



## Customs & Excise

# Web Services Specifications for Payer and Importer Reports

*The information in this document is provided as a guide only and is not professional advice, including legal advice. It should not be assumed that the guidance is comprehensive or that it provides a definitive answer in every case.*

## Contents

Document References .....	3
Abbreviations Used in This Document .....	3
Introduction.....	4
1 Digital Signatures.....	5
2 Public Interface Testing (PIT).....	6
3 Web Services .....	7
3.1 Handshake .....	7
3.2 Period (monthly) details – summary report for Payers.....	8
3.3 Period (monthly) details – combined taxes report for Payers .....	9
3.4 Period (monthly) details – tax type report for Payers.....	10
3.5 Period (monthly) details – combined taxes report for Importers.....	12
3.6 Daily details – summary report for Payers .....	13
3.7 Daily details – combined taxes report for Payers.....	14
3.8 Daily details – tax type report for Payers .....	16
3.9 Unpaid declarations report for Payers .....	17
3.10 Balance .....	18
4 ROS Error Codes.....	19

Version Control		
Version	Date	Change
1.0	22/07/2025	Initial document

## Document References

Reference
1. <i>"Customs and Excise REST Web Service Integration Guide"</i>
2. <i>"Customs &amp; Excise – Web Services Common Specification"</i>

## Abbreviations Used in This Document

Abbreviation	Description
C&E	Customs and Excise
EORI	Economic Operators Registration and Identification
PIT	Public Interface Testing
ROS	Revenue Online Service

## Introduction

This document details the ROS web services specification for retrieval of Customs & Excise transactions.

This document supersedes previous “*Web Services Specifications for Transactions Retrieval*” document, although specifications of the services, except for “handshake”, (sections 3.2 – 3.10) remain unchanged from the previous document.

These web services are built on industry standard technologies. They are available over the Internet as synchronous REST services and ensure the same level of privacy and security as the ROS web site.

This document covers the following Customs & Excise web services:

- Handshake
- Monthly summary report for Payers
- Monthly combined taxes report for Payers
- Monthly tax type report for Payers
- Monthly combined taxes report for Importers
- Daily summary report for Payers
- Daily combined taxes report for Payers
- Daily tax type report for Payers
- Unpaid declarations report for Payers
- Balance

Payers’ transaction lists are provided on either daily or a monthly basis depending on the type of Payer. There is a distinction between non-large and large payers, large payers are those who’s monthly import declaration submissions exceed a defined threshold. Traders deemed as large payers will need to request transaction lists using the daily details endpoints. Revenue will contact large traders informing them of the requirement to retrieve transaction lists daily. Section 2 of the document outlines each of the web services available and indicates if the service is to be used by large or normal payers, or if it is available to all payers.

## 1 Digital Signatures

Any ROS web service request that either returns confidential information or accepts submission of information must be digitally signed. This must be done using a digital certificate that has been previously retrieved from ROS.

The digital signature must be applied to the message in accordance with the modified HTTP Signatures specification as described in *"Customs & Excise REST Web Service Integration Guide"* available on Revenue.ie website.

The digital signature ensures the integrity of the document. By signing the document, we can ensure that no malicious intruder has altered the document in any way. It can also be used for non-repudiation purposes.

If a valid digital signature is not attached, a HTTP status code of 401 (Unauthorised) will be returned. The message body will provide more information on the details of the problem.

For REST GET requests the message to be signed is an empty string.

## 2 Public Interface Testing (PIT)

The [Handshake service](#) is the only service listed in the document available in the PIT environment.

To call the transaction handshake service the request message must be digitally signed, as detailed in [section 1](#).

The steps required to retrieve a successful response from the transaction handshake service are a prerequisite to calling all other listed services.

The document has provided sample response messages for all other web services not provided in the PIT environment.

## 3 Web Services

The following sections describe the operation of each web service.

If successful, the appropriate message will be processed and a synchronous JSON response message will be returned.

If unsuccessful, the synchronous response will be either HTTP code or a *ValidationErrors* message containing a ROS error code.

The full web service URL consists of environment-specific URL and the service-specific endpoint, e.g.:

`https://www.ros.ie/customs/webservice/v1/rest/transactions/balance`

Environment URLs:

Live:	<code>https://www.ros.ie/customs/webservice/v1/rest</code>
Public Interface Test (PIT):	<code>https://softwaretestnextversion.ros.ie/customs/webservice/v1/rest</code>

### 3.1 Handshake

Purpose:	Validate connectivity and digital signature.
Http Method:	GET
Content-type	application/json
Endpoint:	/handshake
Parameters:	None
Successful response:	When the connectivity is established, and the request signed correctly. <pre>{"connectionStatus": "SUCCESS"}</pre>
Unsuccessful response format:	When the connectivity is established, but the request was not signed correctly or there is an issue with the webservice. <pre>{"validationErrors":[{"code":"string", "description":"string"}]}</pre> Any other response – including empty, 4xx or 5xx HTTP status etc. – when the connectivity has not been established or the request is incorrect.

## 3.2 Period (monthly) details – summary report for Payers

Purpose:	Retrieve transactions' summary for given period broken down by tax and by day.
Http Method:	GET
Endpoint:	/transactions/periods/{period}/payer-summary-report
Parameters:	<b>period</b> : first day of accounting period in yyyyymmdd format, e.g. 20221101
Successful response format:	<pre>{   "eori": "string",   "period": "string",   "taxTotal": 0,   "taxBreakdown": [     {       "taxType": "string",       "payableAmount": 0     }   ],   "dailyBreakdowns": [     {       "date": "string",       "taxTotal": 0     }   ] }</pre>
Sample Response	<pre>{   "timestamp": 1660211772103,   "eori": "IE0006286E",   "period": "20220801",   "taxTotal": 400.0,   "taxBreakdown": [{     "taxType_4_3": "A00",     "payableAmount_4_6": 150.0   }, {     "taxType_4_3": "B00",     "payableAmount_4_6": 250.0   }],   "dailyBreakdowns": [{     "date": "20220810",     "taxTotal": 200.0   }, {     "date": "20220811",     "taxTotal": 200.0   }] }</pre>



### 3.3 Period (monthly) details – combined taxes report for Payers

Purpose:	Retrieve list of all paid MRNs with taxes broken down by groups, within given period.
Http Method:	GET
Endpoint:	/transactions/periods/{period}/payer-combined-taxes-report
Parameters:	<b>period</b> : first day of accounting period in yyyyymmdd format, e.g. 20221101
Successful response format:	<pre>{   "eori": "string",   "period": "string",   "paidOrders": [     {       "mrn": "string",       "version": 0,       "amendment": false,       "payer": "string",       "importer": "string",       "importerName": "string",       "declarant": "string",       "declarantName": "string",       "dtReceived": "2022-12-01T14:10:41.884Z",       "taxTotal": 0,       "totalDuty": 0,       "vatOnDuty": 0,       "totalExcise": 0,       "vatOnExcise": 0,       "postponedVat": 0,       "lrn": "string",       "ucr": "string",       "commericalTransportDoc": "string"     }   ] }</pre>
Sample Response	<pre>{   "timestamp": 1660211772103,   "eori": "IE0006286E",   "period": "20220801",   "paidOrders": [{     "mrn": "22IEDUB4BBFC22PER2",     "version": 1,     "amendment": false,     "declarationMsgType": "H1",     "payer": "IE0006286E",     "importer": "IE0006286E",     "importerName": "MR Test O'Neill",     "declarant": "IE4618356U",     "declarantName": "MR Test Murphy",     "dtReceived": "2022-08-11T10:56:11.903+0100",     "taxTotal": 200.0,     "totalDuty": 150,     "vatOnDuty": 50,   ] }</pre>

	<pre> "totalExcise": 0, "vatOnExcise": 0, "postponedVat": 100, "lrn": "EXA214094_06AaxY", "ucr": "123422342", "commercialTransportDoc": "N703124242" }, { "mrn": "22IEDUB4BBFC22PER2", "version": 2, "amendment": true, "declarationMsgType": "H1", "payer": "IE0006286E", "importer": "IE0006286E", "importerName": "MR Test O'Neill", "declarant": "IE4618356U", "declarantName": "MR Test Murphy", "dtReceived": "2022-08-13T17:46:11.903+0100", "taxTotal": 500.0, "totalDuty": 50, "vatOnDuty": 50, "totalExcise": 350, "vatOnExcise": 50, "postponedVat": 100, "lrn": "EXA214094_06AaxY", "ucr": "1234787878342", "commercialTransportDoc": "N703124242" }] } </pre>
--	--

### 3.4 Period (monthly) details – tax type report for Payers

Purpose:	Retrieve list of all paid MRNs with individual tax amounts, within given period.
Http Method:	GET
Endpoint:	/transactions/periods/{period}/payer-tax-types-report
Parameters:	<b>period</b> : first day of accounting period in yyyyymmdd format, e.g. 20221101
Successful response format:	<pre> {   "eori": "string",   "period": "string",   "taxDetails": [     {       "mrn": "string",       "version": 0,       "taxes": {         "additionalProp1": 0,         "additionalProp2": 0,       }     }   ] } </pre>

<p>Sample Response</p>	<pre>{   "timestamp": 1660211772103,   "eori": "IE0006286E",   "period": "20220801",   "taxDetails ": [{     "mrn": "22IEDUB4BBFC22PER2",     "version": 1,     "1D3": 0,     "1A1": 0,     "1B2": 0,     "A00": 5,     "1B3": 0,     "1D5": 0,     "A45": 0,     "B00": 5,     "1D6": 0,     "A35": 0,     "B00EX": 0,     "1S1": 0,     "1E1": 0,     "A40": 0,     "A30": 0,     "1C1": 0,     "2E2": 0,     "A20": 0   }, {     "mrn": "22IEDUB4BBFC22AZR2",     "version": 3,     "1D3": 0,     "1A1": 0,     "1B2": 0,     "A00": 5000,     "1B3": 200,     "1D5": 0,     "A45": 0,     "B00": 50,     "1D6": 0,     "A35": 0,     "B00EX": 0,     "1S1": 0,     "1E1": 0,     "A40": 0,     "A30": 0,     "1C1": 0,     "2E2": 0,     "A20": 0   }] }</pre>
------------------------	--

### 3.5 Period (monthly) details – combined taxes report for Importers

Purpose:	Retrieve list of all paid MRNs with taxes broken down by groups for an importer, within given period.
Http Method:	GET
Endpoint:	/transactions/periods/{period}/importer-combined-taxes-report
Parameters:	<b>period</b> : first day of accounting period in yyyyymmdd format, e.g. 20221101
Successful response format:	<pre>{   "eori": "string",   "period": "string",   "paidOrders": [     {       "mrn": "string",       "version": 0,       "amendment": false,       "payer": "string",       "payerName": "string",       "importer": "string",       "declarant": "string",       "declarantName": "string",       "dtReceived": "2022-12-01T14:10:41.884Z",       "taxTotal": 0,       "totalDuty": 0,       "vatOnDuty": 0,       "totalExcise": 0,       "vatOnExcise": 0,       "postponedVat": 0,       "lrn": "string",       "ucr": "string",       "commericalTransportDoc": "string"     }   ] }</pre>
Sample Response	<pre>{   "timestamp": 1660211772103,   "eori": "IE0006286E",   "period": "20220801",   "paidOrders": [{     "mrn": "22IEDUB4BBFC22PER2",     "version": 1,     "amendment": false,     "declarationMsgType": "H1",     "payer": "IE0006286E",     "payerName": "MR Test O'Neill",     "importer": "IE0006286E",     "declarant": "IE4618356U",     "declarantName": "MR Test Murphy",     "dtReceived": "2022-08-11T10:56:11.903+0100",     "taxTotal": 200.0,     "totalDuty": 150,     "vatOnDuty": 50,     "totalExcise": 0,   ] }</pre>

	<pre> "vatOnExcise": 0, "postponedVat": 100, "lrn": "EXA214094_06AaxY", "ucr": "123422342", "commercialTransportDoc": "N703124242" }, { "mrn": "22IEDUB4BBFC22PER2", "version": 2, "amendment": true, "declarationMsgType": "H1", "payer": "IE0006286E", "payerName": "MR Test O'Neill", "importer": "IE0006286E", "declarant": "IE4618356U", "declarantName": "MR Test Murphy", "dtReceived": "2022-08-13T17:46:11.903+0100", "taxTotal": 500.0, "totalDuty": 50, "vatOnDuty": 50, "totalExcise": 350, "vatOnExcise": 50, "postponedVat": 100, "lrn": "EXA214094_06AaxY", "ucr": "1234787878342", "commercialTransportDoc": "N703124242" }] } </pre>
--	---

### 3.6 Daily details – summary report for Payers

Purpose:	Retrieve transactions' summary for given day, broken down by tax type.
Http Method:	GET
Endpoint:	/transactions/daily/{day}/payer-summary-report
Parameters:	<b>day</b> : relevant day of accounting period in yyyyymmdd format, e.g. 20221125
Successful response format:	<pre> {   "eori": "string",   "date": "string",   "taxTotal": 0,   "taxBreakdown": [     {       "taxType": "string",       "payableAmount": 0     }   ] } </pre>

Sample Response	<pre>{   "timestamp": 1660211772103,   "eori": "IE0006286E",   "date": "20220811",   "taxTotal": 400.0,   "taxBreakdown": [{     "taxType_4_3": "A00",     "payableAmount_4_6": 150.0   }, {     "taxType_4_3": "B00",     "payableAmount_4_6": 250.0   } ]</pre>
-----------------	---

### 3.7 Daily details – combined taxes report for Payers

Purpose:	Retrieve list of all paid MRNs with taxes broken down by groups, in given day. Paid orders submitted on a given day but posted to a different period will be include the period to which the payment order was posted. If the period field is not present the payment order was posted to the accounting period the day , requested, is part of
Http Method:	GET
Endpoint:	/transactions/daily/{day}/payer-combined-taxes-report
Parameters:	<b>day</b> : relevant day of accounting period in yyyyymmdd format, e.g. 20221125
Successful response format:	<pre>{   "eori": "string",   "date": "string",   "paidOrders": [     {       "mrn": "string",       "version": 0,       "amendment": false,       "payer": "string",       "importer": "string",       "importerName": "string",       "declarant": "string",       "declarantName": "string",       "dtReceived": "2022-12-01T14:38:11.352Z",       "taxTotal": 0,       "totalDuty": 0,       "vatOnDuty": 0,       "totalExcise": 0,       "vatOnExcise": 0,       "postponedVat": 0,       "ltn": "string",       "ucr": "string",       "commericalTransportDoc": "string",       "period": "string",     }   ] }</pre>

	<pre>     }   ] } </pre>
Sample Response	<pre> {   "timestamp": 1660211772103,   "eori": "IE0006286E",   "day": "20220811",   "paidOrders": [{     "mrn": "22IEDUB4BBFC22PER2",     "version": 1,     "amendment": false,     "declarationMsgType": "H1",     "payer": "IE0006286E",     "importer": "IE0006286E",     "importerName": "MR Test O'Neill",     "declarant": "IE4618356U",     "declarantName": "MR Test Murphy",     "dtReceived": "2022-08-11T10:56:11.903+0100",     "taxTotal": 200.0,     "totalDuty":150,     "vatOnDuty": 50,     "totalExcise": 0,     "vatOnExcise": 0,     "postponedVat": 100,     "lrn": "EXA214094_06AaxY",     "ucr": "123422342",     "commercialTransportDoc": "N703124242"      }, {       "mrn": "22IEDUB4BBFC22PER2",       "version": 2,       "amendment": true,       "declarationMsgType": "H1",       "payer": "IE0006286E",       "importer": "IE0006286E",       "importerName": "MR Test O'Neill",       "declarant": "IE4618356U",       "declarantName": "MR Test Murphy",       "dtReceived": "2022-08-13T17:46:11.903+0100",       "taxTotal": 500.0,       "totalDuty":50,       "vatOnDuty": 50,       "totalExcise": 350,       "vatOnExcise": 50,       "postponedVat": 100,       "lrn": "EXA214094_06AaxY",       "ucr": "1234787878342",       "commercialTransportDoc": "N703124242"       "period": "20220101"     }   ] } </pre>

### 3.8 Daily details – tax type report for Payers

Purpose:	Retrieve list of all paid MRNs with individual tax amounts, in given day.
Http Method:	GET
Endpoint:	/transactions/daily/{day}/payer-tax-types-report
Parameters:	<b>day</b> : relevant day of accounting period in yyyyymmdd format, e.g. 20221125
Successful response format:	<pre>{   "eori": "string",   "period": "string",   "day": "string",   "taxDetails": [     {       "mrn": "string",       "version": 0,       "taxes": {         "additionalProp1": 0,         "additionalProp2": 0,         "additionalProp3": 0       }     }   ] }</pre>
Sample Response	<pre>{   "timestamp": 1660211772103,   "eori": "IE0006286E",   "day": "20220811",   "taxDetails": [{     "mrn": "22IEDUB4BBFC22PER2",     "version": 1,     "1D3": 0,     "1A1": 0,     "1B2": 0,     "A00": 5,     "1B3": 0,     "1D5": 0,     "A45": 0,     "B00": 5,     "1D6": 0,     "A35": 0,     "B00EX": 0,     "1S1": 0,     "1E1": 0,     "A40": 0,     "A30": 0,     "1C1": 0,     "2E2": 0,     "A20": 0   }, {     "mrn": "22IEDUB4BBFC22AZR2",     "version": 3,     "1D3": 0,     "1A1": 0,</pre>



	<pre> "1B2": 0, "A00": 5000, "1B3": 200, "1D5": 0, "A45": 0, "B00": 50, "1D6": 0, "A35": 0, "B00EX": 0, "1S1": 0, "1E1": 0, "A40": 0, "A30": 0, "1C1": 0, "2E2": 0, "A20": 0     }     ]     }</pre>
--	--

### 3.9 Unpaid declarations report for Payers

Purpose:	Retrieve current list of unpaid MRNs.
Http Method:	GET
Endpoint:	/transactions/payer-unpays-report
Parameters:	None
Successful response format:	<pre> {   "eori": "string",   "unpaidOrders": [     {       "mrn": "string",       "version": 0,       "taxTotal": 0     }   ] }</pre>
	<pre> {   "timestamp": 1659045615352,   "eori": "IE4618356U",   "unpaidOrders": [{     "mrn": "22IEDUB4BBFC22BPP3",     "version": "1",     "taxTotal": 1000.0   }, {     "mrn": "22IEDUB4BBFC22BPP1",     "version": "1",     "taxTotal": 2000.0   }, {     "mrn": "22IEDUB4BBFC22BPP2",     "version": "1",     "taxTotal": 2000.0   }   ] }</pre>

### 3.10Balance

Purpose:	Retrieve current Customs & Excise balances.
Http Method:	GET
Endpoint:	/transactions/balance
Parameters:	None
Successful response format:	<pre>{   "total": "string",   "cash": "string",   "deferred": "string" }</pre>
Sample Response	<pre>{   "total": "11061096.00",   "cash": "9361096.00",   "deferred": "1700000.00" }</pre>

## 4 ROS Error Codes

Certain unsuccessful calls may result in a ROS error response returned to the caller, e.g.:

```
{
  "validationErrors": [{
    "code": "ROS-300-10",
    "description": "Issue with request's timestamp."
  }]
}
```

More information on error codes and common failure scenarios is available in [“Customs & Excise – Web Services Common Specification”](#) document.