Delivered Duty Paid

Implementation Guide

version 1.01

Last updated: 12 September 2025

Contact

Postal Technology Centre - Universal Postal Union Weltpoststrasse 4, 3015 Bern - Switzerland

Phone: +41 31 350 31 11

Service Desk Portal: https://support.upu.int

UPU UNIVERSAL POSTAL UNION

Table of contents

About this document

Intended audience

This user guide is intended for postal operators who want to roll out the UPU Delivered Duty Paid (UPU DDP) solution, using the Postal Technology Centre (PTC) technologies.

Scope of this document

This document provides the guidelines on the necessary IT, operational, as well as contractual steps for being DDP-ready.

It highlights and explains some decisions to go through before starting a DDP project.

In its first version (05.09.2025) the document remains incomplete, but it will be regularly updated in the coming weeks.

The document covers both the generic UPU DDP, and, when applicable, the specificities of DDP for the US inbound flows.

For information on:

- The background of the UPU DDP solution, see "Introduction" on the next page.
- An overview of the UPU DDP APIs, see "DDP API on the PTC Gateway" on page 9.
- Obtaining an authorization token and calling the UPU DDP APIs, see "Accessing the APIs" on page 10.

How to use this manual

 However, the UPU welcomes your input. For queries or service requests, you can raise them at https://support.upu.int.

Introduction

Background

The UPU DDP solution was developed in response to regulatory changes affecting US inbound mail items. The aim is to eventually build a generic DDP solution that could be used for any country of destination.

As of August 29, 2025, the de minimis threshold—the minimum value below which imported goods are exempt from duties and taxes—was abolished for mail items entering the United States. The only remaining exemption applies to goods valued under USD 100 that are explicitly declared as gifts.

Under current US regulations, a U.S. Qualified Party (QP) is required to file the CBP International Mail Duty Worksheet with U.S. Customs and Border Protection (US-CBP) and to handle the financial settlement of any applicable duties and taxes. Currently, the UPU has an active contract with Zonos, a registered Qualified Party. Zonos powers the UPU DDP solution, enabling compliance with these regulations. Additional Qualified Parties will be onboarded as needed to meet the requirements of the Posts.

The UPU DDP solution

Building on existing UPU technologies and services, the PTC is rolling out its DDP solution in phases:

- **UPU DDP APIs**: The first component to be released via the PTC API Gateway. These APIs are intended for users who do not use the UPU's Customs Declaration System (CDS).
- **CDS and Post*Net Data Flows**: These are core to the DDP solution and represent the second phase of development.
- **Changes to UPU*Clearing Financial Flows and International Settlement**: This final phase is planned for a later stage and is subject to approval by the UPU*Clearing User Group.

How does the DDP solution work?

The table below summarizes the key steps in a DDP workflow.

	CDS User	Non-CDS User
Capturing Customs declaration data	• With a CDS interface (Kiosk, Mobile App, CDS screens) • With local system and then upload in CDS	• Local system
Calling the LCC API	• CDS calls the LCC (the call can be made from the local system)	• Local system uses the LCC API on the PTC API Gateway

<table>
<tr><th></th><th>CDS User</th><th>Non-CDS User</th></tr>
<tr><td>Informing the shipper about the LCC</td><td>• Local system (or CDS) is used to inform the shipper</td><td>• Local system is used to inform the shipper</td></tr>
<tr><td>Collecting the LLC</td><td colspan="2">Different options are possible:
• LCC paid directly by the shipper to the QP
• LCC invoiced by Post to the shipper
• LCC paid by the shipper at Post office (OTC)</td></tr>
<tr><td>Preparing the DDP-compliant ITMATT</td><td>• CDS prepares and sends the ITMATT</td><td>• Local system needs to prepare and the send the ITMATT with the DDP information coded in the Observation field, see "ITMATT" below.</td></tr>
<tr><td>Linking the Declaration ID with the S10</td><td>• CDS does that, at the time of item labelling</td><td>• Local system needs to call the QP for the linkage (API on the PTC API Gateway)</td></tr>
<tr><td>Sending the data required for filing to the QP</td><td colspan="2">If the relevant mailboxes are on Post*Net, PTC can make the copies to the selected QP. ITMATT, PREDES, and CARDIT/RESDIT are used for sending the following data to the QP:
• Item ID (S10)
• Declared value
• Country of Origin
• Transport information
If the mailboxes are not on Post*Net, you need to use another data transmission solution to the QP.</td></tr>
<tr><td>Financial settlement to US Customs</td><td colspan="2">Financial settlement is made by the QP. It means that, in case the LCC has been collected by Post, Post need to financially settle with the QP. At the moment, this will be done with the QP invoicing the Post.</td></tr>
</table>

ITMATT

UPU DDP uses ITMATT v1.

The landed cost details will be recorded in the **<observations /> (item.observations)** field of the EDI message:

Listed below are the data elements to be saved in the ITMATT.

Data	Description	Remarks	Example
DDP Flag	The flag indicating the DDP information.	fixed 3-character string	DDP
Declaration ID	The service provider's (e.g., Zonos) ID identifying the landed cost request for a declaration/shipment.	13-character ID	0m123456789ca
Landed Cost breakdown	The breakdown of the landed costs.		
	Currency Code – The associated currency code.	3-letter ISO code	CAD
	Amount – The amount assessed for a given amount type.	max 18 characters	12.00
	Amount Type – The amount type. • Duties (D) • Taxes (T) • Fees (F) • Discounts/Adjustments (A)	1 character	D,T,F,A
PaymentTo	The party (code identifier) that collects the payment.	max 10 characters	UPU CL118
SettleWith	The party (code identifier) to whom the payment is to be made/settled.	max 10 characters	UPU CL118
Hash	The hash of the above landed cost information.	max 64 characters	

The data will be stored as a single formatted string with a maximum length of 196 characters, in the following format:

```
DDP;DeclarationID;LandedCostBreakdown(CurrencyCode+Amount+AmountType,separated by `,`);PaymentTo;SettleWith;Hash
```

Example:

- Declaration ID: 0m123456789ca
- Duties: 1.00 CAD
- Taxes: 2.00 CAD
- Fees: 3.50 CAD
- Discounts: 0.00 CAD

- PaymentTo: J1CCAA, Canada Post
- SettleWith: JJ00DUS01, Zonos

```
DDP;0m123456789ca;CAD1.00D,CAD2.00T,CAD3.50F,CAD0A;J1CCAA;JJ00DUS01;HASH
```

DDP API on the PTC Gateway

This chapter provides a brief overview of the UPU DDP APIs.

For information on using the APIs, see "Accessing the APIs" on the next page.

What's available now

Developer portal (TEST)

Available at https://ptc.developer.azure-api.net/

Available APIs (Test & Production)

- Authentication (JWT): Obtains a short-lived token so your systems can call PTC services securely.
- Landed Cost Calculator: Returns the all-in import amount (duties, taxes, fees) from draft customs data, along with a Declaration ID.
- Declaration - S10 Linking: Binds a finalized item (S10) to its Declaration ID and confirms the payment state for downstream messaging.

How it works:

- Your system sends the declaration data
- Gets the landed cost and a Declaration ID
- At finalization (S10), you recalculate and collect
- ITMATT includes the DDP indicator, Declaration ID, and payment status

Integration steps

1. Request access to the production API platform.
2. Implement APIs in your system.
3. Update your retail flow.
4. Enrich ITMATT.

Accessing the APIs

This chapter outlines the step-by-step process for successfully calling the UPU DDP APIs to calculate the landed costs of mail items. The procedure involves obtaining a JSON Web Token (JWT) to authorize API requests, performing the necessary API calls to compute the landed costs, and linking the calculated costs to the corresponding mail items.

Your organization must already have an API key. Contact the PTC for more information.

UPU DDP API Access URLs/Endpoints

The following API endpoints are available for interacting with external systems or tools.

GET	Authorization token (JWT) https://api.upu.post/auth-rest/auth/{apiKey}
POST	Landed Cost Calculator https://api.upu.post/lcc-rest/calculate
POST	Get HS Codes https://api.upu.post/lcc-rest/gethscodes
POST	Link S10 IDs to Declaration https://api.upu.post/lcc-rest/links10

Test Environment for API calls

To verify the functionality of the APIs, interested postal operators can access the testing portal at: https://ptc.developer.azure-api.net/. With a built-in testing feature, postal operators can try various API operations, make test calls, and familiarize themselves with the API specifications.

Alternatively, test calls can also be made from external systems such as Postman or developer platforms to the testing endpoints indicated below.

GET	Authorization token (JWT) https://ptc.azure-api.net/auth-rest/auth/{apiKey}
POST	Landed Cost Calculator https://ptc.azure-api.net/lcc-rest/calculate
POST	Get HS Codes https://ptc.azure-api.net/lcc-rest/gethscodes

POST	Link S10 IDs to Declaration https://ptc.azure-api.net/lcc-rest/links10

Interact with the UPU DDP APIs

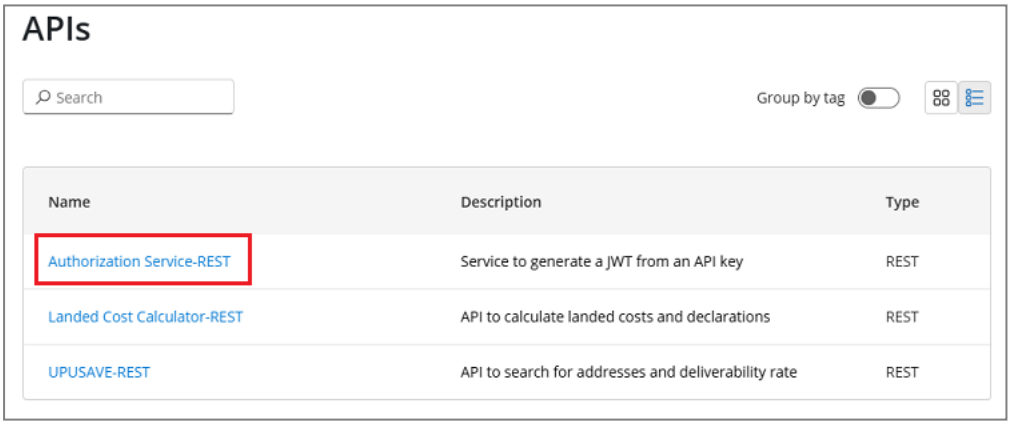
This section describes how to call the UPU DDP APIs using the test platform.

The data shown in the images are fictitious and is for illustration purposes only.

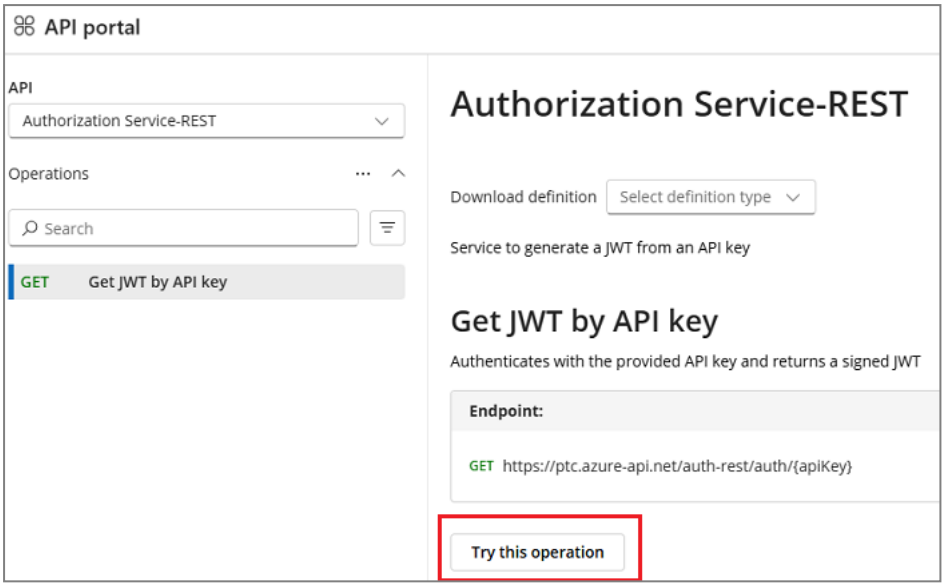
Obtain an authorization token

The calling party must obtain a JWT using the API key provided by the PTC to authorize access to the APIs. The token is valid for 24 hours, after which re-authentication is required to obtain a new one.

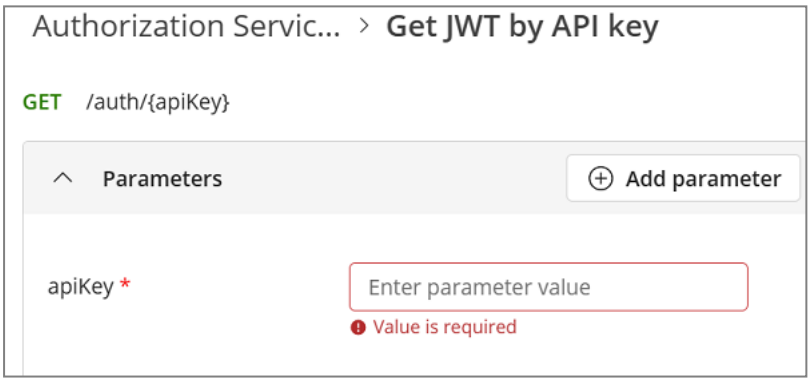
1. To access the testing portal, go to https://ptc.developer.azure-api.net/.
2. Click the **Explore APIs** button. A page listing the UPU APIs with their description is displayed.
3. To generate a token for making API calls, click the **Authorization Service-REST** link.



4. In the page that opens, click the **Try this operation** button.



5. Enter your organization's API key for generating the token in the **apiKey** field.



If the **apiKey** field is not displayed, press **CTRL+F5** on your keyboard to refresh the page. This issue is due to a known limitation in the Azure platform.

6. Click **Send**.
7. Copy and paste the JWT token (without the quotation marks) from the response to a note-pad or a document. With this token, you can now start calling the UPU DDP APIs for the landed cost calculation.

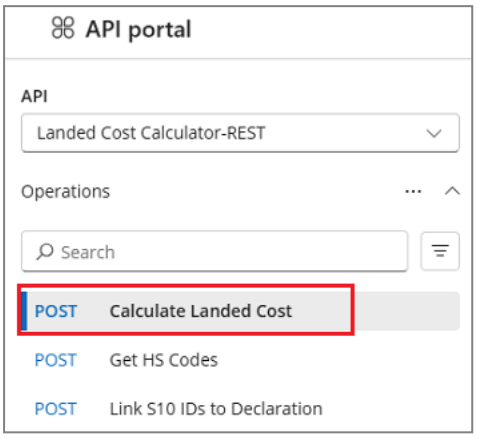
Request a landed cost calculation

The **Landed Cost Calculator** API is the API that calculates the total expense of getting the mail item from its point of origin to its final destination.

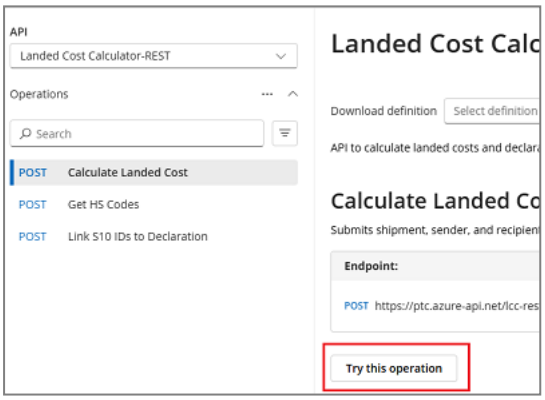
After you receive the landed cost calculation, you can link the cost to mail item (or a list of mail items) via the S10 identifier.

When calling the endpoints from an external system, include the token as an **Authorization Bearer Token** in the header.

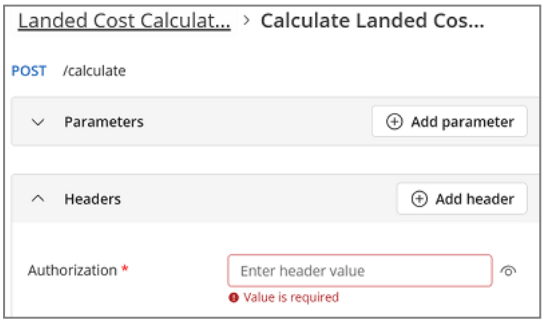
1. In the testing portal (https://ptc.developer.azure-api.net/), click **Landed Cost Calculator-REST** from the list of APIs.
2. In the page that opens, ensure that the **POST Calculate Landed Cost** operation from the left pane is selected.



3. Click the **Try this operation** button.



4. Enter the JWT value in the **Authorization** field.



5. Specify the other body request parameters and click **Send**. Note down the declaration ID received in the response. An example of a request and a response is provided below.

```
{
    "recipient": {
        "address": "456 Market St",
        "locality": "New York",
        "postalCode": "10001",
        "country": "US"
    },
    "sender": {
        "address": "40 Goldfields Road",
        "locality": "Doctor Creek",
        "postalCode": "4352",
        "country": "AU"
    },
    "declaration": {
        "items": [{
            "quantity": 1,
            "description": "T-shirt",
            "hsCode": "610910",
            "countryOfOrigin": "AU",
            "amount": 20,
            "currencyCode": "AUD"
        }],
        "natureType": "gift"
    },
    "shipment": {
        "postage": 5,
        "currencyCode": "AUD"
    },
    "landedCostCalculation": {
        "method": "DDP",
        "currencyCode": "AUD"
    }
}
```

The table below lists the parameters and their descriptions.

Parameter	Description
recipient	The recipient information.
address	The first line (e.g., street address) of the recipient's address.
locality	The locality (city or town) of the recipient.
postalCode	The postal or ZIP code of the recipient.
country	The two-letter ISO country code of the recipient.
sender	The sender information.
address	The first line (e.g., street address) of the sender's address.
locality	The locality (city or town) of the sender.
postalCode	The postal or ZIP code of the sender.
country	The two-letter ISO country code of the sender.

Parameter	Description
declaration	The declaration data.
items	The list of content piece items.
quantity	The quantity of the item.
description	The description of the item.
hsCode	The HS Code of the item.
countryOfOrigin	The item's country of origin.
amount	The cost of the item.
currencyCode	The currency in which the item is priced.
natureType	The nature or category of the declaration: • Gift • Documents • For_Resale • Not_For_Resale
shipment	The shipment or postage information.
postage	The postage or shipping fee.
currencyCode	The currency used for the postage.
landedCostCalculation	The preferences for calculating the landed costs.
method	The method to use for the landed cost calculation. The value must be DDP.
currencyCode	The currency to use for the landed cost subtotals.

A sample API response is shown below.

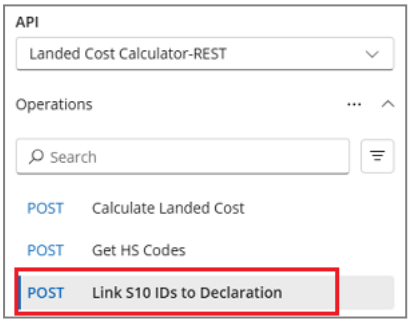

```
{
    "data": {
        "landedCosts": {
            "method": "DDP",
            "currency": "AUD",
            "subTotals": {
                "duties": 5.3,
                "taxes": 0.0,
                "fees": 4.4,
                "discounts": 0.0,
                "shippping": 5.0
            },
            "landedCostTotal": 9.7
        },
        "declaration": {
            "id": "0mv38entahsxr",
            "status": "OPEN",
            "paymentStatus": "OPEN"
        },
        "additionalInfo": ""
    }
}
```


The response data description is as follows:

Parameter	Description
data	The response data.
landedCosts	The calculated landed costs.
method	The method used for the landed cost calculation. The value shown is DDP.
currencyCode	The currency used for the landed cost calculation.
subTotals	The breakdown of the landed costs.
duties	The duties assessed for the landed cost.
taxes	The taxes assessed for the landed cost.
fees	The fees assessed for the landed cost.
discounts	The discounts assessed for the landed cost
shipping	The shipping costs.
landedCostTotal	The total amount of duties, taxes, and fees.

Parameter	Description
	landedCostTotal = duties + taxes + fees
declaration	The declaration data.
id	The ID associated with the declaration/shipment from the provider, Zonos.
status	The current status of the declaration.
paymentStatus	The status of payment for the declaration.
additionalInfo	Additional notes or information, if any.

6. Select the **POST Link S10 IDs to Declaration** operation in the left pane.



7. Click the **Try this operation** button.

8. Enter the JWT value in the **Authorization** field.
9. In the request body, enter the declaration ID from step 4.
10. To associate the declaration ID with a mail item(s), enter one or more S10 IDs.

```
{
    "declarationId": "0mv38entahsxr",
    "S10": "RR123456789AU"
}
```

11. Click **Send**.

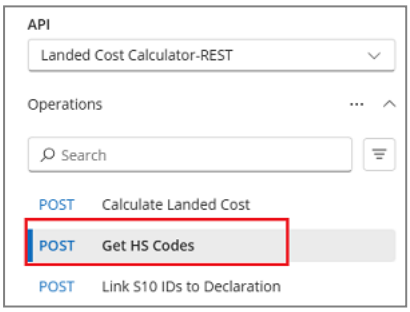
The API returns the response data details with the result set to "SUCCESS" if successful, or "FAILURE" if it fails.

```
{
    "data": {
        "result": "SUCCESS"
    }
}
```

Get HS Codes

If you only have the item description and you want to get the HS code for the item, you can use this method.

1. In the testing portal (https://ptc.developer.azure-api.net/), click **Landed Cost Calculator-REST** from the list of APIs.
2. In the page that opens, ensure that the **POST Get HS Codes** operation from the left pane is selected.



3. Click the **Try this operation** button.
4. Enter the JWT value in the **Authorization** field.
5. In the request body, enter the description of the good.

```
{
    "goodDescription": "tobacco"
}
```

6. Click **Send**.