide will also refer to the Danish Customs and Tax Administration (Skatteforvaltningen) which both agencies, together with five other agencies in the Ministry, are a part of. Until 2018 'Skatteforvaltningen' was called 'SKAT'.

The term 'company' corresponds to the term 'economic operator' in the Union Customs Code (EU-toldkodeksen).

Currently, the document covers only functionality of DMS Import and the H7 and I2 declaration types in specific. The document will be enhanced continuously to cover DMS Export and DMS Transit (NCTS) when this functionality is added.

Technical overview

2

2.1 Relation between DMS System-to-System and DMS Online

DMS can be accessed either via a system-to-system solution (S2S) called DMS System-to-System, where declarations are submitted through the AS4 gateway, or via the systems online user interface (UI) called DMS Online.

In principle, all system functions can be managed through both access points, UI or S2S. If using DMS System-to-System, the recommendation is to use the UI only to look up information to avoid the risk of mismatch of data between own backend and DMS. E.g., if a declaration is lodged via S2S integration but amended via the UI, your backend will not know which data changed via the UI.

2.2 System overview

DMS has two primary services: One for submitting declarations and one for requesting notifications, which are statuses of a declaration.

Besides the primary services, services for additional messages exist. These are services to correct/amend, invalidate, etc.

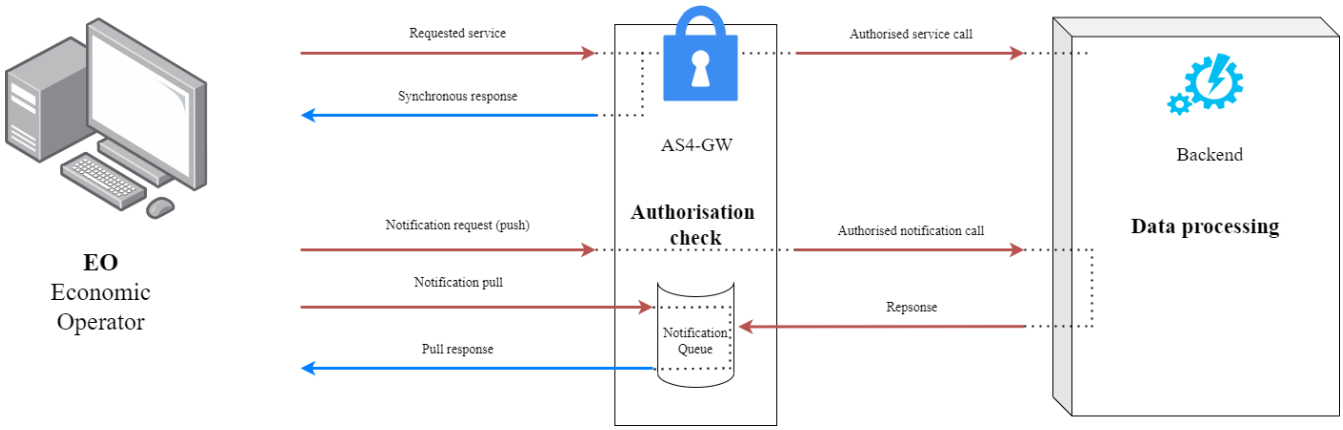

Figure 2.2.1 – System overview

The illustration shows that when lodging a declaration, the desired service is called. At the AS4 gateway the XML is syntax validated and a synchronous response is returned. In case of syntax error, the file is rejected. If syntax validation is passed, the declaration is forwarded to the backend system, where semantic validation and further processing of the declaration will be carried out.

To know if the declaration has been accepted and what its status is, another service – the notification service – needs to be called. The notification service returns all notifications available matching the request parameters. Please refer to section 4 of this document to understand how notifications are used and which notifications to expect for the various declarations and flows.

2.2.1 Services and endpoints

This section describes the S2S services that the system exposes, and how those services can be reached. The system has two main services with multiple other additional services used in special cases.
The normal function is that you call one of the exposed services, and continually call the notification service, to get the result of the operation.

The services that can be reached are:

Service	Description
DMS.Import2.Declaration.Submit	Create a declaration. Type (H1, H2 etc.) is indicated in the AS4 header.
DMS.Import2.Declaration.Amend	Amend/correct a declaration. Used to correct mistakes or add information if new arrives.
DMS.Import2.Declaration.Amend.Goodspresented	Used to amend a declaration with goods presentation data. Only used in certain scenarios.
DMS.Import2.Declaration.Invalidate	Invalidate a declaration. Used to "cancel" a declaration. The same or a modified version of the declaration can be sent with the same LRN at a later point.
DMS.Import2.Declaration.InvalidateRemissionRepay-ment	Used to invalidate a declaration on which a payment has been made. The amount is then remitted or repaid pending its payment status.
DMS.Import2.Notification	The Notification service is used to retrieve the latest up-dates on pending declaration and additional messages.

Table 2.2.1.1 – Service overview

All services are reached through the AS4 gateway on the same endpoint., The AS4-header indicates which service to call. See details on endpoint and AS4-header in the DMS Connectivity Guide.

The endpoints are:

Environment	Hostname	Port
Test	secureftpgatewaytest.skat.dk	6384
Prod	secureftpgateway.skat.dk	6384

Table 2.2.1.2 – Endpoints

This information is combined with details on the Company, creating a complete endpoint:
https://<hostname>:6384/exchange/CVR_<CVR>_UID_<UID>

Declaration submission and additional messages

All declarations and additional messages can be submitted through DMS Online, as well as through DMS System-to-System in an XML format. In both cases, the data that needs to be provided is defined by the EUCDM standard based on WCO.

All schemas used on DMS are available on the Danish Customs and Tax Administration GitHub.

Information on how to submit the declarations and the additional messages are described in the supplied DMS Connectivity Guide.

3.1 Submission

The main functionality is the submission of declarations. Declarations can either be filled out as a standard (IMA), or as a pre-lodged (IMD) declaration. IMA declarations are only sent to DMS and must therefore include a goods location. Whereas an IMD is pre-lodged and goods must be reported to Manifest and presented to Manifest upon arrival. Presentation of goods can also be done using the I2 Goods Presentation notification. Using I2 requires a customs decision as Approved Consignee (ACE). Please clarify with the Danish Customs Agency's customer service how to present. See section 3.7.

Submissions are sent to the system using the Submission XSD (see section 3.9). How to fill in the XML schema for submission and which rules to adhere to can be found in the XML guides for the different declaration types under the related declaration type folder.

When submitting a declaration, the **Declaration.Submit** service should be used, see section 2.2.1.

3.2 Correction

A correction request can be submitted to a declaration **before** the goods have been presented and **before** the declaration has been accepted (when the CWMACC notification is received, see section 4.2.3), meaning that the declaration still must be in its **pre-lodged** state (being an IMD).

> Corrections are sent to the system using the Amendment XSD. It is important that **there is at least one changed data element** when submitting a correction, if not the correction request will be rejected with the given error code (see section 4.2.6 and section 4.2.11).

Rules and details on how to fill out the data elements in the XML for each declaration type, and which data elements can be corrected, can be found on GitHub under the related declaration type folder. Go to section 3.8 to see which declaration types can be corrected.

When submitting a correction request the **Declaration.Amend** service should be used, see section 2.2.1.

3.3 Amendment

An amendment request can be submitted to a declaration **after** the goods have been presented and **after** the declaration has been accepted (the CWMACC notification is received, see section 4.2.3), meaning that the declaration must be an **IMA**, either by directly being submitted as a standard declaration or by having had the goods presented.

> Amendments are sent to the system using the Amendment XSD. It is important that **there is at least one changed data element** when submitting an amendment, if not the amendment request will be rejected with the given error code (see section 4.2.6 and section 4.2.11).

Rules and details on how to fill out the data elements in the XML for the different declaration types, and which data elements can be amended, can be found here under the related declaration type folder, go to section 3.8 to see which declaration types can be amended.

When an amendment request has been submitted, a customs officer will have to manually grant the request. Therefore, it may take some time before the expected notifications arrive.

When submitting an amendment request the **Declaration.Amend** service should be used, see section 2.2.1.

3.4 Invalidation

An invalidation request can be submitted to a declaration when invalidating a declaring is needed in case of non-repayment. If there has been a payment of customs debt on the declaration, please use the Invalidation and Repayment request (see section 3.5).

Invalidation requests are sent to the system using the Invalidation XSD. A customs officer will have to manually grant the request. Therefore, it might take some time before the expected notifications arrive and the declaration is invalidated.

Rules and details on how to fill out the data elements in the XML for invalidating the different declaration types can be found on GitHub here under the related declaration type folder, go to section 3.8 to see which declaration types can be invalidated.

When submitting an invalidation request the **Declaration.Invalidate** service should be used, see section 2.2.1.

3.5 Invalidation and Repayment

An invalidation and repayment request can be submitted to a declaration **after** the goods have been presented, **after** the declaration has been accepted (the CWMACC notification is received, see section 4.2.3) and **after** the customs debt has been paid.

Invalidation and repayment requests are sent to the system using the Invalidation and Repayment XSD. A customs officer will have to manually grant the request. Therefore, it might take some time before the expected notifications arrive and the declaration is invalidated.

Rules and details on how to fill out the data elements in the XML for invalidating the different declaration types can be found on GitHub here under the related declaration type folder, go to section 3.8 to see which declaration types an invalidation and repayment request can be submitted to.

When submitting an invalidation and repayment request the **Declaration.InvalidateRemissionRepayment** service should be used, see section 2.2.1.

3.6 Remission and Repayment

Currently not available as a separate functionality. Is tentatively planned for Release 2 – 2022.

3.7 I2 - Goods Presentation

Most EOs will not use this, as the main presentation notification is to be delivered through Manifest.

> Please implement and use the I2 Presentation notification only if explicitly told so during your onboarding process.

An I2 Goods Presentation notification can be submitted to a pre-lodged H7 declaration (IMD) when the goods are to be presented. The I2 Goods Presentation notification is used for some specific scenarios but contains similar information as the one delivered to Manifest.

I2 Goods Presentation notifications are sent to the system using the I2 XSD. The Goods Presentation notification follows the I2 EUCDM standard, where details on specific data elements and their usage can be seen in the provided schema here.

When submitting an I2 Goods Presentation notification the **Declaration.Amend.Goodspresented** service should be used, see section 2.2.1.

3.8 Overview of declaration types with their possible additional messages

This section displays an overview of declaration types and additional messages. **Note** that some declaration types are additional messages in themselves, e.g. the I2 Goods Presentation declaration, which is a goods presentation message for an H7 declaration.

Function	H1	H2	H3	H4	H5	H6	H7	I1	I2
Submission	N/A	N/A	N/A	N/A	N/A	N/A	**X**	N/A	**X**
Correction	N/A	N/A	N/A	N/A	N/A	N/A	**X**	N/A	**-**
Amendment	N/A	N/A	N/A	N/A	N/A	N/A	**X**	N/A	**-**
Invalidation	N/A	N/A	N/A	N/A	N/A	N/A	**X**	N/A	**-**
Invalidation and Repayment	N/A	N/A	N/A	N/A	N/A	N/A	**X**	N/A	**-**

Goods presentation	N/A	N/A	N/A	N/A	N/A	N/A	**X**	N/A	**-**

Table 3.8.1 – Declaration types

N/A indicates that the functionality has not been developed yet.

3.9 XSDs and test cases

In this section you can find links to all the XSD's used for submission of declarations and for the additional messages mentioned earlier. All XSD's can be found on **GitHub**.

H7			
Type	**XSD**	**Test cases**	**XML Guide**
Submission	H7 Submission XSD	H7 Submission test cases	H7 XML Guide
Correction	H7 Correction XSD	H7 Correction test cases	H7 XML Guide
Amendment	H7 Amendment XSD	H7 Amendment test cases	H7 XML Guide
Invalidation	H7 Invalidation XSD	H7 Invalidation test cases	H7 XML Guide
Invalidation and repayment	H7 Invalidation and Repayment XSD	H7 Invalidation and Repayment test cases	In progress
Repayment and remission	Not available	Not available	Not available

Table 3.9.1 – H7 XSDs and test cases

I2			
Type	**XSD**	**Test cases**	**XML Guide**
I2 – Goods presentation	I2 - Goods Presentation XSD	I2 - Goods Presentation test cases	I2 XML Guide (for H7)

Table 3.9.2 – I2 XSDs and test cases

Notifications

4

When a declaration is submitted, it passes through the system in different states that describes where the declaration is in the customs process. The system communicates the state of a declaration through **notifications**.

Receiving notifications is a two-part process, in which the user is responsible for interacting with and reacting correctly to the notifications. The two parts are **Pushing notification requests** and **Pulling notification responses**. Notifications requests should be pushed from a given time interval and pulls of notification responses from the AS4-GW queue should be interwoven in between – an example of how to work with notification pushes and pulls is described in section 4.2.2.

A notification push contains two timestamps, “from” and “to” accordingly, which describe the time interval within which notifications should be retrieved. Upon pushing a notification request a synchronous response will be elicited by the AS4-GW in the form of an ‘OK’ or an error.

When a push is performed successfully the AS4-GW will asynchronously retrieve a response from DMS and position it in a queue within the AS4-GW, from which a pull can then be performed. This will result in a synchronous response with the notifications requested in the push.

The notification holds information on which declaration the notification relates to, the state of the declaration, customs position response in relation to additional message requests, error codes (if rejected), etc.

In section 4.2 there is a list and description of each notification type, how to read it from the notification response and other relevant information. For further information on which notifications to expect from the different declarations and additional message flows, see Appendix, section 8.1.

4.1 Notification Request Design Suggestion

Notifications are requested in time intervals. E.g., every 5 minutes. See the simplified example below (seconds and milliseconds are omitted for simplicity):

It is recommended that the “from” time is set using the time stamp from the latest received notification, and that the sending time of the request is set to the “to” time for the range.

Action #	Time	Push	Sync Push Response	Pull	Sync Pull Response	Time range	Comment
1	12:00	1				11:55 - 12:00	A request is sent with a 5-minute time range.
2	12:00		1				OK response is received at 12:00.
3	12:02			1			A pull is performed at 12:02.
4	12:02				1		Notification response is received at 12:02 containing notifications from 11:55 to 12:00
5	12:05	2				12:00 - 12:05	A request is sent with the start of the range at time stamp from pull response 1 at Action #4
6	12:05		2				OK response is received 12:05
7	12:07			2			A pull is performed at 12:07.

Action #	Time	Push	Sync Push Response	Pull	Sync Pull Response	Time range	Comment
8	12:07				2		Notification response is received at 12:07 containing notifications from 12:00 to 12:05
9	12:10	3				12:05 - 12:10	A request is sent with the start of the range at time stamp from pull response 2 at Action #8
10	12:10		None				No OK response is received due to some unforeseen error.
11	12:15	4				12:05 - 12:15	A request is sent with the start of the range at time stamp from response 2 at Action #4
12	12:15		4				OK response is received 12:15
13	12:17			3			A pull is performed at 12:17
14	12:17				3		Notification response is received at 12:17 containing notifications from 12:05 to 12:15

Table 4.1.1 – Notification Request Design

4.2 List of notifications, error codes, warnings, and their text description

4.2.1 Overview of notifications

Below you see a list of all the notifications the system can produce, as well as a description of when and how they are issued. The tables below illustrate to which declaration types the various notifications can be received.

Code	H1	H2	H3	H4	H5	H6	H7	I1	I2
CWMACC	**X**	N/A	N/A	N/A	N/A	N/A	**X**	N/A	**X**
CWMCLE	**X**	N/A	N/A	N/A	N/A	N/A	**X**	N/A	**X**
CWMCTL	**X**	N/A	N/A	N/A	N/A	N/A	**X**	N/A	**X**
CWMINV	**X**	N/A	N/A	N/A	N/A	N/A	**X**	N/A	**-**
CWMRCV	**X**	N/A	N/A	N/A	N/A	N/A	**X**	N/A	**X**
CWMREJ	**X**	N/A	N/A	N/A	N/A	N/A	**X**	N/A	**X**
CWMREQ	**X**	N/A	N/A	N/A	N/A	N/A	**X**	N/A	**-**
CWMRES	**X**	N/A	N/A	N/A	N/A	N/A	**X**	N/A	**X**
CWMTAX	**X**	N/A	N/A	N/A	N/A	N/A	**X**	N/A	**X**

Table 4.2.1.1 – Which notifications can be expected for the different declaration types

N/A indicates that the functionality has not been developed yet.

Code	Title	Description
CWMACC	Declaration acceptance notification	The submitted declaration has been accepted
CWMCLE	Declaration clearance notification	Procedure is accepted and goods can be released.
CWMCTL	Declaration control notification	The declaration has been selected for control
CWMINV	Declaration invalidation notification	The declaration has been invalidated
CWMRCV	Request receival notification	The submitted request has been received
CWMREJ	Rejection notification	The declaration/request has been rejected
CWMREQ	Customs position on request notification	Customs position response on request
CWMRES	Result of request notification	Result of corrections made to the declaration, either by submitter or customs
CWM-ROG	Release of Goods	The goods have been released
CWMTAX	Customs debt notification	Notification of details on customs debt with which the declarant or his representative is informed about the details of the customs debt

Table 4.2.1.2 – List of notification types

For an overview of which notifications can be expected for submission and the additional messages, see Appendix in section 8.2.

4.2.2 Reading notifications

When requesting notifications from a given time interval, the notifications can arrive in bundles. The notification bundle is indicated by the `<Notifications> ..... </Notifications>`-tags which then can contain multiple notifications indicated by the `<Notification> ..... </Notification>`-tags, see the example below:

```
<Notifications>
    <Notification>
        <NotificationEventType>CWMxxx</NotificationEventType>
        <NotificationSID>cca1dd33-2f53-4df8-85ff-d8d1727cf972</NotificationSID>
        <Declaration>
            <MRN>21DKXARQJHQNAHO4R0</MRN>
            <LRN>NOTIFICATION_01</LRN>
            <SubmitterReferenceNumber>NOTIFICATION_01</SubmitterReferenceNumber>
              ……….
        </Declaration>
              ……….
    </Notification>
    <Notification>
        <NotificationEventType>CWMxxx</NotificationEventType>
        <NotificationSID>ea989da5-bf32-4fa7-84ae-a6c02b0a1302</NotificationSID>
        <Declaration>
            <MRN>21DKUYRRHDAKJ512R3</MRN>
            <LRN>NOTIFICATION_02</LRN>
            <SubmitterReferenceNumber>NOTIFICATION_02</SubmitterReferenceNumber>
              ……….
        </Declaration>
              ……….
    </Notification>
    ……
</Notifications>
```

Table 4.2.2.1 – Notification example

It depends on the notification type which fields and what information comes after the `<SubmitterReferenceNumber>`-element in the `<Notification>`-elements. But all notifications have **common data elements** that provide information of the declaration.

Below is an overview of the information contained in the different **common data elements**:

Element name	Description
NotificationEventType	The type of notification, defined as the code described in Table 4.2.1
NotificationSID	A unique ID used for all notifications
Declaration	Contains information that applies to the entire declaration
MRN	The MRN of the submitted declaration that the notification belongs to
LRN	The LRN of the submitted declaration
SubmitterReferenceID	The submitted LRN on the declaration

Table 4.2.2.2 – Information contained in Notifications

The following sections will give an insight into the different notification types, what to be aware of, and how to read them. For a larger overview of the different data elements, and for the notifications they occur in, see Appendix, section 8.2.

4.2.3 CWMACC - Declaration acceptance notification

The declaration acceptance notification informs the submitter that the declaration has been accepted. If there are no errors in the declaration, the notification will appear when submitting a standard declaration (IMA), or after presenting the goods declared in a pre-lodged declaration (IMD). You can see in the Appendix when CWMACC will appear in the notification flow.

Below is an example of the CWMACC-notification:

```
<Notification>
    <NotificationEventType>CWMACC</NotificationEventType>
    <NotificationSID>d6cdf594-014d-4015-8e2c-b52488fe6eb5</NotificationSID>
    <Declaration>
        <MRN>21DKYUDDGTIGAYF4R6</MRN>
        <LRN>CWMACCNOTIFICATION</LRN>
        <VersionID>1</VersionID>
        <SubmitterReferenceNumber>CWMACCNOTIFICATION</SubmitterReferenceNumber>
        <AcceptanceDateTime>
            <DateTimeString formatCode="304">20210831073944Z</DateTimeString>
        </AcceptanceDateTime>
        <SubmitterID>12345678</SubmitterID>
    </Declaration>
    <NotificationCreatedDate>
        <DateTimeString formatCode="304">20210831073955Z</DateTimeString>
    </NotificationCreatedDate>
</Notification>
```

Table 4.2.3.1 – Notification example

Besides from the common data elements described in section 4.2.2 there are only a few elements that can be retrieved from this notification:

Element name	Description
VersionID	The version number of the declaration. If corrections or changes (eg. presentation of goods) has been made to the declaration beforeit has been accepted, this number will be >1 depending on how many changes have been applied
AcceptanceDateTime	The date and time of the acceptance of the declaration
NotificationCreatedDate	The time of creation of the notification. The same as *IsuueDateTime*

Table 4.2.3.2 – Information in Notification

4.2.4 CWMCLE - Declaration clearance notification

The CWMCLE notification contains information about the clearance for the procedure, and therefore also about the release of the goods (if this has not already been done). It is sent out only after (though not necessarily directly after) the declaration has been accepted and the CWMACC notification has been sent.

```
<Notification>
        <NotificationEventType>CWMCLE</NotificationEventType>
        <NotificationSID>55776a77-686a-4356-bbcc-96076bfed6cc</NotificationSID>
        <Declaration>
            <MRN>21DKI9XIGESJOSWER9</MRN>
            <LRN>CWMCLENOTIFICATION</LRN>
            <VersionID>1</VersionID>
            <SubmitterReferenceNumber>CWMCLENOTIFICATION</SubmitterReferenceNumber>
        </Declaration>
        <AdditionalInformation>
            <StatementCode>Considered Satisfactory</StatementCode>
            <StatementTypeCode>AFB</StatementTypeCode>
        </AdditionalInformation>
        <IssueDateTime>
            <DateTimeString formatCode="304">20210825122955Z</DateTimeString>
        </IssueDateTime>
    </Notification>
```

Table 4.2.4.1 – Notification example

As seen in the sample, the notification contains a section called `<AdditionalInformation> …. </AdditionalInformation>`. The additional information contains information relevant for the trader.

Based on what is indicated in the `<StatementCode>` the additional information can be different types of information. See the table below:

Element name	Description
VersionID	The version of the declaration that has been cleared
AdditionalInformation	Contains relevant information for the submitter
StatementCode	Description of the relevant information. In the example above it shows the result of the control of the goods
StatementTypeCode	Describes what kind of additional message the Additional Message is – 'AFB' is a Customs Position Motivation.

Table 4.2.4.2 – Information in Notification example

4.2.5 CWMCTL – Control Notification

The CWMCTL notification informs the submitter that the related declaration has been selected for control. Because control has to be performed, it might take longer than usual for the declaration to go through the flow.. If the declaration goes through the control you will receive either a CWMTAX or a CWMCLE notification (depending on whether the initial declaration is an IMD or an IMA).

```
<Notification>
        <NotificationEventType>CWMCTL</NotificationEventType>
        <NotificationSID>0ed8fb30-68b2-4f64-bcbe-5bbf938641f9</NotificationSID>
        <Declaration>
            <MRN>22DKCBQY0YHWGDMOR0</MRN>
            <LRN>CWMCTLNOTIFICATION </LRN>
            <SubmitterReferenceNumber>CWMCTLNOTIFICATION </SubmitterReferenceNumber>
        </Declaration>
        <IssueDateTime>
            <DateTimeString formatCode="304">20220201135121Z</DateTimeString>
        </IssueDateTime>
    </Notification>
```

Table 4.2.5.1 – Notification example

This notification does not contain any information regarding the control. It is only sent to notify the submitter of the control.

4.2.6 CWMINV - Declaration invalidation notification

The CWMINV notification appears when an already accepted declaration has been invalidated**. For the declaration to reach the 'Invalidated' state, a customs officer (in most cases) must approve the invalidation request**. It is the final notification to be sent in the Invalidationflow.

```
    <Notification>
        <NotificationEventType>CWMINV</NotificationEventType>
        <NotificationSID>6c6a763f-bf9a-4088-93f5-bbdffc2ad226</NotificationSID>
        <Declaration>
            <MRN>21DKYUDDGTIGAYF4R6</MRN>
            <LRN>CWMINVNOTIFICATION</LRN>
            <SubmitterReferenceNumber>CWMINVNOTIFICATION</SubmitterReferenceNumber>
        </Declaration>
        <AdditionalInformation>
            <StatementCode>3</StatementCode>
            <StatementTypeCode>AFB</StatementTypeCode>
        </AdditionalInformation>
        <NotificationCreatedDate>
            <DateTimeString formatCode="304">20210912084006Z</DateTimeString>
        </NotificationCreatedDate>
    </Notification>
```

Table 4.2.6.1 – Notification example

The notification provides information on the invalidation request in the <AdditionalInformation>-element

Element name	Description
AdditionalInformation	Contains additional information about the request
StatementCode	Encoded reason for invalidation. In the example **3 is 'invalidation pr trader's request'.**
StatementTypeCode	Describes what kind of additional information the Additional information is – **'AFB' is a Customs Position Motivation**.

Table 4.2.6.2– Information in Notification example

4.2.7 CWMRCV – Receival notification

The CWMRCV notification appears when a pre-lodged declaration or additional message request has been received by the system.

There are ways to distinguish which kind of declaration or additional message request was received, therefore an explanation on how to tell the difference is provided in the following sections:

4.2.7.1 CWMRCV of pre-lodged (IMD) declaration

When submitting a pre-lodged declaration, the following CWMRCV-notification will look like the example below:

```
<Notification>
    <NotificationEventType>CWMRCV</NotificationEventType>
    <NotificationSID>287df338-e51e-440f-bdff-4e3253fbc874</NotificationSID>
    <Declaration>
        <MRN>21DKRSYEMQS5OOTGR1</MRN>
        <LRN>CWMRCVNOTIFICATION_01</LRN>
        <SubmitterReferenceNumber>CWMRCVNOTIFICATION_01</SubmitterReferenceNumber>
        <SubmitterID>12345678</SubmitterID>
    </Declaration>
    <NotificationCreatedDate>
        <DateTimeString formatCode="304">20210915172430Z</DateTimeString>
    </NotificationCreatedDate>
</Notification>
```

Table 4.2.7.1.1 – Notification example

As seen in the example, there is no further information than the common elements described in section 4.2.2. The CWMRCVnotification for a pre-lodged (IMD) declaration **does not contain** an <AdditionalMessage> element, see section below.

4.2.7.2 CWMRCV of pre-lodged (IMD) declaration with warnings

Warnings are sent after submission of a pre-lodged (IMD) declaration when there is something that the submitter should be aware of, e.g., a quota or restriction on a goods item, or if there are errors in the declaration.
Instead of initially rejecting a pre-lodged declaration with data that would have resulted in a rejection (CWMREJ) of the declaration upon goods presentation, the submitter receives warning codes in the CWMRCVnotification. The submitter then has a chance to submit a correctionrequest and thereby correct the wrongful data (see also section 4.2.11).

```
<Notification>
        <NotificationEventType>CWMRCV</NotificationEventType>
        <NotificationSID>c13210dc-25a1-4e6b-b961-2c01786f2742</NotificationSID>
        <Declaration>
            <MRN>21DKOSUS711H36XJR7</MRN>
            <LRN>CWMRCVNOTIFICATION_02</LRN>
            <SubmitterReferenceNumber>CWMRCVNOTIFICATION_02</SubmitterReferenceNumber>
             <SubmitterID>12345678</SubmitterID>
        </Declaration>
        <Error>
            <ValidationCode>DKW2012</ValidationCode>
        </Error>
        <Error>
            <ValidationCode>DKW2011</ValidationCode>
        </Error>
        <Error>
            <ValidationCode>DKW2005</ValidationCode>
        </Error>
        <NotificationCreatedDate>
            <DateTimeString formatCode="304">20210916082921Z</DateTimeString>
        </NotificationCreatedDate>
</Notification>
```

Table 4.2.7.2.1 – Notification example

There can be multiple warnings sent in the CWMRCVnotification, all shown in an `<Error>`-element. The `<ValidationCode>` element contains the warning code indicating what the error is and thereby which data elements should be corrected.

For a full list of warnings and error codes, see here.

4.2.7.3 CWMRCV of additional message

After submission of an additional message, e.g., correction/amendment/invalidation/I2/etc., the submitter will receive the CWMRCV notification when the message has been received by the system.
To be able to refer to that additional message, **an URN (which is much like an MRN) is assigned to the additional message,** and the request is now considered **an additional message** to the declaration.

```
    <Notification>
```

```
        <NotificationEventType>CWMRCV</NotificationEventType>
        <NotificationSID>c2159418-5726-4448-8f15-d5a7167d8da2</NotificationSID>
        <Declaration>
            <MRN>21DK6QXM5OVPTWONR2</MRN>
            <LRN>CWMRCVNOTIFICATION_03</LRN>
            <SubmitterReferenceNumber>CWMRCVNOTIFICATION_03</SubmitterReferenceNumber>
        </Declaration>
        <NotificationCreatedDate>
            <DateTimeString formatCode="304">20210916090154Z</DateTimeString>
        </NotificationCreatedDate>
        <AdditionalMessage>
            <URN>21DKCORFEMO96YDO05</URN>
        </AdditionalMessage>
    </Notification>
```

Table 4.2.7. 3.1 – Notification example

In the example above, the URN of the initial declaration that the request was submitted to can be seen in the top of the notification under the `<Declaration>` element, whereas the URN of the additional message/request is stated in the `<AdditionalMessage>` element.

A way to distinguish which type of request the CWMRCV belongs to is to look at the URN in the <AdditionalMessage> element:

For a CWMRCV notification received from the receival of a **correction/amendment-request**, the URN will be given as

xxxx**COR**xxxxxxxxxxx – the 5th to 7th characters is '**COR'**

For a CWMRCV notification received from the receival of an **invalidation request**, the URN will be given as

xxxx**INV**xxxxxxxxxxx – the 5th to 7th characters are '**INV'**

For a CWMRCV notification received from the receival of an **I2/Goods Presentation Notification**, the URN will be given as

xxxx**GPR**xxxxxxxxxxx – the 5th to 7th characters are '**GPR'**

The URN of the additional message will also appear in the CWMREJ notification under the `<MRN>` element if the request is rejected, and in the CWMREQ notification when, if necessary, customs has taken position on the request, see more in section 4.2.8 and 4.2.9.

4.2.8 CWMREJ - Rejection notification

4.2.8.1 CWMREJ of initial submitted declaration

When submitting a standard (IMA) declaration and sequentially receiving a CWMREJ notification it means that there are errors in the submitted declaration that results in the declaration not passing validation. Whether it is breaking a business rule or submitting an invalid code or ID, the CWMREJ notification contains information on the specifics of the error(s).

When submitting a standard (IMA) declaration and sequentially receiving a CWMREJ notification it means that there are errors in the submitted declaration that results in the declaration not passing validation. Whether it is breaking a

business rule or submitting an invalid code or ID, the CWMREJ notification contains information on the specifics of the error(s).

```
    <Notification>
        <NotificationEventType>CWMREJ</NotificationEventType>
        <NotificationSID>4dcdf061-a884-4fde-a749-91adeddb492f</NotificationSID>
        <Declaration>
            <LRN>CWMREJNOTIFICATION_01</LRN>
            <VersionID>1</VersionID>
            <SubmitterReferenceNumber>CWMREJNOTIFICATION_01</SubmitterReferenceNumber>
            <RejectionDateTime>
                <DateTimeString formatCode="304">20210916080304Z</DateTimeString>
            </RejectionDateTime>
             <SubmitterID>12345678</SubmitterID>
        </Declaration>
        <Error>
            <ValidationCode>DK1045</ValidationCode>
            <ValidationText>Error in "Importer Identification no." (3/16),  when "Additional proce-
dure" (1/11) is F49, the "Importer Identification no" (3/16) must be DK09999981.</ValidationText>
            <Pointer>
                <DocumentSectionCode>$.consignmentShipment[?(@.sequenceNumber ==0)].par-
ties[?(@.partyRoleType == IM) ].partyIdentification</DocumentSectionCode>
            </Pointer>
            <Pointer>
                <DocumentSectionCode>$.consignmentShipment[?(@.se-
quenceNumber == 0)].goodsItems[?(@.sequenceNumber ==1)].procedureCombination</DocumentSectionCode>
            </Pointer>
        </Error>
        <NotificationCreatedDate>
            <DateTimeString formatCode="304">20210916080305Z</DateTimeString>
        </NotificationCreatedDate>
    </Notification>
```

Table 4.2.8.1.1 – Notification example

As for the CWMRCV notification with warnings, it also keeps information on the error in the `<Error>`-element. However, there are a few more elements containing information on where to find the error:

In the example above a business rule was broken. The broken business rule is explained as an error code in `<ValidationCode>` and error description in `<ValidationText>`. The `<Pointer>` -element(s) indicate(s) which data element(s) in the declaration should be changed for the declaration to be accepted.

See a full list of error codes.

4.2.8.2 CWMREJ of additional message

Faulty additional messages can also be rejected. As for the CWMREJ notification of a rejected declaration the errors for the rejection of an additional message are also displayed in the the `<Error>` element of the notification.

In the `<Error>`-element an error code is contained in the `<ValidationCode>` element as well as an error description in the `<ValidationText>` element, as seen in

the example below.

```
<Notification>
        <NotificationEventType>CWMREJ</NotificationEventType>
        <NotificationSID>ea596738-a493-451e-86d4-f0c66d2e639f</NotificationSID>
        <Declaration>
            <MRN>21DK6QXM5OVPTWONR2</MRN>
            <LRN>CWMREJNOTIFICATION_02</LRN>
            <SubmitterReferenceNumber>CWMREJNOTIFICATION_02</SubmitterReferenceNumber>
            <RejectionDateTime>
                <DateTimeString formatCode="304">20210916081418Z</DateTimeString>
            </RejectionDateTime>
        </Declaration>
        <Error>
            <ValidationCode>DMS10001</ValidationCode>
            <ValidationText>Obligation error: obligation rule not met</ValidationText>
        </Error>
        <NotificationCreatedDate>
            <DateTimeString formatCode="304">20210916081419Z</DateTimeString>
        </NotificationCreatedDate>
        <AdditionalMessage>
            <MRN>21DKCORFEMO96YDO05</MRN>
        </AdditionalMessage>
    </Notification>
```

Table 4.2.8.2.1 – Notification example

Unlike the CWMREJ notification of the rejection of a declaration, this CWMREJ notification also contains an <AdditionalMessage> element, **which gives the MRN of the <AdditionalMessage>element from the CWMRCV notification of the submitted additional message** (additional message, see also section 4.2.7.3), so that the submitter knows which submitted additional message has been received and thereafter rejected.

See a full list of error codes.

4.2.8.3 CWMREJ after I2 – Goods presentation

The CWMREJ notification can appear after submission of an I2 in two scenarios:

- The I2 declaration is rejected
- The Hx declaration is rejected

Rejection of the I2 declaration

In case the I2 declaration is rejected, the CWMREJ notification will contain error code(s) and error description(s) in an <Error> element describing the error(s) in the I2 declaration. It will also contain an <Addition-alMessage> element indicating that it is the additional message that is rejected and not the initial Hx declaration. The MRN in the <AdditionalMessage> element will be on the format xxxx**GPR**xxxxxxxxxx, meaning that the additional message that is being rejected is a GPR – Goods Presentation. See also the example below.

```
<Notification>
        <NotificationEventType>CWMREJ</NotificationEventType>
```

```
        <NotificationSID>590f9ad6-b9f5-4abf-bec7-3837a219fe30</NotificationSID>
        <Declaration>
            <MRN>21DKIMEZICQN4VJJR2</MRN>
            <LRN>CWMREJNOTIFICATION_03</LRN>
            <SubmitterReferenceNumber>CWMREJNOTIFICATION_03</SubmitterReferenceNumber>
            <RejectionDateTime>
                <DateTimeString formatCode="304">20211118102017Z</DateTimeString>
            </RejectionDateTime>
        </Declaration>
        <Error>
            <ValidationCode>DK1072</ValidationCode>
            <Pointer>
                <DocumentSectionCode>$.consignmentShipment[?(@.sequenceNumber==0)].loca-
tions[?(@.locationRoleType=="14")].physicalAddress.countryCode</DocumentSectionCode>
            </Pointer>
        </Error>
        <NotificationCreatedDate>
            <DateTimeString formatCode="304">20211118102019Z</DateTimeString>
        </NotificationCreatedDate>
        <AdditionalMessage>
            <MRN>21DKGPRJBYGXSEVD00</MRN>
        </AdditionalMessage>
    </Notification>
```

Table 4.2.8.3.1 – Notification example

In this example, the CountryCode element under GoodsLocation was invalid in the I2 declaration. The Hx declaration is still in the state of 'Pending Goods Presentation' and a new (and corrected) I2 declaration can be submitted to the initial Hx declaration.

Rejection of the initial Hx declaration

The second case of receiving a CWMREJ notification after submitting an I2 declaration is when the initial Hx declaration ends up being rejected. This is the case when the pre-lodged (IMD) Hx declaration contains errors that were given as warnings in its CWMRCV notification (see section 4.2.6.2 and 4.2.11) that needed correction and have not been corrected by submitting a correction (see section 3.2). In this case, the CWMREJ notification will contain the corresponding error code(s) and error description(s) of the warning code(s) from the CWMRCV notification in an `<Error>`-element describing the error(s) in the Hx declaration. Unlike the CWMREJ notification from the rejection of an I2 declaration CWMREJ notification this scenario **does not contain an** `<AdditionalMessage>` **element**, see example below.

```
<Notification>
        <NotificationEventType>CWMREJ</NotificationEventType>
        <NotificationSID>c028b808-2989-4610-bc99-388cb120a40b</NotificationSID>
        <Declaration>
            <MRN>21DKH9EYOCY6AGJRR8</MRN>
            <LRN>CWMREJNOTIFICATION_04</LRN>
            <VersionID>2</VersionID>
            <SubmitterReferenceNumber>CWMREJNOTIFICATION_04</SubmitterReferenceNumber>
            <RejectionDateTime>
```

```
                <DateTimeString formatCode="304">20211118104422Z</DateTimeString>
            </RejectionDateTime>
        </Declaration>
        <Error>
            <ValidationCode>DK2011</ValidationCode>
            <ValidationText>Error, in "Declarant identification No." (3/18), the declar-
ant must be registered as an importer in DK.</ValidationText>
        </Error>
        <Error>
            <ValidationCode>DK2005</ValidationCode>
            <ValidationText>Error in "Declarant identification No." 3/18, the number does not ex-
ist or is not valid.</ValidationText>
        </Error>
        <NotificationCreatedDate>
            <DateTimeString formatCode="304">20211118104423Z</DateTimeString>
        </NotificationCreatedDate>
    </Notification>
```

Table 4.2.8.3.2 – Notification example

Here, an invalid Declarant ID was declared in the Hx declaration and was not corrected after receiving warnings in its CWMRCV notification, resulting in the Hx declaration being rejected when the goods where presented.
In this case, the Hx declaration will have to be resubmitted with a new LRN.

4.2.9 CWMREQ – Customs position on request notification

When submitting an additional message, the submitter will receive a CWMREQ notification when the additional message has been processed, either by the system or a Customs officer.

```
<Notification>
        <NotificationEventType>CWMREQ</NotificationEventType>
        <NotificationSID>9bad72db-c2f3-4966-8419-029120c19058</NotificationSID>
        <Declaration>
            <MRN>21DKRSYEMQS5OOTGR1</MRN>
            <LRN>CWMREQNOTIFICATION</LRN>
            <SubmitterReferenceNumber>CWMREQNOTIFICATION</SubmitterReferenceNumber>
        </Declaration>
        <AdditionalInformation>
           <StatementTypeCode>AFB</StatementTypeCode>
           <StatementDescription>Granted automatically.</StatementDescription>
       </AdditionalInformation>
        <CustomsPosition>
            <ID>1d249c92-4f74-4506-8bec-314232194725</ID>
            <Type>GRANTED</Type>
        </CustomsPosition>
        <NotificationCreatedDate>
            <DateTimeString formatCode="304">20210915172540Z</DateTimeString>
```

```
        </NotificationCreatedDate>
        <AdditionalMessage>
            <MRN>21DKCORQLZX8POKV08</MRN>
        </AdditionalMessage>
    </Notification>
```

Table 4.2.9.1 – Notification example

The notification contains information about the custom position in the `<CustomsPosition>` element, including the ID of the decision as well as the type of position.

If there are comments from customs they can be viewed in the `<AdditionalInformation> … </AdditionalInformation>` element.
Additionally, the notification contains, as for the CWMREJ notification (see section 4.2.8), the MRN of both the initial declaration as well as the MRN of the **additional message** in the `<AdditionalMessage>` element, **which shows the MRN of the `<AdditionalMessage>` element from the CWMRCV notification of the submitted additional message**.

4.2.10 CWMRES - Result of request notification

The CWMRES notification arrives after the declaration has been accepted (the CWMACC notification is received), and only in cases where the goods are presented after lodging the declaration, or when the declaration is amended after acceptance.

When submitting a pre-lodged declaration (IMD), the CWMRES notification will contain information on all changes the declaration has gone through in the process, e.g., changing type (from IMD to IMA) when goods are presented, changes in or amendment of location of goods, etc.

For a standard (IMA) declaration, the CWMRES notification will only appear if there is submitted a valid amendment request after acceptance.

```
 <Notification>
        <NotificationEventType>CWMRES</NotificationEventType>
        <NotificationSID>ab73bd28-3417-4653-8d86-8b4ec4f65b9e</NotificationSID>
        <Declaration>
            <MRN>21DKJYMHJEYP3CQDR2</MRN>
            <LRN>CWMRESNOTIFICATION</LRN>
            <VersionID>3</VersionID>
            <SubmitterReferenceNumber>CWMRESNOTIFICATION</SubmitterReferenceNumber>
            <amendment>
                <createdBy>CWM</createdBy>
                <sequenceNumber>1</sequenceNumber>
                <value>24.00</value>
                <pointer>$.consignmentShipment[?(@.sequenceNumber == 0)].goodsItems[?(@.se-
quenceNumber == 1)].declaredCustomsValue.value</pointer>
                <timestamp/>
                <declarationVersion>1</declarationVersion>
            </amendment>
```

```
            <amendment>
                <createdBy>CWM</createdBy>
                <sequenceNumber>2</sequenceNumber>
                <value></value>
                <pointer>$.type</pointer>
                <declarationVersion>2</declarationVersion>
            </amendment>
            <amendment>
                <createdBy>CWM</createdBy>
                <sequenceNumber>3</sequenceNumber>
                <value>DKFDH</value>
                <pointer>$.consignmentShipment[?(@.sequenceNumber == 0)].locations[?(@.locationRole-
Type == '14')].locationId</pointer>
                <timestamp/>
                <declarationVersion>2</declarationVersion>
            </amendment>
            <amendment>
                <createdBy>CWM</createdBy>
                <sequenceNumber>4</sequenceNumber>
                <value>0003</value>
                <pointer>$.consignmentShipment[?(@.sequenceNumber == 0)].locations[?(@.locationRole-
Type == '14')].locationAdditionalId</pointer>
                <timestamp/>
                <declarationVersion>2</declarationVersion>
            </amendment>
            <amendment>
                <createdBy>CWM</createdBy>
                <sequenceNumber>5</sequenceNumber>
                <value>U</value>
                <pointer>$.consignmentShipment[?(@.sequenceNumber == 0)].locations[?(@.locationRole-
Type == '14')].locationIdentificationType</pointer>
                <timestamp/>
                <declarationVersion>2</declarationVersion>
            </amendment>
            <amendment>
                <createdBy>CWM</createdBy>
                <sequenceNumber>6</sequenceNumber>
                <value>A</value>
                <pointer>$.consignmentShipment[?(@.sequenceNumber == 0)].locations[?(@.locationRole-
Type == '14')].locationType</pointer>
                <timestamp/>
                <declarationVersion>2</declarationVersion>
            </amendment>
            <amendment>
                <createdBy>CWM</createdBy>
                <sequenceNumber>7</sequenceNumber>
                <value>DK</value>
                <pointer>$.consignmentShipment[?(@.sequenceNumber == 0)].locations[?(@.locationRole-
Type == '14')].physicalAddress.countryCode</pointer>
                <timestamp/>
                <declarationVersion>2</declarationVersion>
```

```
            </amendment>
        </Declaration>
        <IssueDateTime>
            <DateTimeString formatCode="304">20210922115010Z</DateTimeString>
        </IssueDateTime>
    </Notification>
```

Table 4.2.10.1 – Notification example

The <amendment> element gives insight into which changes the declaration has gone through in the process.

In the example above, the pre-lodged declaration has gone through a correction before goods presentation, taking the declaration from <VersionID> from 1 to 2. Thereafter, the goods have been presented, and the location of the goods have been added to the declaration during goods presentation, taking the declaration <VersionID> from 2 to 3.
Even if the CWMRES notification arrives after submitting an I2 there is no <AdditionalMessage> element, as the changes stated in the CWMRES notification relates to the initial Hx declaration.

If there is a need to distinguish between a CWMRES notification arriving as the result of goods presentation, the example below can be used as a guide:

```
<amendment>
   <createdBy>CWM</createdBy>
   <sequenceNumber>2</sequenceNumber>
   <value></value>
   <pointer>$.type</pointer>
   <declarationVersion>2</declarationVersion>
</amendment>
```

Table 4.2.10.2 – Notification example

This <amendment> element indicates, in the <pointer> element, that the declaration has changed from IMD to IMA. There is no value in the <value>-element, as there is not directly given a value to amend, but the system changes the declaration type automatically.
If this <amendment> element is not present in the CWMRES notification, the notification is a result of a direct amendment of the Hx declaration (see section 3.3).

Element name	**Description**
Amendment	
createdBy	The system that created the amendment
sequenceNumber	Number uniquely identifying the amendment object
Value	The updated value of the amended/corrected data element
Pointer	Pointer indication the amended/corrected element
declarationVersion	The version number of the declaration that was amended/corrected

Table 4.2.10.3 – Information in Notification

4.2.11 CWMROG – Release of Goods

The CWMROG notification means the exact same thing as the CWMCLE notification (see section 4.2.4) and contains information about the release of the goods. It is sent out only after (though not necessarily directly after) the declaration has been accepted and the CWMACC notification has been sent.

> However, this notification is identical to the CWMCLE notification and is considered a bug in the system. **It does not mean that an error has occurred, and it should be handled exactly as the CWMCLE notification**.

```
<Notification>
        <NotificationEventType>CWMROG</NotificationEventType>
        <NotificationSID>55776a77-686a-4356-bbcc-96076bfed6cc</NotificationSID>
        <Declaration>
            <MRN>21DKZ5YSSN6JHMQBR3</MRN>
            <LRN>CWMROGNOTIFICATION</LRN>
            <VersionID>1</VersionID>
            <SubmitterReferenceNumber>CWMROGNOTIFICATION</SubmitterReferenceNumber>
        </Declaration>
        <AdditionalInformation>
            <StatementCode>A1</StatementCode>
            <StatementTypeCode>AFB</StatementTypeCode>
        </AdditionalInformation>
        <IssueDateTime>
            <DateTimeString formatCode="304">20210825122955Z</DateTimeString>
        </IssueDateTime>
    </Notification>
```

Table 4.2.11.1 – Notification example

Although the notification is a bug it can still appear sometimes when submitting a declaration. We therefore provide a description on how to read it:

As seen in the sample, the notification contains a section called `<AdditionalInformation> …. </AdditionalInformation>`. The additional information contains information relevant forthe trader .

Based on what is indicated in the `<StatementCode>`, the additional information can be different types of information. See the table below (or the CWMCLE notification, see section 4.2.4):

Element name	Description
VersionID	The version of the declaration that has been cleared
AdditionalInformation	Contains relevant information for the submitter

StatementCode	Description of the relevant information. In the example above it indicated the result of the control of the goods – 'A1' means 'Considered Satisfactory' (as for the CWMCLE notification)
StatementTypeCode	Describes what kind of additional message the Additional Message is – 'AFB' is a Customs Position Motivation.

Table 4.2.11.2 – Information in Notification

4.2.12 CWMTAX – Customs debt notification

When submitting a declaration, the submitter is notified by the CWMTAX notification that a calculation of customs debt has been done. The CWMTAX notification appears after the initial submission of a declaration, which for a standard (IMA) declaration is the final calculation (unless amendments are requested and granted for the customs value of a goods item), and for a pre-lodged (IMD) declaration it is only a preliminary calculation – there will be a recalculated customs debt when goods are presented, and the CWMTAX notification will be sent from the system again.

```
<Notification>
        <NotificationEventType>CWMTAX</NotificationEventType>
        <NotificationSID>685eefec-f413-425d-a055-927856d36993</NotificationSID>
        <Declaration>
            <MRN>21DKRSYEMQS5OOTGR1</MRN>
            <LRN>CWMTAXNOTIFICATION</LRN>
            <VersionID>1</VersionID>
            <SubmitterReferenceNumber>CWMTAXNOTIFICATION</SubmitterReferenceNumber>
            <DutyTaxFee>
                <Payment>
                    <ReferenceID>DK19552101:1</ReferenceID>
                    <PaymentAmount currencyID="DKK">75.1</PaymentAmount>
                    <TaxAssessedAmount>0</TaxAssessedAmount>
                </Payment>
            </DutyTaxFee>
            <GoodsShipment>
                <GovernmentAgencyGoodsItem>
                    <SequenceNumeric>1</SequenceNumeric>
                    <Commodity>
                        <DutyTaxFee>
                            <Payment>
                                <PaymentAmount currencyID="DKK">75.1</PaymentAmount>
                                <TaxAssessedAmount currencyID="DKK">75.1</TaxAssessedAmount>
                            </Payment>
                            <SpecificTaxBaseQuantity unitCode="DKK">301</SpecificTaxBaseQuantity>
                            <DeductAmount currencyID="DKK">0</DeductAmount>
                            <TaxRateNumeric>25.0</TaxRateNumeric>
```

```
                        <TypeCode>B00</TypeCode>
                    </DutyTaxFee>
                </Commodity>
            </GovernmentAgencyGoodsItem>
        </GoodsShipment>
    </Declaration>
    <IssueDateTime>
        <DateTimeString formatCode="304">20210915172600Z</DateTimeString>
    </IssueDateTime>
</Notification>
```

Table 4.2.12.1 – Notification example

As seen in the example above, the notification contains information of the payment under the `<Declaration>` element and the `<GovernmentAgencyGoodsItem>` element, each having slightly different sub-elements.

Element name	Description
Declaration	Information on the payment for the entire declaration
DutyTaxFee	DutyTaxFee captures Duty/Tax/Fee data of a particular duty/tax/fee type
Payment	This element contains information on a given payment
ReferenceID	Payment ID
PaymentAmount	The actual amount paid, or to be paid, for all items in the declaration, rounded down to one digit.
TaxAssessedAmount	Assessed amount of duty/tax/fee (includes all types of charges and duties). Assessed per duty/tax/fee type by declaration.
GoodsShipment	GoodsShipment captures the data of the shipment of the goods belonging to one particular consignment crossing the border of the Customs area
GovernmentAgencyGoodsItem	Information on the payment for the specific goods item
SequenceNumeric	The number of the goods item as given on the submitted declaration.
Commodity	Details about the properties of the goods
DutyTaxFee	DutyTaxFee captures Duty/Tax/Fee data of a particular duty/tax/fee type
Payment	This element contains information on the base of the calculation of a given payment

Element name	Description
PaymentAmount	The actual amount paid, or to be paid, for the specific item, rounded down to one digit.
TaxAssessedAmount	Assessed amount of duty/tax/fee (includes all types of charges and duties). Assessed per duty/tax/fee type by item
SpecificTaxBaseQuantity	The quantity on which a duty or tax or fee will be assessed (FreightChargeAmount + CustomsValueAmount)
DeductAmount	Amount of relief applicable from a duty or tax
TaxRateNumeric	Rate of duty or tax or fee applicable to commodities or of tax applicable to services (25.00 = 25%)
TypeCode	Code for type of tax to be applied (eg., B00 is VAT)
IssueDateTime	The time of creation of the notification. The same as *NotificationCreatedDate*

Table 4.2.12.2 – Information in Notification

As seen in the example above, the notification contains information of the payment under the `<Declaration>`-

4.2.13 Error codes and warnings

When filling out the data elements for submitting a declaration, errors can happen.

If there are errors when submitting a standard declaration (IMA) **the declaration will be rejected and a CWMREJ notification will be sent with error codes indicating the error in the declaration** (see section 4.2.7)**.** The declaration should then be resubmitted by using the submission XML (the LRN can be reused in this case) with the corrected content in the data elements.

When submitting a pre-lodged declaration (IMD) with errors, the declaration will not immediately be rejected – **instead the errors will be presented as warnings and sent in the CWMRCV notification** (see section 4.2.6). This provides a chance to **correct the declaration before the goods are presented** (see section 3.2)**.** If the errors are not corrected before the presentation of goods, the declaration will be rejected when the goods are presented, and the declaration will have to be resubmitted as for a standard declaration (IMA).

However, a warning is not always an error. It can simply be a warning about restrictions on commodity codes or other relevant information to be aware of in the declaration.

The way to distinguish a warning code from an error code, besides looking at the type of declaration (IMA or IMD), is that warnings are given on the form **DKWxxxx**, whereas error codes are in the format **DKxxxx**, **CWMxxxxxx**, **DMSxxxxxx**, etc.

The warning and error codes and their descriptions can be found on GitHub.

Error handling

5.1 Rejected declaration

How to handle a rejected declaration depends on the reason it was rejected. The error is described in the CWMREJ notification (see section 4.2.7), indicating which rule was broken, or which invalid data was entered in the declaration. **Only standard (IMA) declarations will be instantly rejected. Pre-lodged (IMD) declarations will receive warnings through CWMRCV notifications (see section 4.2.6.2).**

If the error(s) occurred from the submitter's end, the declaration can be resubmitted when the invalid data has been corrected. The LRN can be reused until the declaration is rightfully accepted (has received the CWMACC notification, see section 4.2.3).

If the error(s) occur due to system downtime or issues, the declarations can end up being rejected as well. If system downtime is not announced on 'Driftsmeddelelser', the main system or one of the external systems might be down, and the declaration data cannot be properly validated, resulting in rejected declarations. In this case you can contact TOLDST Servicedesk with information on the rejected declaration(s).

When the system is up and running again, the declarations should be resubmitted. The LRN(s) can be reused until the declaration(s) are accepted (has received the CWMACC-notification, see section 4.2.3).

5.2 Missing notifications

Sometimes the user does not receive the expected notifications. Below are sections describing some common scenarios to be aware of, and how to handle them.

5.2.1 No CWMCLE notification

Sometimes it happens that certain declarations do not go through to clearance (no CWMCLE notification is received) and are "stuck" in the flow after the CWMTAX notification.

The reason for that is usually that there is a manual work task for 'manual cash payment' pending for a customs officer to handle. This will occur in the following scenarios:

- **The declarant is a private person:** the EORI number in 3/18 – Declarant ID is **'DK09999981'** and 1/11 – Additional Procedure is **'C07'**.
- **The declarant is not registered for import with deferred payment:** the EORI number in 3/18 – Declarant ID is not registered for deferred payment.

In these cases, the declaration has to be handled manually and can get clearance only when the manual cash payment task has been handled.

> Unfortunately, there is currently no notification stating that the declaration has a manual cash payment pending, but there is a way to see it in the CWMTAX notification (see section 4.2.10).

For a **non-manual cash payment** declaration, the CWMTAX notification will look as follows:

```
<Notification>
        <NotificationEventType>CWMTAX</NotificationEventType>
        <NotificationSID>685eefec-f413-425d-a055-927856d36993</NotificationSID>
        <Declaration>
            <MRN>21DKRSYEMQS5OOTGR1</MRN>
            <LRN>CWMTAXNOTIFICATION</LRN>
            <VersionID>1</VersionID>
            <SubmitterReferenceNumber>CWMTAXNOTIFICATION</SubmitterReferenceNumber>
            <DutyTaxFee>
                <Payment>
                    <ReferenceID>DK19552101:1</ReferenceID>
                    <PaymentAmount currencyID="DKK">75.1</PaymentAmount>
                    <TaxAssessedAmount>0</TaxAssessedAmount>
                </Payment>
            </DutyTaxFee>
            <GoodsShipment>
                <GovernmentAgencyGoodsItem>
                    <SequenceNumeric>1</SequenceNumeric>
                    <Commodity>
                        <DutyTaxFee>
                            <Payment>
                                <PaymentAmount currencyID="DKK">75.1</PaymentAmount>
                                <TaxAssessedAmount currencyID="DKK">75.1</TaxAssessedAmount>
                            </Payment>
                            <SpecificTaxBaseQuantity unitCode="DKK">301</SpecificTaxBaseQuantity>
                            <DeductAmount currencyID="DKK">0</DeductAmount>
                            <TaxRateNumeric>25.0</TaxRateNumeric>
                            <TypeCode>B00</TypeCode>
                        </DutyTaxFee>
                    </Commodity>
                </GovernmentAgencyGoodsItem>
            </GoodsShipment>
        </Declaration>
        <IssueDateTime>
            <DateTimeString formatCode="304">20210915172600Z</DateTimeString>
        </IssueDateTime>
    </Notification>
```

Table 5.2.1.1 – Notification example

And for a **manual cash payment** declaration, the CWMTAX notification will look as follows:

```
<Notification>
        <NotificationEventType>CWMTAX</NotificationEventType>
        <NotificationSID>685eefec-f413-425d-a055-927856d36993</NotificationSID>
        <Declaration>
```

```
            <MRN>21DKRSYEMQS5OOTGR1</MRN>
            <LRN>CWMTAXNOTIFICATION</LRN>
            <VersionID>1</VersionID>
            <SubmitterReferenceNumber>CWMTAXNOTIFICATION</SubmitterReferenceNumber>
            <DutyTaxFee>
                <Payment>
                    <ReferenceID>af2c8a94-e617-11eb-a177-1eb09731c923</ReferenceID>
                    <PaymentAmount currencyID="DKK">75.1</PaymentAmount>
                    <TaxAssessedAmount>0</TaxAssessedAmount>
                </Payment>
            </DutyTaxFee>
            <GoodsShipment>
                <GovernmentAgencyGoodsItem>
                    <SequenceNumeric>1</SequenceNumeric>
                    <Commodity>
                        <DutyTaxFee>
                            <Payment>
                                <PaymentAmount currencyID="DKK">75.1</PaymentAmount>
                                <TaxAssessedAmount currencyID="DKK">75.1</TaxAssessedAmount>
                            </Payment>
                            <SpecificTaxBaseQuantity unitCode="DKK">301</SpecificTaxBaseQuantity>
                            <DeductAmount currencyID="DKK">0</DeductAmount>
                            <TaxRateNumeric>25.0</TaxRateNumeric>
                            <TypeCode>B00</TypeCode>
                        </DutyTaxFee>
                    </Commodity>
                </GovernmentAgencyGoodsItem>
            </GoodsShipment>
        </Declaration>
        <IssueDateTime>
            <DateTimeString formatCode="304">20210915172600Z</DateTimeString>
        </IssueDateTime>
    </Notification>
```

Table 5.2.1.2 – Notification example

The main difference between these two notifications can be found in the **Payment** segment on the declaration level:

Non-manual cash payment:

```
            <DutyTaxFee>
                <Payment>
                    <ReferenceID>DK19552101:1</ReferenceID>
                    <PaymentAmount currencyID="DKK">75.1</PaymentAmount>
                    <TaxAssessedAmount>0</TaxAssessedAmount>
                </Payment>
            </DutyTaxFee>
```

Table 5.2.1.3 – Notification example

Manual cash payment:

```
<DutyTaxFee>
    <Payment>
        <ReferenceID>af2c8a94-e617-11eb-a177-1eb09731c923</ReferenceID>
        <PaymentAmount currencyID="DKK">75.1</PaymentAmount>
        <TaxAssessedAmount>0</TaxAssessedAmount>
    </Payment>
</DutyTaxFee>
```

Table 5.2.1.4 – Notification example

> Notice that when there is a manual cash payment pending on the declaration, an UUID will be shown in the <ReferenceID>-element instead of the Declarant ID.

This is how one can ascertain if the reason that a declaration does not automatically go through to clearance and receives a CWMCLE notification is that it has a pending manual cash payment for a customs officer to handle.

5.2.2 No CWMTAX notification

There are certain scenarios in which a declaration does not trigger a CWMTAX notification. This can be the case when:

- The submitted declaration contains an IOSS number
- The data element 1/11 – Additional procedure is C08 (except for tobacco, alcohol, perfume and toilet water)

5.2.3 No CWMRCV notification/No notifications

The user should always be able to pull notifications from a given time interval. If the user keeps getting empty notifications for a correct time interval, it might be due to system downtime. If system downtime is not announced on 'Driftsmeddelelser', the main system or one of the external systems might be down, and the declaration data cannot properly be received by the system. In this case contact TOLDST Servicedesk with information on the declaration(s) with missing notifications.

When the system is up and running again, the declarations should be resubmitted. The LRN(s) can be reused until the declaration(s) are accepted (has received the CWMACC notification, see section 4.2.3).

Getting access

As mentioned earlier, DMS can be accessed either via S2S integration or by using the system's UI (the Trader Portal). Regardless of which access is required, a prerequisite is that an agreement of access has been granted by Toldstyrelsen.

For both UI and S2S you will initially get access to the test environment called TFE – Test for Erhverv. TFE can be used to prepare for production before going live on the production environment.

6.1 DMS Online

Once an agreement of access is in place, login to DMS Online requires that the employee who needs access has a personal employee certificate (Medarbejdercertifikat – MOCES), as login is handled through NemId. It is the LRA (Local Rights Administrator) of the economic operator that can grant roles to the employees who need access. Roles are managed on TastSelv Erhverv here: skat.dk/tastselverhverv

On Github we have a guide that explains how roles are assigned to the employees. The guide is in Danish and named 'Vejledning i brugeoprettelse til DMS'.

Once the roles are assigned the employee can login. The URLs are:

TFE-environment - https://tfe.toldsystemet.toldst.dk/swp.trader.customs

PROD-environment - https://toldsystemet.toldst.dk/swp.trader.customs

Add links to usermanual, clickguides and other onboarding material when provided.

6.2 DMS System-to-System

A system certificate is required in order to access the AS4 gateway. Our DMS Connectivity Guide found on GitHub explains in detail how the connectivity to the AS4 gateway is enabled.

The system certificate needs to be granted the right roles as well. It is the LRA (Local Rights Administrator) of the economic operator that can grant roles to the employees who need access. Roles are managed on TastSelv Erhverv here: skat.dk/tastselverhverv

On Github we have a guide that explains how roles are assigned to the employees. The guide is in Danish and named Vejledning i brugeoprettelse til DMS.

Verifying functionality

7

To verify the functionality of the declaration types and additional messages, as well as the ability to requests and receive notifications, we recommend that you follow the Basic Test cases found on GitHub.

Appendix

8

8.1 Flow of declarations, and related notifications

This section describes which notifications the system produces, and when in the flow the notification is produced.

The diagrams in the coming sections follow the annotation shown on **Fejl! Henvisningskilde ikke fundet.**

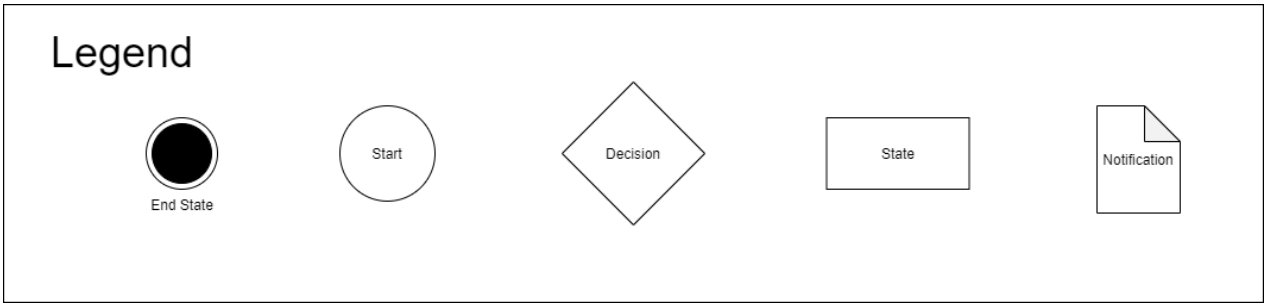

Figure 8.1.1 – Overview of diagram notation.

The End State represents a state where the flow is terminated in some respect and will not continue. An End State can trigger a notification in some cases. In most cases an End State signifies that the declaration was rejected, accepted, or converted to another flow.

The Start State signifies the starting point of a flow. There are therefore only transitions out of a Start State, and it is not possible to return, in technical terms. The start state is the first state from which it is possible for the system to issue notifications that the notification service can retrieve, meaning that it is past the semantic validation in the AS4 gateway.

A Decision State is a state from which it is possible to take one of many transitions. Only one transition will be used, and each transition out of the Decision State will be clearly labelled with the conditions necessary to take that transition. Most Decision States relate to a significant analysis performed in the flow, such as validation.

A State is the simplest construct in that it only offers one transition out, which means that this transition will always be taken.

A Notification, signified by the paper symbol, is not a state. It is therefore not possible to transition into a Notification. An arrow from any of the other symbols in the diagram to a Notification signifies that that symbol will issue a Notification which can be retrieved by the Notification service.

8.1.1 H7 Notification flows

8.1.1.1 Submission

There are two kinds of declarations to submit: a standard declaration and a pre-lodged declaration. When the goods are presented to an IMD declaration, it will turn into an IMA declaration and go through the IMA notification flow

8.1.1.1.1 Pre-lodged IMD

The flow for IMD (pre-lodged declarations) is shown in Figure 8.1.1.1.1 below.

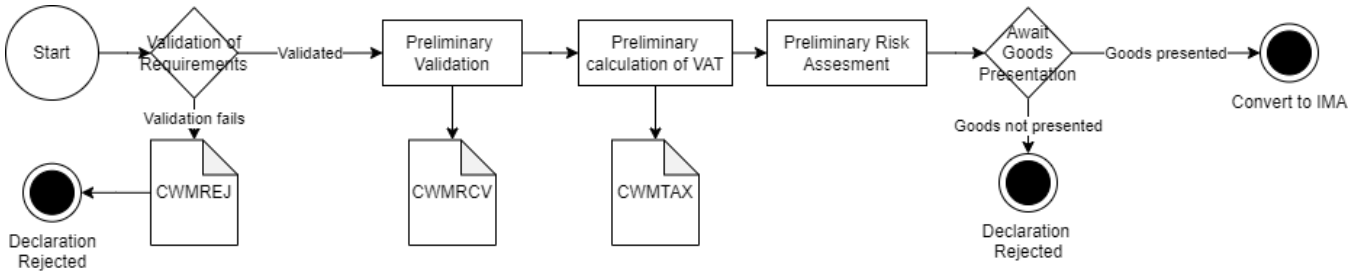

Figure 8.1.1.1.1 – The flow for an IMD declaration.

As shown in the diagram, an IMD declaration can generate the following notifications:

Title	Code	Description
Declaration Accepted Notification	CWMRCV	The submitted declaration is received.
Customs Debt Trader Notification	CWMTAX	Notification of customs debt.
Declaration Rejection Notification	CWMREJ	Declaration has been rejected.

An IMD declaration is pre-lodged which means that information about the declaration is sent to the system before the declaration has arrived in the country. An IMD therefore also has preliminary validations, risk assessment and calculations of VAT. The IMD declaration is converted to an IMA when a goods presentation declaration has been declared. From this point the declaration therefore follows the I2 declaration flow and will therefore also produce the same notifications. To sum up, an IMD declaration will have both preliminary validation, risk assessment and calculation of VAT, and a final validation, risk assessment calculation of VAT.

8.1.1.1.2 Standard IMA

The flow for IMA (standard declarations) is shown in Figure 8.1.1.1.2 below.

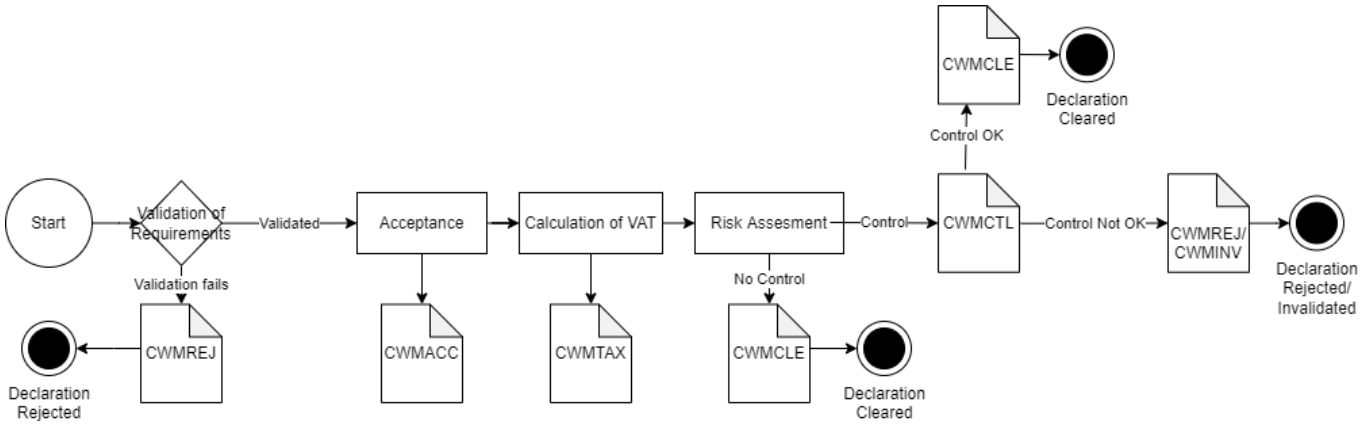

Figure 8.1.1.1.2 – The flow for an IMA declaration.

As shown in the diagram, an IMA declaration can generate the following notifications:

Title	Code	Description
Declaration Accepted Notification	CWMACC	The submitted declaration is accepted.

Customs Debt Trader Notification	CWMTAX	Notification of customs debt.
Declaration Clearance Notification	CWMCLE	Procedure is accepted and goods can be released.
Declaration Rejection Notification	CWMREJ	Declaration has been rejected.
Declaration Control Notification	CWMCTL	Declaration has been selected for control.
Declaration Invalidation Notification	CWMINV	Declaration has been invalidated.

An IMA declaration is submitted directly as the goods are presented or following an IMD after goods are presented. The IMA has final validations, risk assessment and calculations of VAT. The declaration can be selected for control based on preliminary or final risk assessment.

8.1.1.2 Correction

Flow for Correction of a declaration is shown in Figure 8.1.1.1.1.

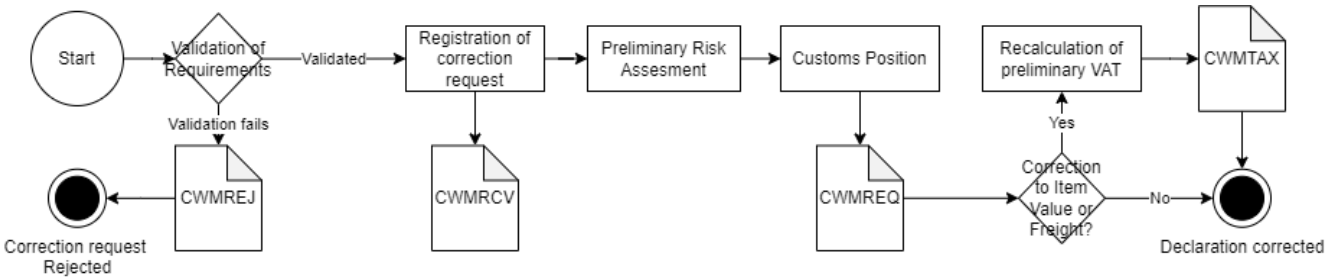

Figure 8.1.1.2.1 – the flow for a Correction

As shown in the diagram, a correction can generate the following notifications:

Title	Code	Description
Declaration Accepted Notification	CWMRCV	The submitted correction application is received.
Declaration Rejection Notification	CWMREJ	Correction application has been rejected.
Customs Position on Message Notification	CWMREQ	Decision on correction application.
Customs Debt Trader Notification	CWMTAX	Notification of customs debt.

A correction can be submitted to an IMD before goods are presented. After submission of a correction the declaration will go through preliminary validations, risk assessment and, depending on if the correction was for change of item value or freight, a recalculation of VAT. If so, there will be sent a new CWMTAX notification with the preliminary VAT.

8.1.1.3 Amendment

Flow for Amendment of a declaration is shown in Figure 8.1.1.3.

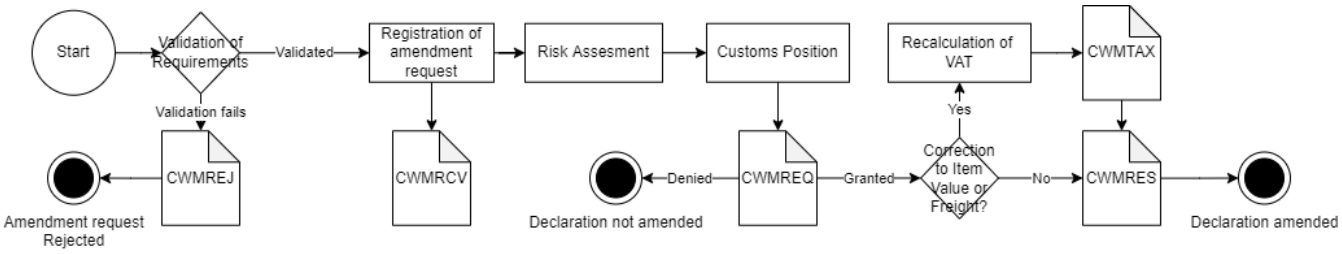

Figure 8.1.1.3 – the flow for an Amendment

As shown in the diagram, an amendment can generate the following notifications:

Title	Code	Description
Declaration Accepted Notification	CWMRCV	The submitted amendment application is received.
Declaration Rejection Notification	CWMREJ	Amendment application has been rejected.
Customs Position on Message Notification	CWMREQ	Decision on amendment application.
Customs Debt Trader Notification	CWMTAX	Notification of customs debt.
Corrected Declaration Notification	CWMRES	The result of the amendment to the declaration.

A correction can be submitted to an IMA after goods are presented. After submission of an amendment the declaration will go through validations, risk assessment and, depending on if the amendment was for change of item value or freight, a recalculation of VAT. If so, there will be sent a new CWMTAX notification with the preliminary VAT. There will be sent a notification, CWMRES, with the results of the amendment.

8.1.1.4 Invalidation

The flow for Invalidation of a declaration is shown below in Figure 8.1.1.4.1.

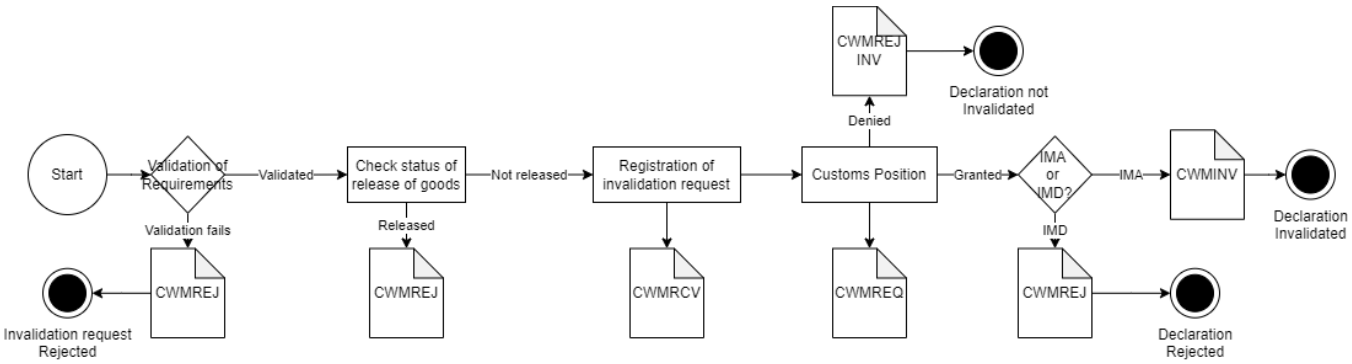

Figure 8.1.1.4.1 – the flow for an Invalidation

As shown in the diagram, an invalidation can generate the following notifications:

Title	Code	Description

Declaration Accepted Notification	CWMRCV	The submitted invalidation application is received.
Declaration Rejection Notification	CWMREJ	The declaration/invalidation application has been rejected.
Customs Position on Message Notification	CWMREQ	Decision on invalidation application.
Declaration Invalidation Notification	CWMINV	Declaration has been invalidated.

The invalidation request can be sent before the release of goods. If the declaration is an IMD, the declaration will be rejected, and a new one with the same LRN can be submitted. If the declaration is an IMA the declaration will be invalidated, and a new one with the same LRN cannot be submitted, the LRN has to change.

8.1.1.5 Invalidation and Repayment

The flow for in Invalidation and Repayment of a declaration is shown below in Figure 8.1.1.5.1

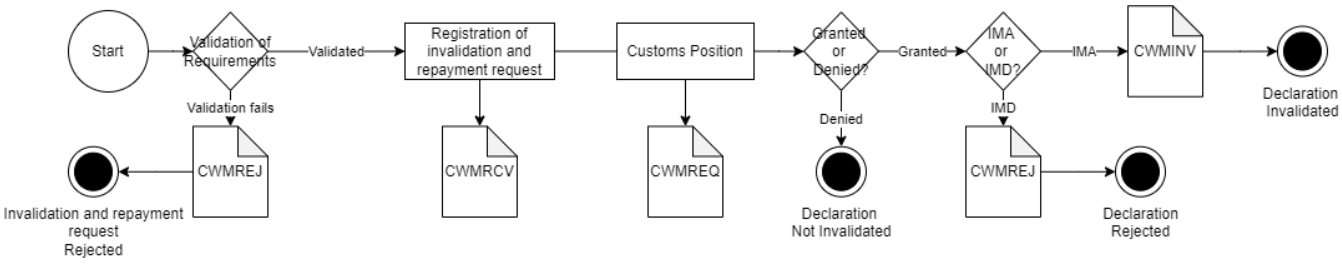

Figure 8.1.1.5.1 – the flow for Invalidation and Repayment

As shown in the diagram, an invalidation and repayment can generate the following notifications:

Title	Code	Description
Declaration Accepted Notification	CWMRCV	The submitted invalidation application is received.
Declaration Rejection Notification	CWMREJ	The declaration/invalidation application has been rejected.
Customs Position on Message Notification	CWMREQ	Decision on invalidation application.
Declaration Invalidation Notification	CWMINV	Declaration has been invalidated.

The Invalidation and Repayment request can be sent after payment of VAT. If the declaration is an IMD, the declaration will be rejected, and a new one with the same LRN can be submitted. If the declaration is an IMA the declaration will be invalidated, and a new one with the same LRN cannot be submitted, the LRN has to change.

8.1.1.6 I2 – Goods presentation

The flow for an I2 Presentation Notification flow is shown below in Figure 8.1.1.6.1.

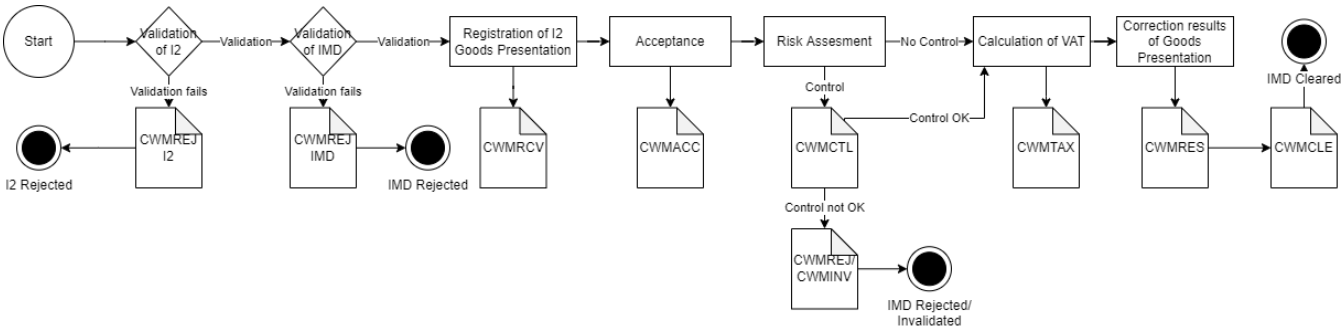

Figure 8.1.1.6.1 – Flow for an I2 Presentation Notification

As shown in the diagram, an I2 declaration can generate the following notifications:

Title	Code	Description
Declaration Accepted Notification	CWMACC	The declaration is accepted.
Customs Debt Trader Notification	CWMTAX	Notification of customs debt.
Declaration Clearance Notification	CWMCLE	Procedure is accepted and goods can be released.
Declaration Rejection Notification	CWMREJ	Declaration has been rejected.
Declaration Accepted Notification	CWMRCV	The submitted I2 declaration is received.
Corrected Declaration Notification	CWMRES	The results of the I2 goods presentation to the declaration.
Declaration Control Notification	CWMCTL	Declaration has been selected for control.
Declaration Invalidation Notification	CWMINV	Declaration has been invalidated.

An I2 can be submitted to present goods for an IMD declaration. If the I2 declaration fails validation, only the I2 declaration is rejected, and a new I2 declaration can be submitted. If the I2 declaration passes validation, but the IMD declaration fails the final validation (i.e., as a result of not correcting non-valid data as given by warnings), the whole declaration is rejected. A new IMD/IMA declaration should be submitted if needed.

If all validations pass, the now merged declaration will go through final risk assessment and VAT calculation. A notification with the results of the I2 goods presentation for the IMD declaration will be generated.

UDVIKLINGS OG FORENKLINGS STYRELSEN

www.ufst.dk
ISBN [000-00-0000-0000-0]

Udviklings- og Forenklings-styrelsen er en del af Skatteforvaltningen