PTV LOGISTICS

PTV Developer Migration Guide

Registration, Developer APIs, Tutorials, Onboarding

Version 1.3

Contents

Short title	PTV Developer Migration Guide
Template history	V1.0.0 dated 2025-06-18

1 Registration and Sign In

PTV Developer provides high-performance APIs designed to enhance logistics software with advanced geographical and logistical capabilities (hereinafter referred to as the "**Services**"). These APIs are modular, scalable, and customizable, enabling users to efficiently plan, calculate, and optimize road transport operations.

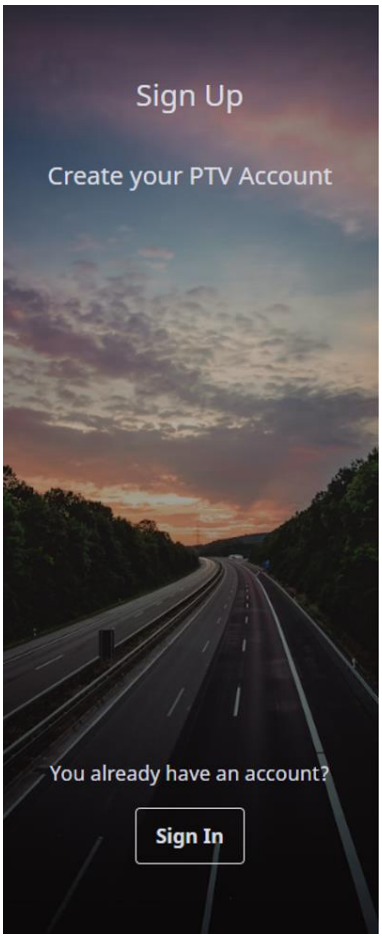
Start free trial: https://developer.myptv.com/en

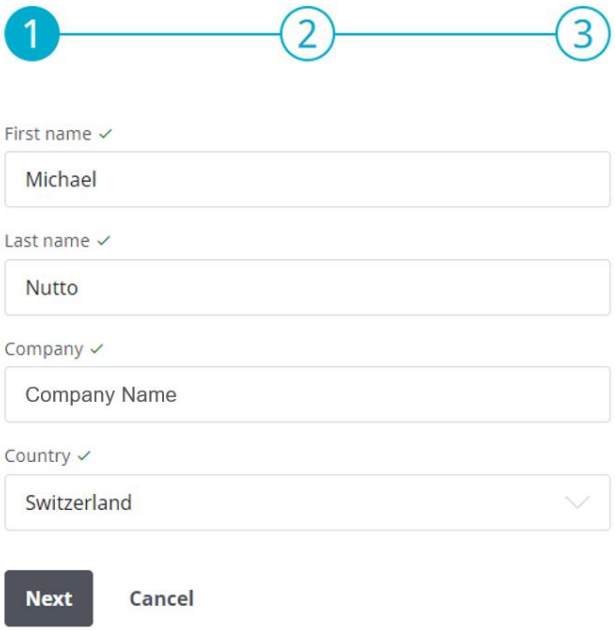
Start free trial

1.1 Registration process

To create a PTV account, follow these three steps. We recommend assigning an administrator to manage the account, as only the administrator account benefits from an unlimited testing period. Note: Free accounts are limited to five hundred transactions per day.

First step: Fill the input fields as follows (exemplarily illustrated):

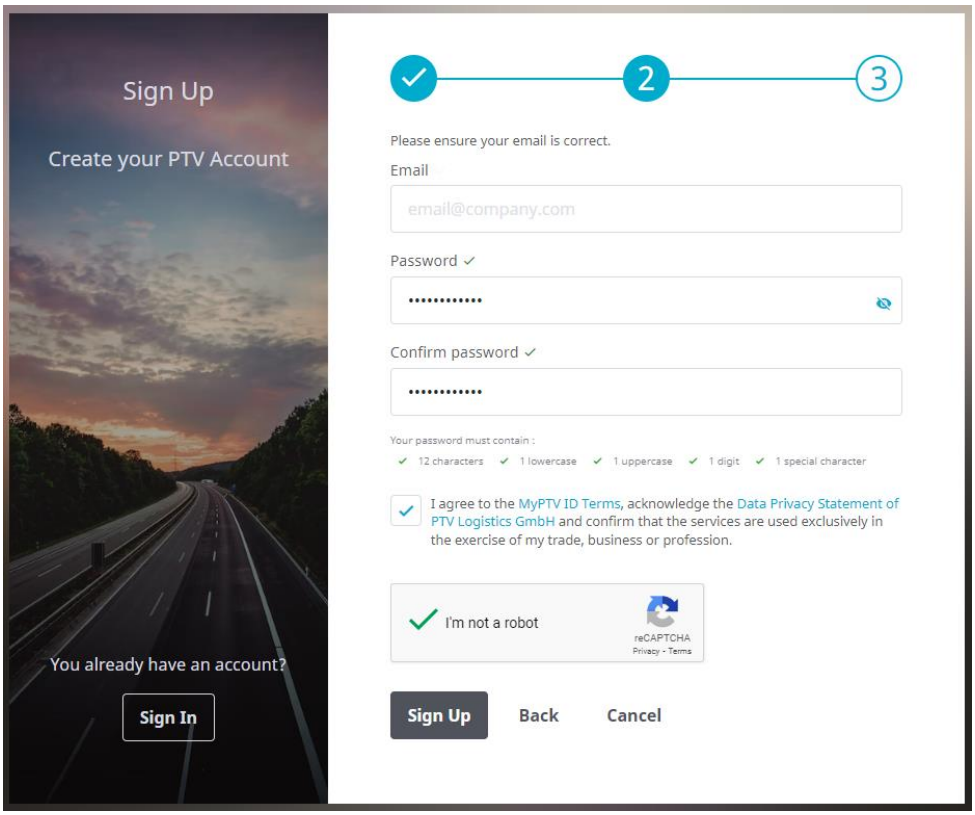




Second step: Add your email address (PTV recommends a group address to ensure a smooth process for administration) and choose a password according to the following rules:

- Minimum twelve characters
- Minimum one lowercase
- Minimum one uppercase
- Minimum one digit
- Minimum one special character

Agree to the MyPTV ID terms, PTV's Data Privacy. Confirm you are not a robot.



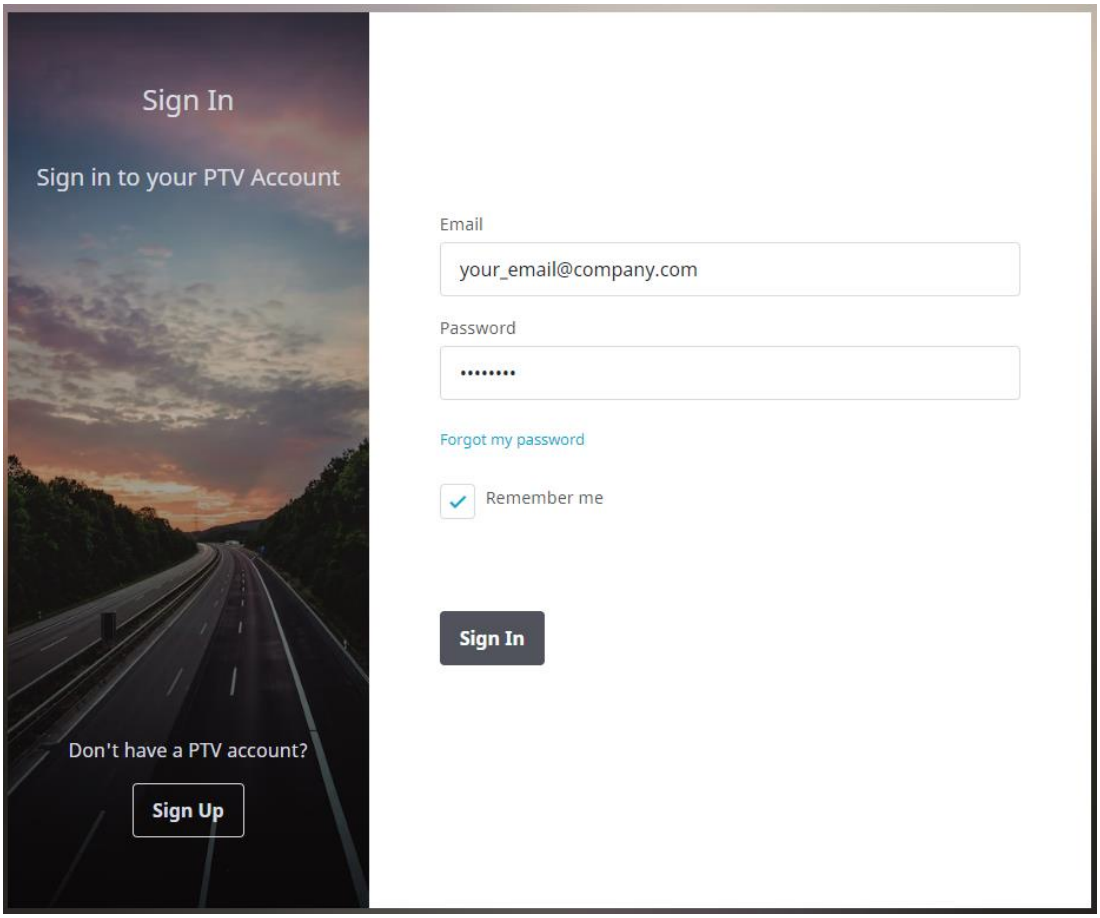
Sign up and follow the instructions in step 3. You will receive an email to finish your registration process.

Third step: Activation is completed through email address verification.

1.2 Sign In

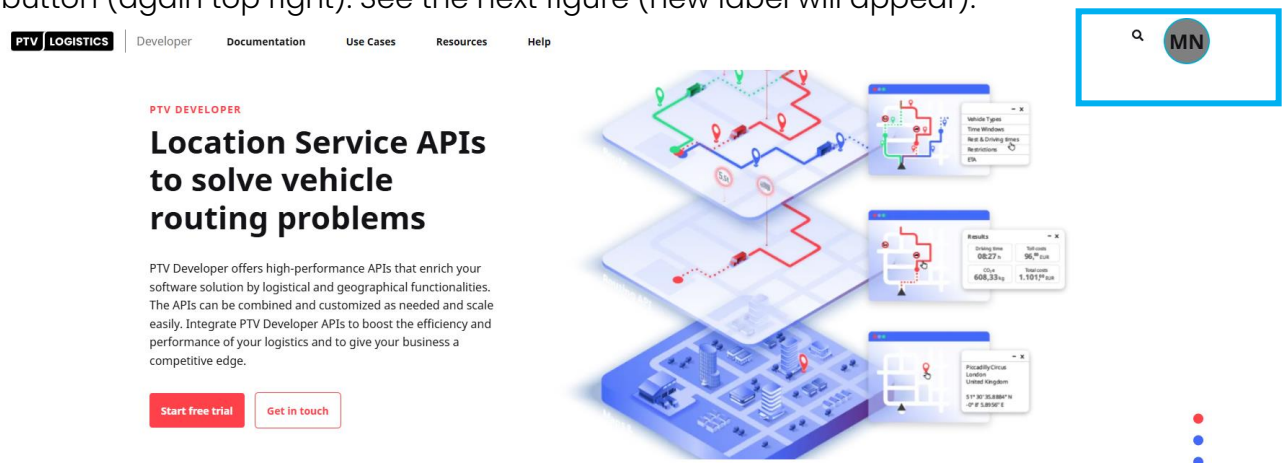
Open the PTV Logistics Homepage: http://developer.myptv.com and use the Login-Button (top right).

The following application will appear to sign in:

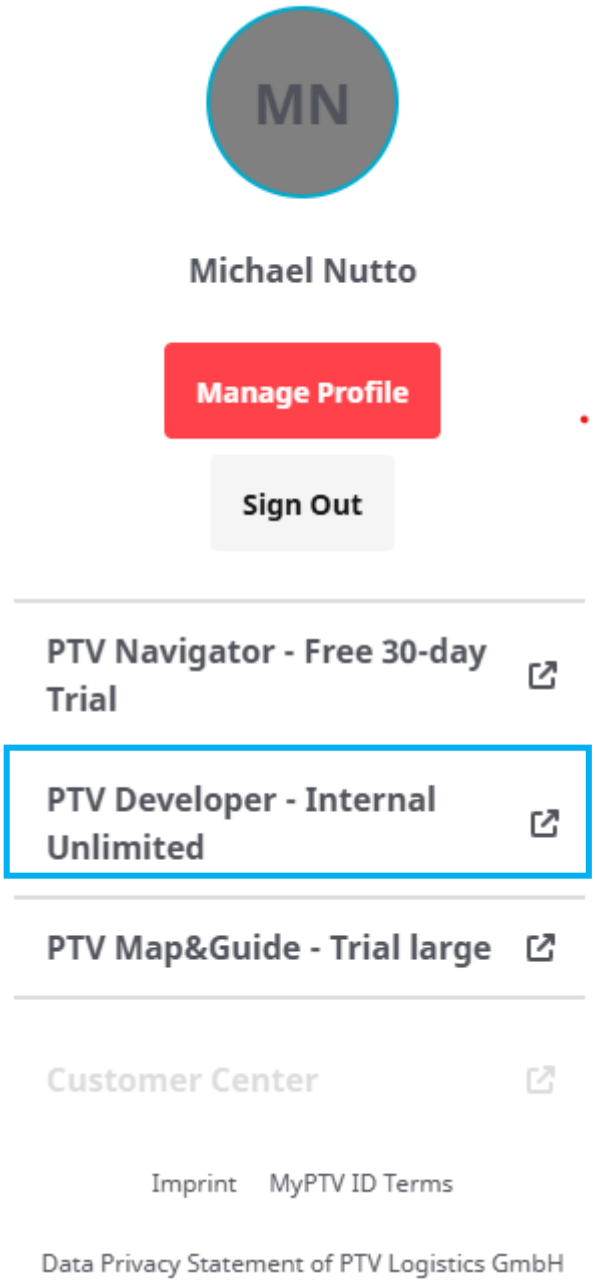


1.3 MyPTV Platform

Get access to the platform by starting PTV Logistics homepage and click on the button (again top right). See the next figure (new label will appear):

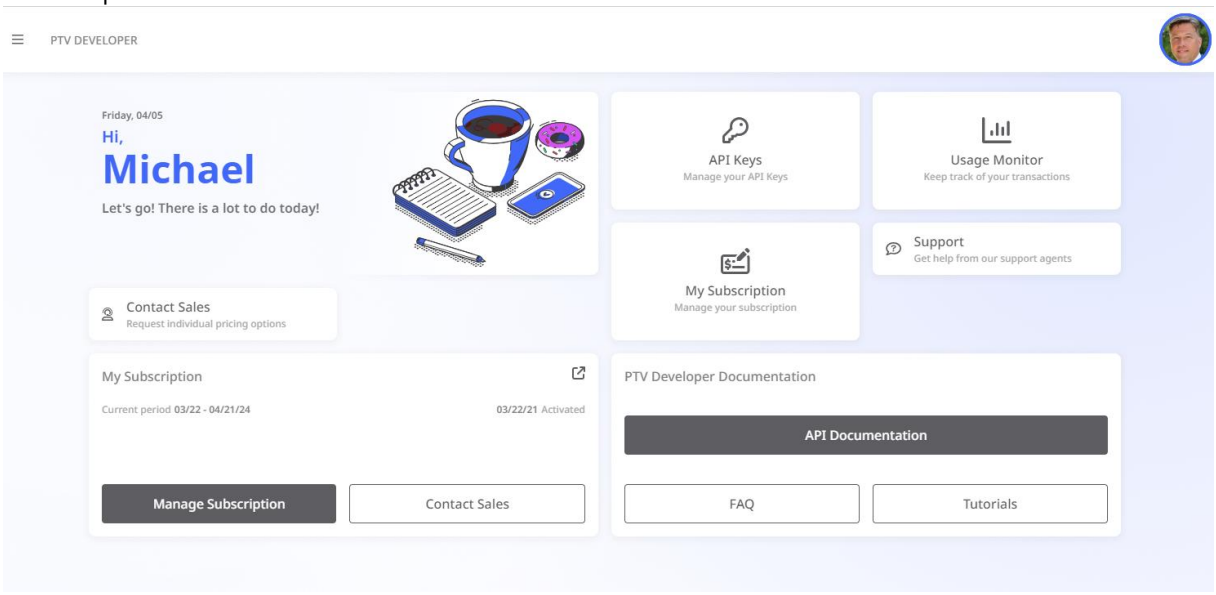


Select your application “PTV Developer – free”.



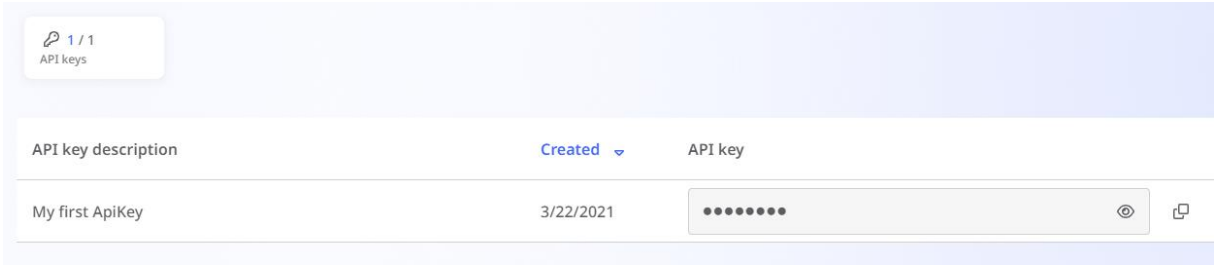
1.4 MyPTV Management Tool

After successful completion, the following window will appear managing PTV Developer Services:



1.4.1 API Key Management

The platform allows you to manage multiple API keys. Free plans are limited to one key, while standard and custom plans support up to twenty-five keys.

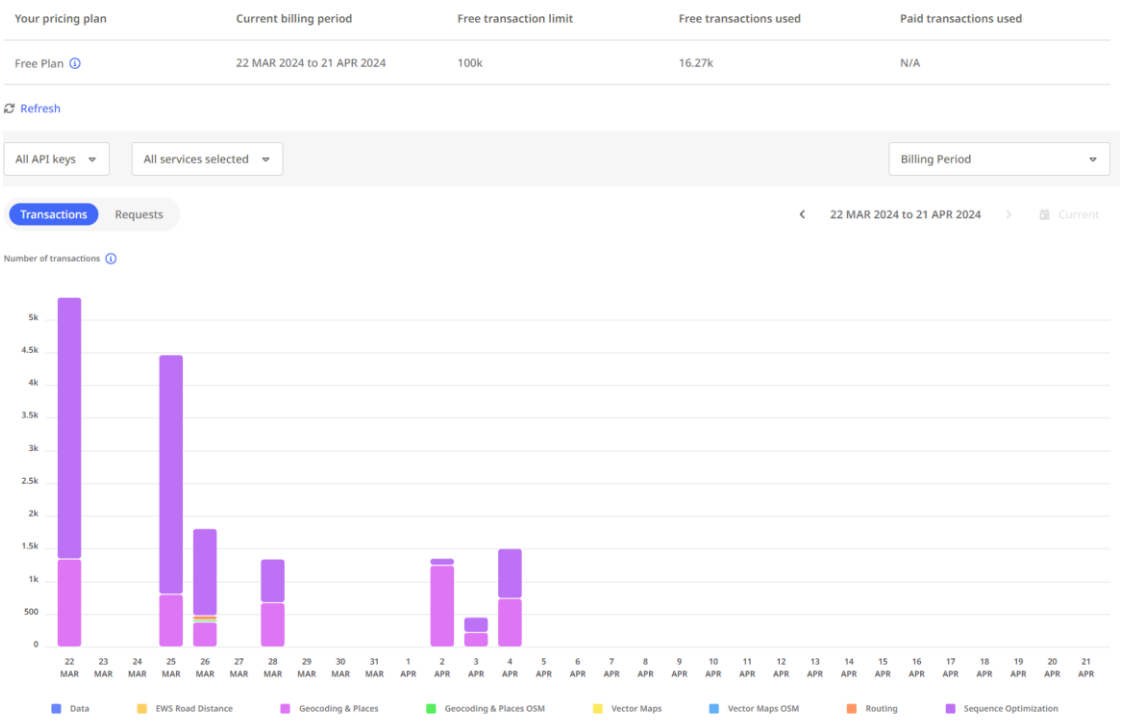


1.4.2 Subscription Management

As already mentioned, subscriptions can be managed via PTV Developer. Selecting the Free Plan with limited functionalities, productive use is excluded, and transactions are limited to five hundred per day. It is recommended you ask for the Developer Kit Plan extending the transaction to maximum 300,000. Please contact your sales representative for detailed information.

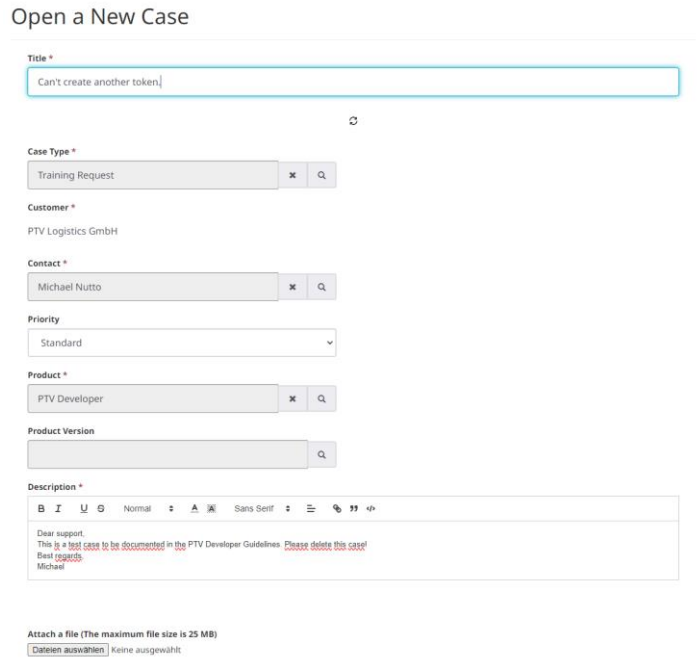
1.4.3 Usage Monitoring

Using the Usage Monitor, customers will be enabled to keep track of their transactions respectively consumed requests divided into the services used.

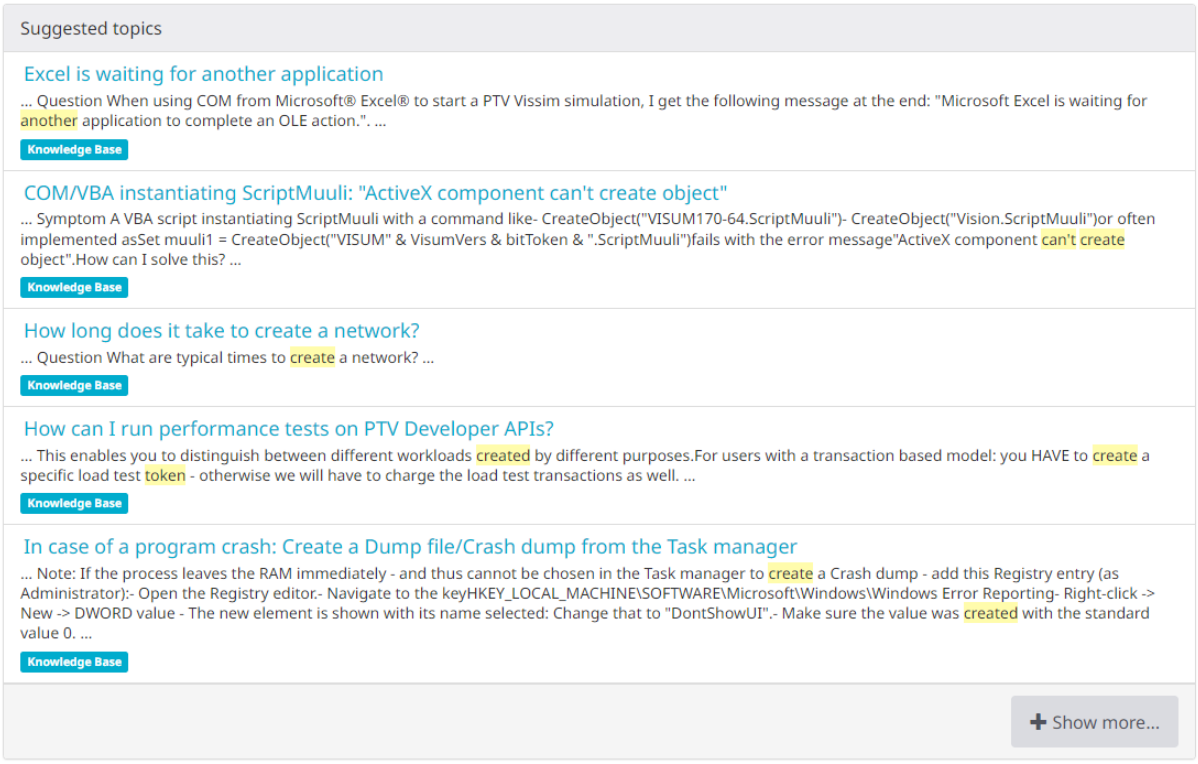


1.4.4 Support Case

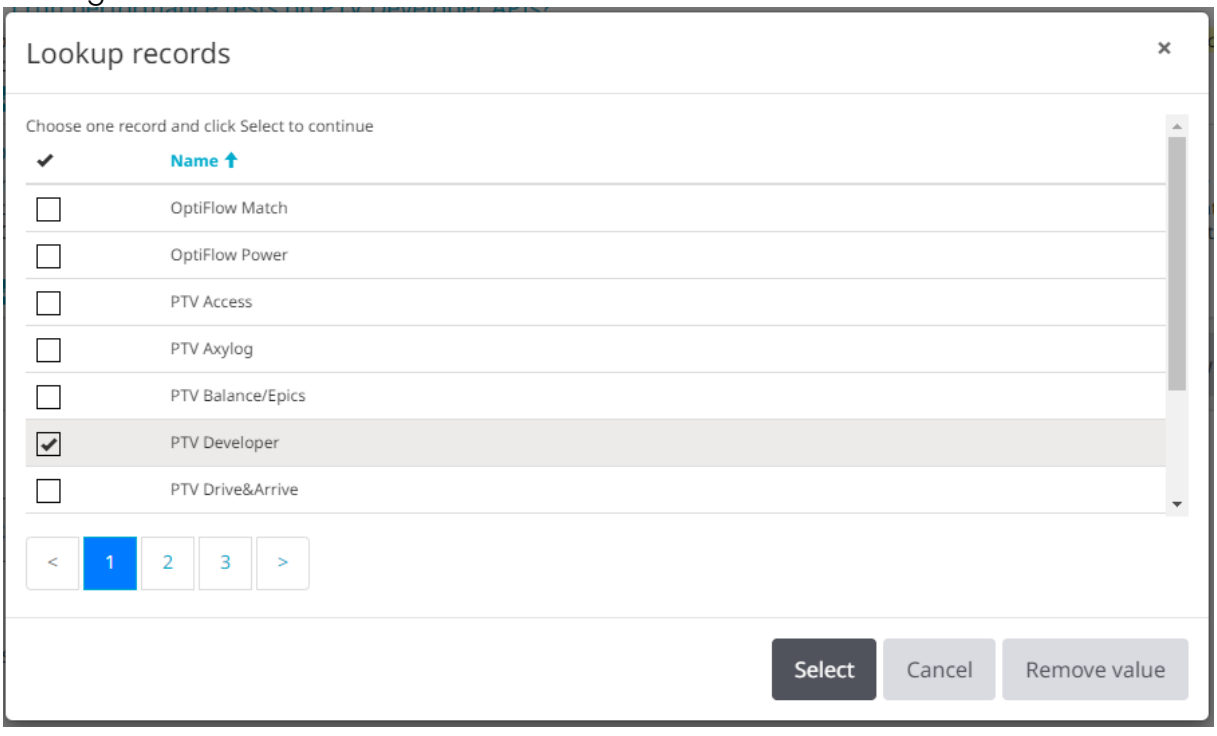
A new case can be created and managed by the platform. Open a new case and follow the instructions. Fill in the required text respectively select boxes as shown in the next figures.



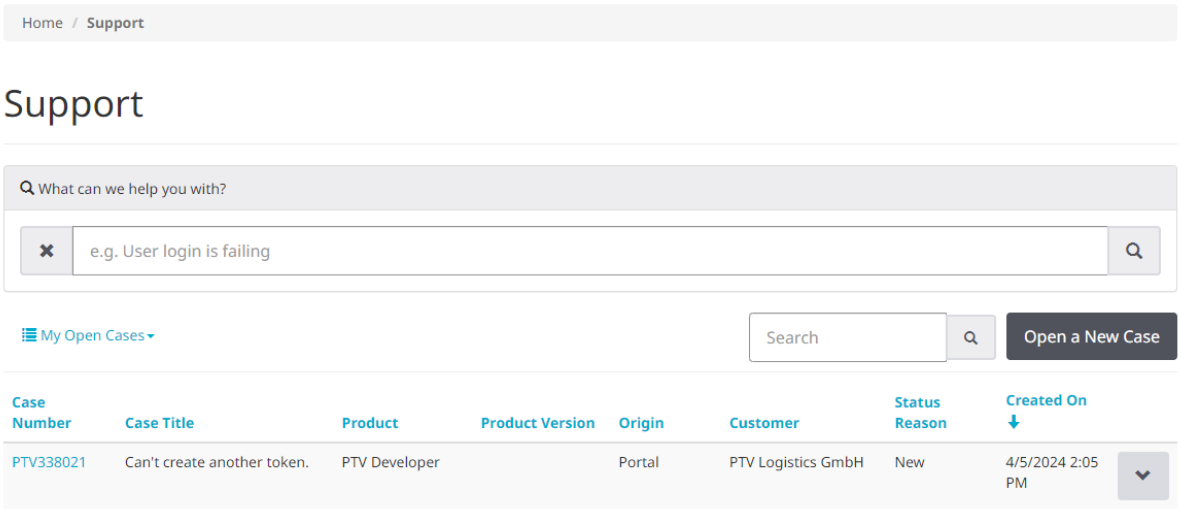
Please note that “suggested topics” will be presented according to the given title to accelerate the support process.



A knowledge base is already in operation. Please check the content before submitting the case.



Please select PTV Developer. After submission, a PTV specific case number will be generated to keep track of the support case.



Please refer to the unique case number to address further questions or comments.

1.4.5 Knowledge Base

PTV Logistics offers a knowledge base for content regarding functional and non-functional topics. Please use the following link:
https://support.ptvlogistics.com/en-US/knowledgebase/category/?id=CAT-01101

2 PTV Developer API

After successful registration and reception of the token as described in chapter 1, PTV Logistics provides various sections of information to enable customers to start with the first application. Please read carefully the instructions provided in the chapter "General" published in the menu bar "Resources": https://developer.myptv.com/en/resources/tutorials. If you have any questions regarding the PTV Developer APIs, please feel free to contact our support.

2.1 How to use Postman

Please download the current version of Postman under the official website: https://www.postman.com/downloads/. The app will be used to execute requests from PTV Developer APIs for testing purposes. By importing the OpenAPI specifications, customer will be enabled to test the capabilities of our Services.

2.2 OpenAPI Specification of PTV Developer

After creating the workspace and generating the collections (exemplarily shown for the collection "**Route Optimization OptiFlow**") via the imported APIs as described below, client can evaluate the GET- and POST-calls.

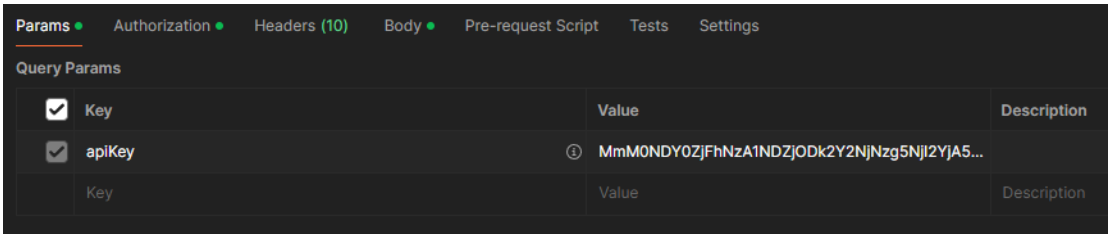
2.2.1 Link to the Open API specification files

- **Accounting:** https://api.myptv.com/meta/services/account/v1/openapi.json
- **Data:** https://api.myptv.com/meta/services/data/v1/openapi.json
- **Geocoding:** https://api.myptv.com/meta/services/geocoding/v1/openapi.json
- **Rastermaps:** https://api.myptv.com/meta/services/rastermaps/v1/openapi.json
- **Vectormaps:** https://api.myptv.com/meta/services/maps/v1/openapi.json
- **Map Matching:** https://api.myptv.com/meta/services/mapmatch/v1/openapi.json
- **Routing:** https://api.myptv.com/meta/services/routing/v1/openapi.json
- **Matrix Routing:** https://api.myptv.com/meta/services/matrixrouting/v1/openapi.json
- **EWS Road Distance:** https://api.myptv.com/meta/services/ews/v1/openapi.json
- **Route Optimization OptiFlow:** https://api.myptv.com/meta/services/routeoptimization/optiflow/v1/openapi.json
- **Load Space Optimization:** https://api.myptv.com/meta/services/binpacking/v1/openapi.json

2.2.2 Get into operation

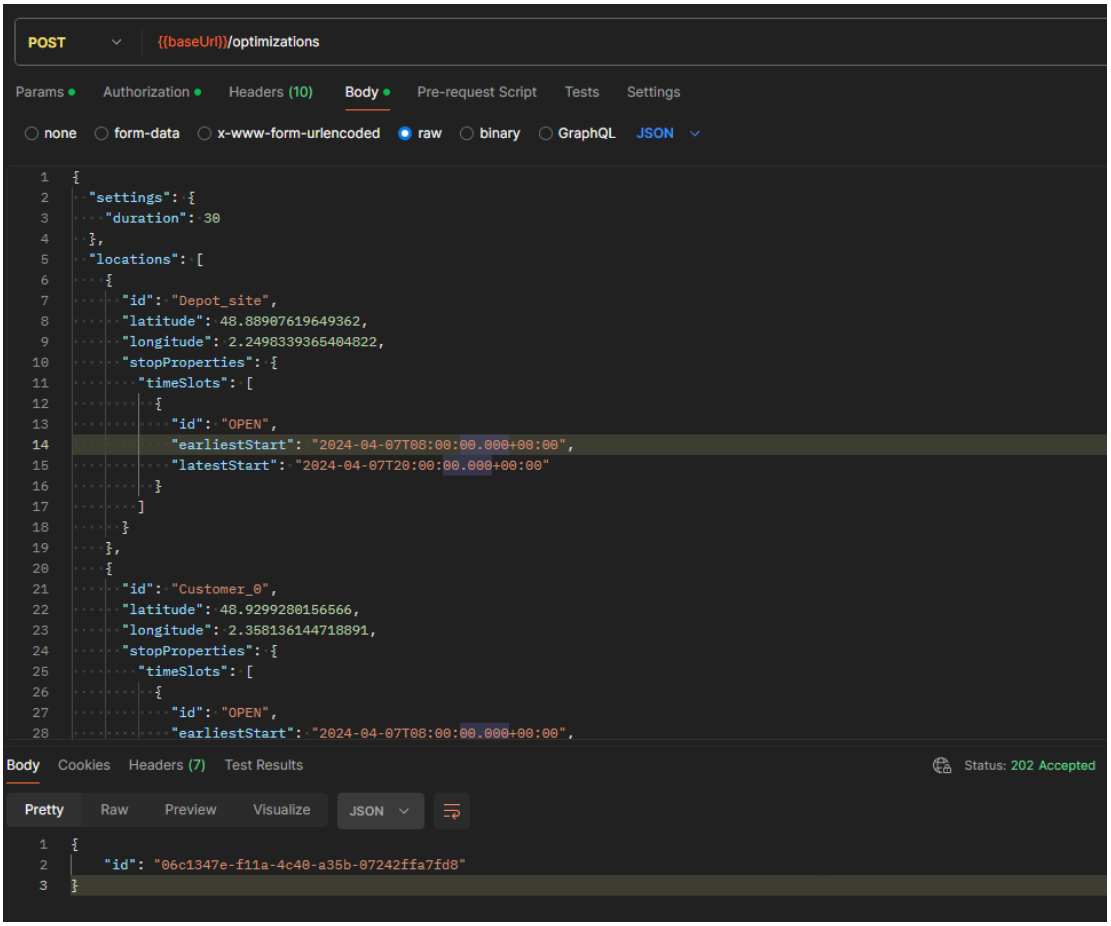
Select collection Route Optimization OptiFlow and execute the code samples presented in the documentation for the Route Optimization OptiFlow API. Link:
https://developer.myptv.com/en/documentation/route-optimization-optiflow-api/code-samples/route-optimization-optiflow-api-depot

1. Check Query Params. Value should be filled with a valid API Key.

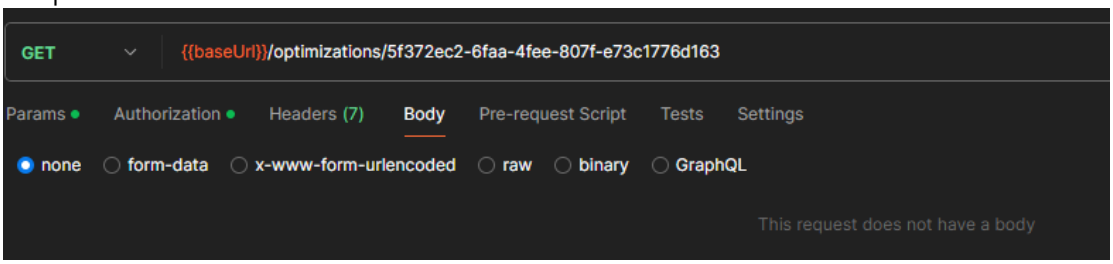


If not, please re-configure your keys and values in the menu bar Authorization.

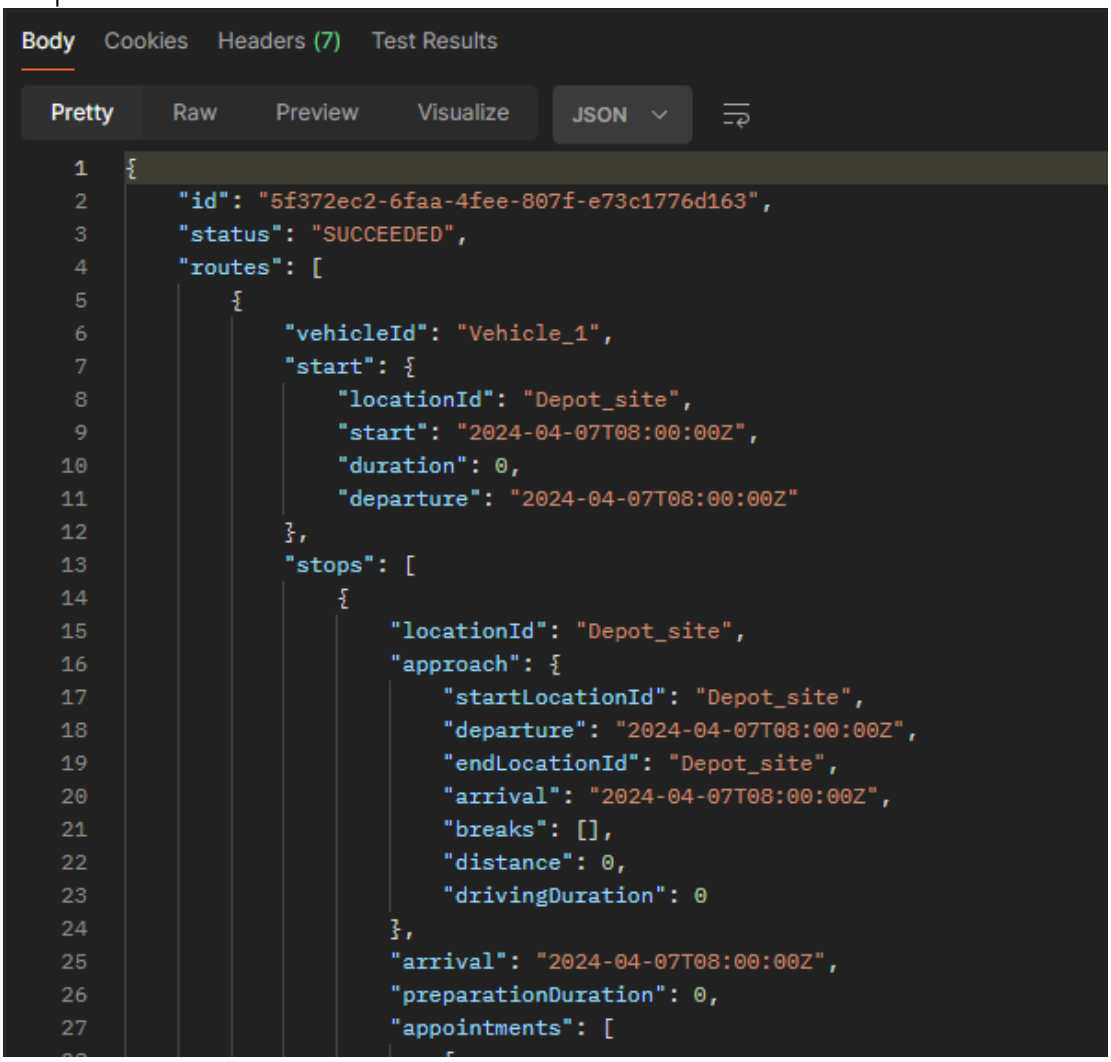
2. Both keys “Content-Type” and “Accept” should be enabled in the menu bar Headers. Value is accordingly defined as: application/json. Copy the code sample into the body and send the request. If everything works as instructed, you should receive a unique identifier (UUID).



3. Get the response by inserting the UUID in the GET call.
Request



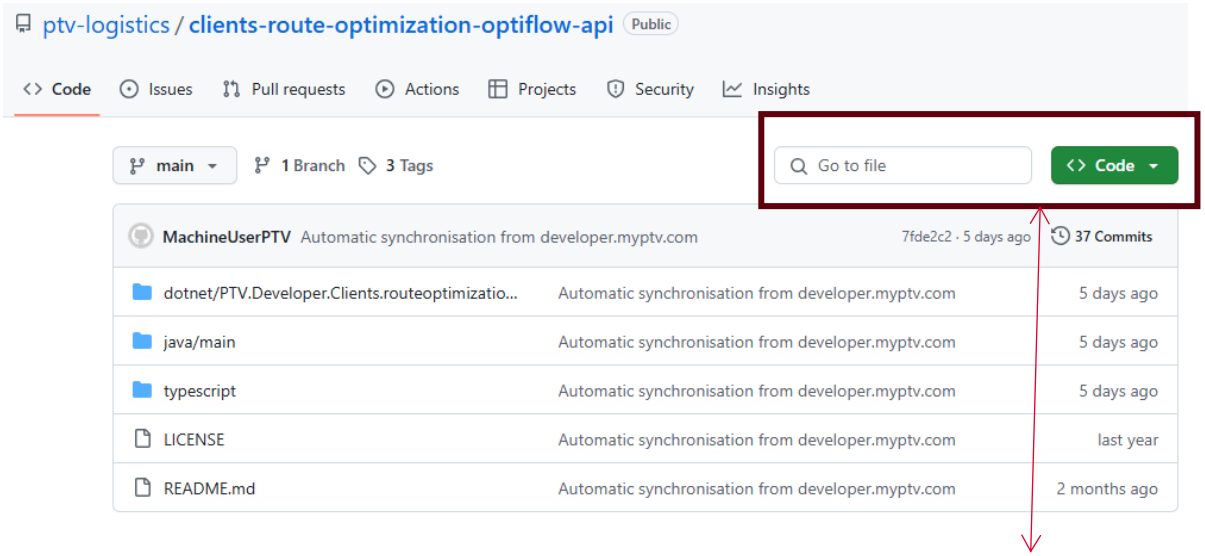
Response



2.3 How to Generate Clients

Depending on the programming language (C# or Java), PTV Logistics recommends downloading the specific clients on GitHub:
https://github.com/PTV-Logistics

Download the required API (exemplarily shown via the Route Optimization OptiFlow API) by using the following link:
https://github.com/PTV-Logistics/clients-route-optimization-optiflow-api



Please note that PTV Logistics provides API clients for Java, C# and TypeScript.

2.3.1 Initializing your client (Java)

To initialize PTV Developer clients, the following code snippet could be used:

```
ApiClient apiClient = Configuration.getDefaultApiClient()
    .setRequestInterceptor(builder -> builder.setHeader("ApiKey", "Enter your ApiKey here"));
PlansApi plansApi = new PlansApi(apiClient);
```

Replace hereby "**PlansApi**" with the designated service.
By executing the request, a valid response should be returned presenting the current API Version of the service used. Exemplarily shown via a Java call:

```
URI uri = new URI(openapiJson).parseServerAuthority();
URL url = uri.toURL();

ObjectMapper mapper = new ObjectMapper();
Map<String, Object> map = mapper.readValue(url, Map.class);

Map<String, String> mapInfo = (Map<String, String>) map.get("info");
String apiVersion = mapInfo.get("version");

System.out.println("Show current version: " + apiVersion);
```

2.3.2 Performance issue

Before the client starts a route optimization request with several thousand orders (request limits are defined by 50,000 orders, 50,000 locations and 1,000 vehicles), the data should be compressed exemplarily as follows (e.g. Java):

```
set localVarRequestBuilder.header("Content-Encoding", "gzip");
```

and compress the body with:

```
byte[] localVarPostBody = memberVarObjectMapper.writeValueAsBytes(optimizationRequest);

String requested = new String(localVarPostBody, StandardCharsets.UTF_8);
ByteArrayOutputStream baOutStream = new ByteArrayOutputStream();
GZIPOutputStream gzipOutStream = new GZIPOutputStream(baOutStream);
gzipOutStream.write(localVarPostBody, 0, localVarPostBody.length);
gzipOutStream.close();

byte[] localVarPostBody_1 = baOutStream.toByteArray();
```

2.3.3 Validate dependencies

Various dependencies for different programming languages may occur during the client generation process. Especially by using JAVA (current SDK) and Type-Script dependencies must be added to the development project.
Detailed information can be found in the README part of the clients presented in GitHub.

2.4 Asynchronous Requests

Avoiding performance issues due to long running requests, PTV Developer provides asynchronous calls especially for the optimization services. The concept can be described in three steps:

1. Start and create services and check the progress (status report) within a reasonable period (create an async function to poll the status)
2. If the status is succeeded, the client will receive the response via the returned UUID (get services)
3. Stop or delete services via the UUID

Detailed information:
https://developer.myptv.com/en/resources/tutorials/general/how-use-asynchronous-requests-ptv-developer-apis

2.5 Testing with MS Excel

For light services like geocoding and routing, an easy approach can be realized to perform bulk requests for validating addresses or calculating distances, durations and costs.
PTV Logistics recommends using Microsoft Excel, which consumes web-services by adding the URL and client's API key. Download the example on our website:
https://developer.myptv.com/en/resources/tutorials/general/how-use-ptv-developer-microsoft-excel

Please note that the cell size is limited to 32.767 characters.

2.6 PTV Developer Blog

Be part of our community and receive the news on our platform "PTV Developer Blog":

https://developer-blog.ptvlogistics.com/

PTV Logistics recommends subscribing the PTV Developer Blog presenting new functions, improvements, updates and relevant background information.

Since February, the new Route Optimization OptiFlow API has been available to optimize routes based on various costs. See the following blog post:
https://developer-blog.ptvlogistics.com/2024/02/14/route-optimization-optiflow-api-is-now-available/

3 Tutorials

PTV Logistics recommends starting with the code samples provided in the tutorials: https://developer.myptv.com/en/resources/tutorials. This tutorial helps developers to get easy access to the interfaces providing the source code and necessary knowledge how to use PTV Developer for geocoding, rendering, map matching, routing and optimizations. Please note taking care of the prerequisites and the guidelines defined for each sample.

3.1 Geocoding

Interactive address search by using the search control: https://developer.myptv.com/en/resources/tutorials/geocoding/interactive-address-search-preview.

3.2 Map Matching

Positioning Matching: The code sample illustrates the map matching by specifying the position and returning the matching results, e.g. the address near the road.

Track Matching: The code sample illustrates the track matching workflow by selecting pre-defined tracks. Toll costs can be included.

3.3 Map Rendering

Vector and Raster Map Rendering by using different frameworks respectively JS libraries such as React, OpenLayer or Leaflet.

3.4 Matrix Calculation

Matrix Calculation with and without time dependencies. It results in distance, travel time and toll cost matrix (if activated).

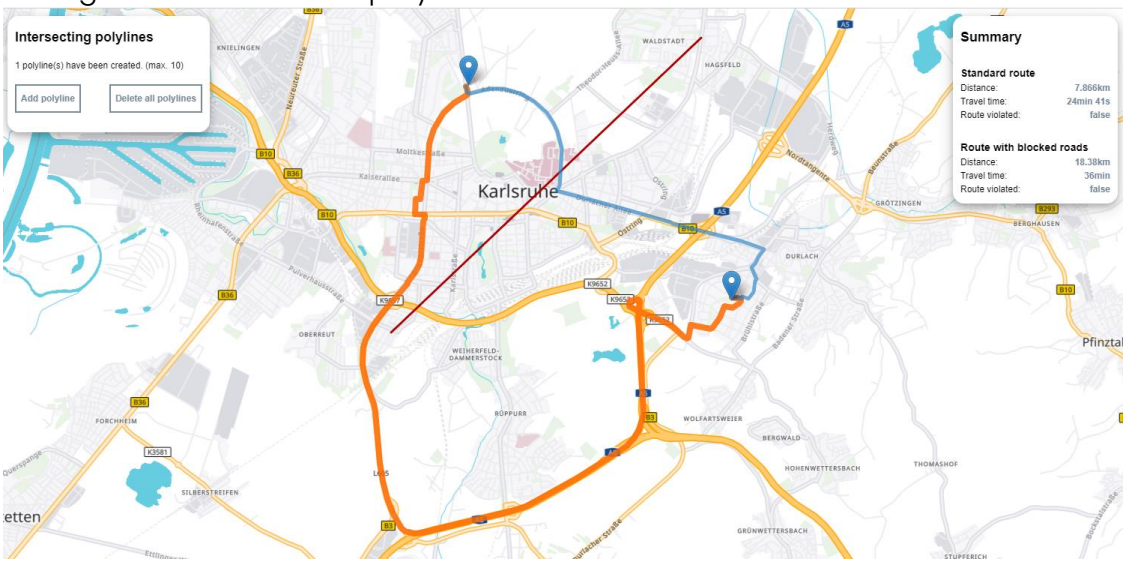
3.5 Routing

A-B Routing: Enter addresses and receive travel time and distance of the calculated route

Toll Calculation: Toll costs will be calculated by clicking on the map. At least two stops must be defined. The following options can be added: Vehicle profile, specific vehicle characteristics, currencies and the required output.

Emissions Routing: Emissions will be calculated by clicking on the map. At least two stops must be defined. The following options can be added: Required emission specification, vehicle profile and specific characteristics for the vehicle. The emissions according to the selected norm will be returned.

Block Intersecting Roads: An easy approach blocking specific road segments or areas to prohibit crossing these streets. Difference between default and changed route will be displayed.



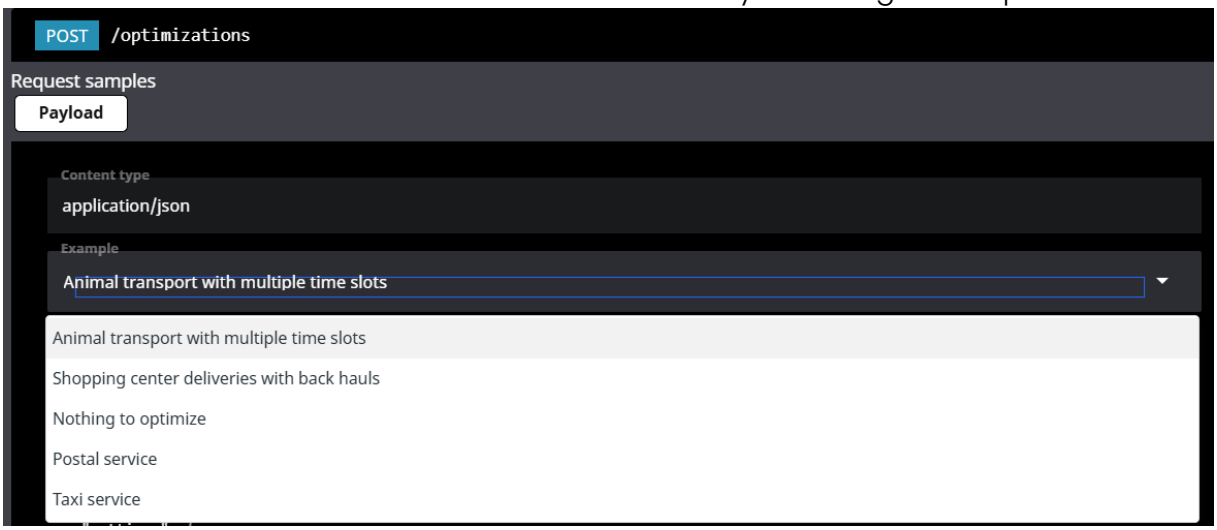
Interactive Routing App with React: Define a route by clicking on the vector map. Select vehicle profile at first and determine distance and travel time afterwards.

Reachable areas: The code sample illustrates two areas around a given position. The size depends on the selected profile, the horizon type and the driving direction.

Interactive Routing App with React: The goal is to build a simple routing by selecting the vehicle profile and setting waypoints which results in distance and travel time outcome. React helps to build a simple web application.

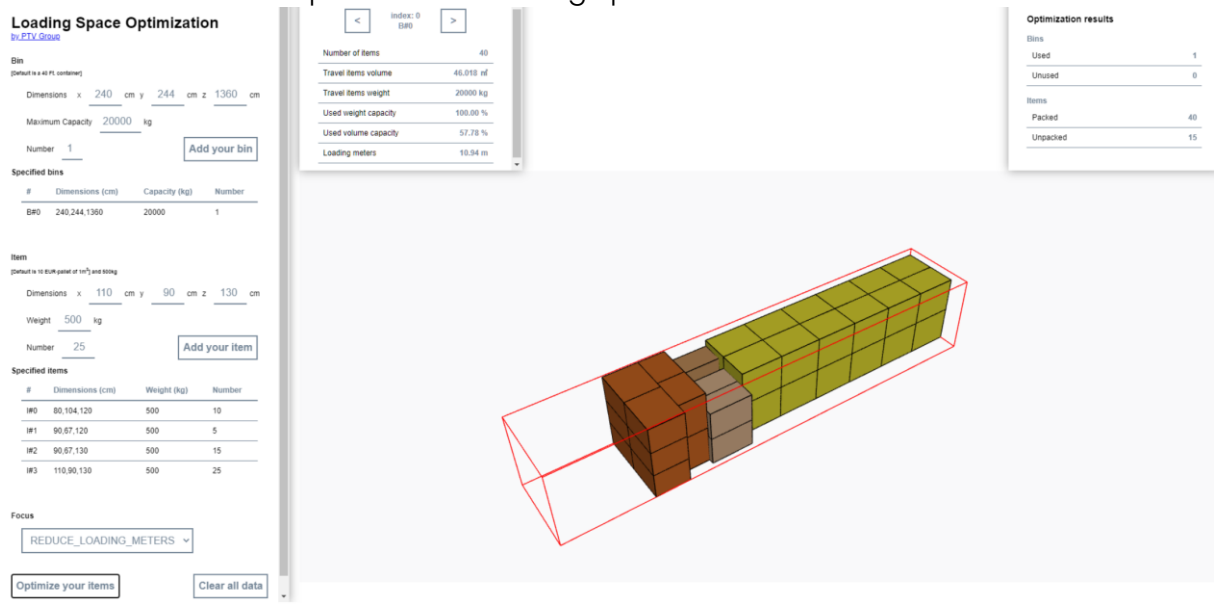
3.6 Route Optimization OptiFlow

The Route Optimization OptiFlow API provides code samples for **Depot-based or Pickup-Delivery orders**. The code samples explain how to create and optimize a route consisting of depot-based or pickup-delivery orders.
Various examples including the corresponding responses are provided in chapter **API Reference.** Select the use case automatically receiving the response:



3.7 Loading Space Optimization

Bin packing: The codes sample illustrates how to optimize the loading space of a container by defining dimensions and weights of your bins and items. Additionally, consider various constraints such as allowed stacking, tilting or rotation. Select method and optimize the loading space.



4 Onboarding

The PTV Developer platform offers a comprehensive suite of cloud-based API services designed to support and optimize transport logistics.

4.1 Getting familiar with the APIs

These services include tailored services such as **geocoding**, **map rendering**, **map matching**, **routing**, and **route optimization,** each playing a critical role in enabling efficient, data-driven logistics operations.

By integrating these APIs into your digital infrastructure, you gain access to powerful tools that streamline complex logistics processes. Whether you are building applications for fleet management, delivery planning, or real-time navigation, PTV Developer APIs provide:

- **Seamless integration** into existing systems through standardized interfaces.
- **Customizable solutions** tailored to specific operational requirements.
- **Automation and digitalization** of workflows for faster, more reliable decision-making.
- **Cost savings** through optimized routing and resource allocation.
- **Greater flexibility** and responsiveness to dynamic changes in logistics environments.
- **Enhanced competitiveness** by enabling smarter, faster, and more adaptive logistics strategies.

This chapter will give an overview of the core functionalities of each API.

4.1.1 Account API

With the Account service, you can manage your API keys and track their usage. The Account API provides the possibility to create, update and delete API keys with the typical CRUD operations:

Name	Description
createApiKey	Create a new API key
getApiKeys	Get all API keys of the current subscription
updateApiKey	Update the description of an API key
deleteApiKey	Delete an API key
getUsage	Retrieve the usage for the given period

4.1.1.1 Good to know

Using the Account API, you need first to create a master API key. After registration, the administration tool is available.

4.1.1.2 Tracking Usage

Usage can be requested for any time period with a limit of 60 days or for the current billing period. The response is hierarchically structured:

- Level 1: List of all API keys including deleted API keys and if available the master API key
- Level 2: List of all services (per API key)
- Level 3: List of all days with the charged transactions and the number of sent requests (per API key and per service)

4.1.2 Geocoding API

Geocoding enables you to receive exact positions or to find the corresponding addresses by entering geo-coordinates (decimal latitude and longitude).

The following methods are available:

Name	Description
searchLocationsByText	Single-field search by adding free-form texts
searchLocationsByAddress	Multi-field search by entering specified fields (country, state, locality, postal code, street and house-number)
getSuggestionByAddress	Receive suggestions for the given address added as a multi-field input
getSuggestionByText	Receive suggestions for the given address added as a free-form text input
searchLocationsByPosition	Set the position (lat,lon) and receive the adjacent locations
searchPlacesByText	Search for places based on a single-field text input
searchPlacesByPosition	Search for places near a given geographical position
searchPlacesByArea	Searches for places within a requested area
getPlaceCategories	Returns the list of place categories

4.1.2.1 Good to know

The score describes how well a result location and the request match. This is very useful to validate the quality of the address search.

4.1.2.2 Batch Geocoding

Batch processing helps you to send a bulk of addresses in one asynchronous API call. After successful execution, a unique id returns which will be used to request the status or to receive the locations.

4.1.3 Vector Maps API

The Vector Maps API allows you to render geographical vector tiles with easily integrable open-source frameworks. The service uses the Web Mercator projection based on WGS84.
The following methods are available:

Name	Description
getVectorTile	Returning map tiles based on zoom level and the given tile index
getTile	Returning map tiles based on zoom level, the given tile index and the specified layers to be displayed. Supported layers are truckRestrictions, trafficPatterns, lowEmissionZones and toll

4.1.3.1 Good to know

The Vector Maps service provides highly configurable map styles. You have the choice of using one of our predefined styles or building your own.
Various data layers such as traffic patterns, toll roads or truck restrictions can be displayed activating the corresponding layers. See Showcase.

4.1.3.2 Open Street Map (OSM)

Vector Maps can also be provided via OSM. Just use Vector Maps OSM API

4.1.3.3 Raster Maps

If you want to keep the 2D raster map tiles, use the Raster Maps API.

4.1.4 Map Matching API

The PTV Developer Map Matching API allows you to track and map your fleet and vehicles' movements.
The following methods are available:

Name	Description
matchPosition	Matches a single unrelated position on a map
createMatchedTrack	Creates a complete track by triggering a matching calculation for it
getMatchedTrack	Gets the results of a matching calculation specified by its ID
deleteMatchedTrack	Deletes a matched track specified by its ID

4.1.4.1 Good to know

Three modes are available to define the calculation behavior: performance, quality and standard.

4.1.4.2 Query Parameters

The service provides specific query parameters to tailor the output:

- Heading: The direction of travel defined by a clockwise angle
- calculationMode: This affects the quality and performance of the matching
- results: Selects which properties the MatchePosition will contain
- language: The preferred language for the response

4.1.5 EWS Road Distance API

The PTV Developer EWS Road Distance API offers an industry standard distance matrix as a common basis for tender management, invoice auditing and credit note procedures.
The following methods are available:

Name	Description
getRelation	Gets the truck distance and additional information for a relation between two locations
getSuggestions	Gets suggestions for a location. Provide at least two characters in **postalCode** or **locality** to obtain results

4.1.5.1 Good to know

The PTV Developer EWS Road Distance API offers:

- Standard distance matrix for trucks:
  EWS road distances are calculated using up-to-date road network data including truck attributes like physical or legal restrictions. In addition, a dedicated truck routing profile is used to calculate realistic distances for truck transports.
- Distance calculation for more than 550,000 places in Europe:
  For EWS we use location information on more than 550,000 places in Europe, including most recent administrative codes and postcode information.
- Toll distances for Germany and Austria:
  As toll costs are an important factor for transport costs, truck toll distance information for Germany and Austria are available as well.
- Easy integration into all systems – via API:
  Gain access to the comprehensive EWS database using our PTV Developer API. Especially modern cloud solutions like ERP etc. benefit from our Data as a Service (DaaS) portfolio.

4.1.5.2 Query Parameters

Enhancing the call, select the following optional attributes besides defining the start and destination to determine the relation between these locations:

- region: Searching the reference locations in the given region
- dataVersion: According to contractual agreements, select the year to determine total and toll distances
- results: Comma-separated list that defines which results besides the distance should be returned.

4.1.6 Routing API

The Routing API is designed to calculate precise routes and accurate arrival times especially when taking truck relevant constraints into account. Speaking of vehicles, predefined profiles are provided for the most common vehicle types such as trucks 40t, 11.99t or 7.49t.
Many requirements and challenges need to be considered when optimizing routes for commercial vehicles:

- Vehicle-specific parameters such as size, weight, legal restrictions, toll costs or time constraints which will override the default parameters of the predefined profiles.
- Routing-relevant options such as general truck routes, traffic and routing modes, geographic restrictions, avoidance of specific road types or speed factors influencing the route direction and travel time.
- Location-specific attributes to access specific positions or to manipulate the route at certain waypoints.

The following methods are provided for route calculations:

Name	Description
calculateRoute	Calculate a route by adding a list of waypoints. Minimum pair is the start and the destination. Typically returning distance, travel time, toll costs and emissions in total or per leg. If activated, the polyline will also be added to the result list for rendering purposes
calculateRoutePost	Additionally considering opening intervals and working hours. Please note this is a http Post call
getRouteByRouteId	Returns the route of a previously calculated route
getEstimatedTimeOfArrival	Calculates the estimated time of arrival (ETA) based on a previously calculated route and the position of the vehicle

4.1.6.1 Good to know

Besides distance and driving time determination, the following values can also be calculated:

- **Toll costs** for most countries in Europe, the United States of America, Australia and New Zealand
- **Emissions** based EN16258, FRENCH_CO2E_DECREE_2017_639 and ISO14083 (2022,2023)

Specific truck routes are provided to be activated in the route calculation as follows:

- **Preferred routes** such as for long trucks in Germany and Netherlands
- **Public roads** in Sweden
- **General truck routes** for transit or must be used according to legal regulations.
- **Routes** that belong to the highway network as defined by the Surface Transportation Assistance Act in the US
- **Routes** for vehicle combinations as defined by the Australian Transport Administration

The result list consists of many attributes which can be returned as follows:

- **Typically required result attributes**: Route Id, legs, polylines and toll costs including toll sections and systems
- **Events**: Toll, maneuver, border, violation, violation polylines, waypoint, UTC offset change, combined transport, traffic, traffic polylines, low emission zone, delivery only, delivery only polylines, schedule, schedule with driving, EV status, EV status polylines and EV charge
- **Emissions**: EN16258 (2012, HBEFA), ISO14083 (2022, 2022 default consumption, 2023, 2023 default consumption, FRENCH_CO2E_DECREE_2017_639
- **Specific selected result attributes**: Alternative routes, schedule report, guided navigation, monetary costs and EV report

4.1.6.2 Bulk routing via matrix calculations

The Matrix Routing API allows you to perform bulk routes. See chapter 4.1.8 for more information.

4.1.6.3 Reachable areas and locations

Using the following GET and POST requests to calculate areas reachable from a waypoint and within given horizons:

Name	Description
calculateReachableAreas	Calculates the areas which can be reached from a waypoint, within given horizons (limited to 25 km or 20 minutes)
startAndCreateReachableAreas	Starts and creates the areas which can be reached from a waypoint or from a route, within given horizons. This type of request is recommended when the **horizons** parameter is more than 20 minutes or 25 km
getReachableAreas	Gets the results of a reachable areas calculation specified by its ID
deleteReachableAreas	Cancels a reachable areas calculation and deletes the calculated results specified by its ID
startAndCreateReachableLocations	Starts the calculation of the sets of reachable and unreachable locations from the given ones and creates them as the result
getReachableLocations	Gets the results of a reachable locations calculation specified by its ID
deleteReachableLocations	Cancels a reachable locations calculation and deletes the calculated results specified by its ID

4.1.7 Data API

With the Data service you can obtain additional data as follows:

Get map information:
Receiving a response about the integrated low emission zones and detailed information about the geographical units including code, country, state, continent and specific features about toll and the toll systems.

Get combined transports:
Adding the query parameters either position or text. A list of selected combined transports will be returned with additional information such as duration, type or list of vehicles which are allowed to use the combined transport mode.

Get predefined vehicle profiles:
Providing a list of reference profiles as response. The list contains information about name, description, region, vehicle parameters, currency and values of monetary cost options.

Get roads from points on the map:
Returning a list of selected roads according to the given points.

Get vehicle models:
Returns a list of model-based vehicles containing the predefined profile, vehicle type, physical and legal properties of the engine and specific characteristics of the battery.

Get API changes:
Responding API changes such as type, date, description, API, version and link to the corresponding website.

4.1.7.1 Custom Road Attributes

The Data API also provides methods to manipulate road attributes as follows:

Name	Description
createCustomRoadAttributeScenario	Creates a scenario to edit road segments
getAllCustomRoadAttributeScenarios	Get a list of all scenarios the user has created
getCustomRoadAttributeScenario	Returns a scenario specified by its ID
updateCustomRoadAttributeScenario	Perform a full update of an existing custom road attribute scenario
deleteCustomRoadAttributeScenario	Delete a custom road attribute scenario

4.1.7.2 Good to know

Map information: The map covers the whole world and is divided into geographical units which are usually countries. For some countries such as the United States, Canada, Australia and India the geographical units are federal states.

Toll: Focus on providing toll information related to trucks if available. For other vehicle types, it cannot be guaranteed that toll can be calculated.

Vehicle profiles: The properties of a vehicle are stored in a profile, matching typical transport situations. It describes the physical and legal properties, such as height and weight, which are used to calculate a route. Further use cases are toll and emission calculation, which are also highly vehicle dependent.

Custom Road Attributes: With custom road attributes you have the possibility to modify attributes of roads and combined transports in the map data to your individual needs (e.g. to consider short-term roadworks on a daily used road and currently not available in the map data). Please note that such changes are relevant for routing and apply to all vehicle profiles. Therefore two steps are necessary: First create a scenario for one or more selected roads or combined transports with the Data API and second use this scenario in the Routing API.

It is also possible to allow/prohibit them, reduce the driving speed and modify vehicle dimensions like weight or height. Moreover, you can prohibit combined transports. It is planned to add additional possibilities at a later time.

4.1.8 Matrix Routing API

The PTV Developer Matrix Routing API allows you to calculate distances and travel times between multiple locations quickly and accurately. You can retrieve accurate results for different predefined vehicle configurations, to automatically respect road limitations like truck attributes depending on your vehicle characteristics.
The service provides the following methods:

Name	Description
calculateMatrix	Creates a matrix by calculating distances and travel times between given origins and destinations. Requests will be rejected if the number of relations is greater than 250,000
startMatrixCalculation	Creates a matrix asynchronously by calculating distances, travel times and toll costs between given origins and destinations. The request is limited to four million relations
getStatus	Requests the status of a matrix calculation
getMatrix	Requests the results of a matrix calculation operation
cancelMatrixCalculation	Cancels a matrix calculation and deletes the calculated results specified by its ID
getRouteId	Returns a routeId representing a route with the specified list of locations

4.1.8.1 Good to know

Calculating huge distance matrices, it is recommended that the content is always Gzip-encoded within the request body.
Distance matrix calculation can be performed when using the reference profiles defining a vehicle by a set of default attributes. The standard values can be requested via Data API.

4.1.8.2 Query Parameters

In comparison to the Routing API, a smaller set of parameters are available to be selected for the matrix calculation as follows:

- Currency: Currency of the toll price
- Start time: Defines the start time
- Toll time: Defines the date and time at which to calculate toll prices
- Duration: Defines the duration for the calculation of travel time profiles
- Traffic mode: Defines how to consider traffic in a matrix calculation

- **Detour factor:** Defines the average detour an actual route on the road requires, compared to the direct distance
- **Average speed:** Defines the average speed of the vehicle
- **Result format:** Defines the format for the output such as distances, travel times or toll costs

4.1.8.3 Time-dependent Matrix

Matrices are often calculated based on an average traffic situation. Please note getting more precise values is possible with following parameters: **trafficMode, startTime** and **duration**.

The following use cases are handled, depending on the combination of these options:

- **time-independent matrix:** distances and travel times are based on an average traffic situation
- **point-in-time matrix:** distances and travel times are calculated based on a traffic situation for a certain date/time
- **multiple travel time matrix:** travel time profiles are returned for a period of time, based on the known traffic situation

4.1.9 Route Optimization OptiFlow API

Getting familiar with the API, PTV Logistics recommends reading the concept first. The API defines relevant terms to avoid any misunderstandings when mapping a use case. The intention is that the client should understand the meanings of locations, stops, depots, orders and tasks. How to use constraints, categories or time slots. How to use a vehicle profile to receive the expected output, etc.

4.1.9.1 Introduction

The optimization pursues two target functions:

- Number of scheduled orders to be optimized for pickups and deliveries
- Cost reduction by considering unscheduled orders and the cost of the route (cost per hour, cost per kilometer and fixed costs of the selected vehicle).

Three types of orders are provided by the API:

1. **Pickup orders:** Transporting goods from location to depot.
2. **Delivery orders:** Transporting goods from depot to location or from location to depot.

3. **Pickup-delivery orders:** Transporting goods between location A to B.

Regarding loading goods, dimensions such as weight, volume and pallets will also be considered. The maximum loading size of the vehicle will ensure that only permitted combinations will be calculated.
By using the **outsourcing costs**, orders will be unscheduled when additional costs exceed the outsourcing costs. This becomes relevant prioritizing transport orders.

4.1.9.2 Common definitions

A **task** schedules a pickup or delivery order by adding time slots determining when a task can be scheduled for execution. Please note that an **appointment** aggregates a consecutive series of tasks within a time slot.

A **location** is defined as follows:

1. **Locations** where a vehicle starts or ends (e.g. depot or home location)
2. **Stops** where pickup or delivery will be carried out. Hereby, a stop consists of stopProperties as follows:
    a. Parameter **preparationDuration**: Defines the time to prepare before performing the task
    b. Parameter **timeSlots**: Defines intervals in which tasks can be executed.

The **constraints** consist of various **categories** defining the relation between the entities: Vehicle, order and depot. Loading incompatibilities will also be considered by adding the corresponding constraints.

- Type **orderCategories:** The sequence of order categories must be considered when transporting goods (e.g. fresh, cooling, frozen, …)
- Type **vehicleCategory:** Defining the vehicle categories which contain the restrictions (e.g. cooling compartment)
- Type **loadedOrderCategory:** Orders belonging to the category must be loaded to the assigned vehicle
- Type **forbiddenOrderCategory:** Orders will not be loaded or unloaded if the restriction is still active.
- Type **taskCategories:** The sequence of task categories must be considered when transporting routes

The following combinations are available to appropriately use the existing constraints:

- Enumeration: ORDER_REQUIRES_VEHICLE – an order can only transported by an assigned vehicle
- Enumeration: VEHICLE _REQUIRES_ ORDER – a vehicle can only transport orders if permitted by the order category
- Enumeration: FORBIDDEN_COMBINATION – An order of the specified category cannot be transported by a vehicle with the given category.

4.1.9.3 Good to know

The main goal of the Route Optimization OptiFlow API is to effectively allocate tasks to routes assigned to a specific vehicle by minimizing costs. Important restrictions – like legal, physical or customized ones – are considered to match the real dependencies in transportation.
Before designing your transport as input for route optimization, please consider following rules:

1. Upon arrival at the depot, all orders must be unloaded first. Orders are delivered in the opposite order they were picked up.
2. The LIFO rule applies in general: The last item of the pickup orders will be delivered first. Please note that the combination of pickup and delivery orders might not adhere to the principle.
3. For an A-B transportation the LIFO principle dictates that the most recent pickup-up item must be delivered before previous items.

4.1.9.4 Object definitions

The body of the request is structured according to OpenAPI version 3 as follows:

- Object settings: Define the maximum duration [s] how long the optimization should run. The maximum time is 24 hours. The parameter duration is obligatory.
- Array of objects named as locations: A maximum list of locations where goods can be picked up or delivered. Please note that id, latitude and longitude are required parameters.
- Object orders: A list of orders that should be scheduled. Please note that by using the objects pickups, delivery and pickupDeliveries, the parameters *id* and *locationId* are mandatory.
- Array of objects named as vehicles: A list of vehicles which can be used for pickup and delivery. The list of vehicles is limited to 2,500 vehicles. Please note that the following parameters are obligatory: *id*, *locationId*, *earliestStartTime*, *latestEndTime*, *profile*, *perHour* and *perKilometer*.

- Array of objects named as **depots**: A list of depots for transportation tasks. The list of depots is limited to fifty items. Please note that *id* and *locationId* are required parameters. The use of these objects is optional.
- Array of objects named as **routes**: A list of routes that should be reconstructed prior to optimization. The list contains a unique identifier, the starting point, a sequence of tasks and a list of breaks scheduled on the route. The use of these objects is optional.
- Object **constraints**: Describes restrictions for orders, vehicles and the specified tasks. The use of this object is optional. The use of this object is optional.
- Object **rules**: A list of conditions and the duration for extra preparation that describe the rules to modify location properties. The use of this object is optional.
- Object **metadata**: The name and a list of user-defined tags will be listed in the result. The use of this object is optional.

4.1.9.5 Optimization – API Reference

The following methods are provided by the route optimization OptiFlow API:

Name	Description
startOptimization	Initiates a cost-minimizing optimization that efficiently schedules orders onto routes. The routes are assigned to the provided vehicles and satisfy the given constraints. Once the optimization is accepted, the optimization starts for the requested duration after which the resulting routes can be retrieved.
getOptimizationResult	Retrieves the current result of the optimization. While the optimization is running, this will return intermediate metrics to track the progress. Once succeeded, the optimized routes are returned.
stopOptimization	Stops a running optimization procedure. Once accepted, the optimization will stop as soon as possible, and the final state of the routes can be retrieved.

4.1.10 Loading Space Optimization API

The PTV Developer Loading Space Optimization API allows you to optimize cargo loadings and make drop-off and servicing times at stops efficient by minimizing free volumes and considering unloading sequence.

4.1.10.1 Good to know

The service provides an optimization based on cuboids to be packed. Important conditions like the sequence of delivery or restrictions concerning stacking, tilting or rotating are taken into consideration.

Open 3D visualization libraries are available from various programming languages such as Java, Python etc.

4.1.10.2 API Reference

The following methods are available:

Name	Description
packBins	Performs a bin packing operation as described by the request and directly returns the packing result in the response.
startBinPacking	Starts a bin packing operation as described by the request. The result informs if operation is accepted and in this case returns an operation ID.
getStatus	Returns the status of a bin packing operation.
getPackedBins	Retrieving the results of a bin packing operation specified by its ID.
cancelBinPacking	Terminates a bin packing operation and deletes the results specified by its ID.

4.2 General notes

PTV Developer offers a suite of RESTful APIs built on the OpenAPI standard, enabling seamless integration into existing system environments. The platform supports modular deployment—users can select specific services (e.g., routing, fleet tracking or route optimization) and scale as needed. The comprehensive suite of cloud-based location service APIs helps to overcome the complex logistical and geographical challenges faced by modern transportation and logistics companies. The flexible use of the APIs, tailored for the entire transportation chain, ensures that your system remains agile and scalable without unnecessary overhead.

Our APIs are engineered to deliver practical, real-world solutions. They consider real-life constraints such as truck-specific routing, live traffic conditions (powered by TomTom), and fleet optimization to provide the most accurate travel times and cost-efficient routes. This leads to improved communication, better planning, and increased operational efficiency.

PTV Developer is backed by state-of-the-art cloud technology and a high service level agreement (SLA) of 99.9% uptime, ensuring maximum reliability and performance.

Special note to European customers: All data is securely hosted within the European Union, guaranteeing full compliance with GDPR (DSGVO).

Whether you are visualizing global maps (powered by HERE data), calculating precise ETAs, or optimizing fleet operations, PTV Developer equips you with the tools to build intelligent, future-ready logistics solutions.