



Fraktjakt API

Version 4.9.0 - EN

2025-11-07

Table of contents

Welcome!	5
Complete Shipping Modules	5
Ready made programming libraries	5
Help	5
Latest version of manuals	6
Always contact Fraktjakt	6
Overview	7
Versions	8
Why integrate with Fraktjakt?	15
Configuration Possibilities with Fraktjakt	16
Useful functions	16
Server	18
Testing	18
Test mode	19
TEST API – the test environment	19
Interaction Diagram	20
Integration Scenarios	21
1. Several shipping services are presented to the customer	21
2. One shipping service is presented to the customer	21
3. Fixed shipping price, using webshop's pre-determined shipping service	22
4. Fixed shipping price, webshop chooses shipping service later on	22
5. Webshop portal searches and orders shipping for a member webshop	22
6. Show what's happening with a shipment after it's been purchased in Fraktjakt	23
Registrations	24
Register a company with a shipping-integration	24
Register an integration that is linked to your existing company	25
About the API calls	26
General information about the XML in the API calls	26
Common to all API calls	27
Common in all API responses	27
Consignor element	29
Address elements	30
Address management	33
Commodity elements	34
Parcel-elements	40
Fraktjakt Query API	41
XML for Query API calls	41
When the response is more important than precision	42

Example 1 - Regular shipping query	47
Example 2 - Regular shipping query with given parcels.....	48
Example 3 - Regular shipping query with multiple packages.....	49
Example 4 - Search for a specific shipping service	50
Example 5 - Regular FOREIGN shipping query.....	50
Response.....	52
Requery API (cached shipment search).....	59
XML for Requery API calls.....	59
Example.....	60
Response.....	60
Order API - make a shipment ready to be purchased.....	61
Calltype 1 - Create an order from a previously created shipment.....	61
Call type 2 - Create and order directly without a previously created shipment.....	62
XML for Order API calls.....	63
Notes:.....	70
Example - Call type 1 Domestic.....	71
Example - Call type 1 International.....	71
Example - Call type 2 Domestic.....	72
Example - Call type 2 International.....	73
Response.....	75
Notes:.....	78
Shipment API	
(Merchant controlled).....	79
Call.....	79
Notes.....	86
Example - Domestic.....	86
Response.....	87
Customs Handling.....	89
Address, Sender, Recipient Block (Sender and Recipient Address).....	89
Commodity Block (Product Information).....	89
Order API and Shipment API (General Customs Information).....	91
Recommendations.....	91
Track & Trace API.....	92
Do you want to create your own page instead?.....	93
The Call.....	93
Response.....	94
Status Codes.....	97
Shipping Documents API.....	98
Call.....	98
PROD.....	98
API.....	98

Response.....	99
Dokument types.....	101
Links to the shipment.....	102
Webshops API	
(create a user and a webshop).....	103
XML for webshop API calls.....	104
Response.....	107
Service Point Locator API.....	108
Service Point Selector API.....	111
How to let customers choose agents.....	112
Option 1 - Agent Selection Link.....	112
Option 2 - Service Point Locator API and <agent_id> tag.....	113
Option 3 - Service Point Locator API and Service Point Selector API.....	113
Shipping Products API.....	114
Status API.....	115
Testing using the API Test Page.....	116
Tips when planning a parcel module integration.....	117
Get faster response times.....	117
Webhook Callback.....	118
Other integration possibilities.....	119
Common parameters to all links.....	119
View information about an individual agent.....	119
Change password.....	119
Administration of the integration associated with the logged-in user.....	119
Administration of addresses.....	119
Track a shipment based on tracking number and shipping company.....	119
Information on all shipping services currently supported via Fraktjakt.....	120
More information on search results.....	120
Terms and conditions for the shipping product.....	120
Other information about the shipping product.....	120
Common problems.....	121
Error reporting.....	122
Support.....	123
Contact us	123
Customer support office hours.....	123
Appendix 1 - Language.....	124
Appendix 2 - Countries in Fraktjakt.....	125

Welcome!

This document describes how to use the Fraktjakt API to integrate Fraktjakt with other system and services such as webshops and business systems.

In order to use the Fraktjakt API, you need to have some knowledge of HTML and XML.

Complete Shipping Modules

If you want to integrate Fraktjakt with an online store such as Klarna checkout, Visma Webshop, Nordisk E-Handel, CubeCart, OCD/Systems, Joomla, FortNox, Quickbutik, AbenteCart, LiteCart, Odoo, OpenCart, Fortnox, PrestaShop, Shopify, WooCommerce EasyCart, Joomla Virtuemart, X-Cart, Zen Cart, OsCommerce, or Magento, there are complete modules for it. Instead, we recommend that you download these modules from <https://www.fraktjakt.se/services/download>. Most modules are completely free of charge.

Ready made programming libraries

If you are programming in PHP, Python or Ruby on Rails, everything has already been done. There are ready-made libraries that are only to download and use.

A complete list of modules and libraries can be found here:

<https://www.fraktjakt.se/services/ecommerce?locale=en>

Help

If you encounter any problems while trying to integrate your system with Fraktjakt, we will be glad to help out. We appreciate any questions and comments that you may have about our product and about this documentation.

Contact us at api@fraktjakt.se or https://www.fraktjakt.se/om_fraktjakt/kontakt

There is also a Google group for announcements and support that we strongly recommend

<https://groups.google.com/d/forum/fraktjakt-api>

Latest version of manuals

During your integration work you may need to change some settings on your registered Fraktjakt account. Instructions on how to do this (currently only in Swedish) can be found in the Fraktjakt webshop manual for integrators:

https://www.fraktjakt.se/downloads/fraktjakt_manual_webbutik.pdf

The latest version of the document you are reading right now can be found at:

https://www.fraktjakt.se/downloads/Fraktjakt_API_en.pdf

Always contact Fraktjakt

We strongly recommend that you always contact our integration support before you start and build an integration with Fraktjakt.

Write to api@fraktjakt.se and we will help you with planning the work. It usually saves time.

Overview

Fraktjakt is a service for comparing, purchasing and managing shipping services over the Internet. It is available both as a website (<https://www.fraktjakt.se>) and as an API (Application Programming Interface) that can be called by other services on the Internet. This document describes how you can use the *Fraktjakt API* to integrate Fraktjakt with other systems and services, such as webshops, business systems, on-line marketplaces, etc.

The basic philosophy of the Fraktjakt API is that it should be as simple as possible to integrate any system with Fraktjakt. Therefore, we try to keep details such as configurations and settings separate from the API; these tasks are performed manually using the Fraktjakt website instead.

All new releases we release should also be backward compatible with old versions. Therefore, it is important to send in the version of the API that is used in the call. The version is specified in the `<api_version>` element.

The current version of the API is solely XML-based. We may add interfaces based on Web Services or SOAP if we see that there is a demand for this.

Versions

Important changes made in this document. Notable additions or changes since the last version of this document are indicated with a yellow background.

Version	Date	Change and additions
4.9.0	20251107	New Shipping Products API added. Rate-limiting fully activated.
4.8.0	20250820	Support for <commodities> inside <parcel>-elements
4.7.0	20250605	Support for limited quantity of dangerous goods with the tags dangerous_goods and dangerous_goods_un_nr on a commodity. Name and description-fields for commodities has more information. Note on how <address_from> should be used. Clarification about the call-type to Fraktjakt. Clarification about shipment_id in Order API type 1.
4.6.1	20250430	Information about the Test API server being shut down and replaced with the setting for creating test shipments.
4.6	20250130	New link to shipping company logos in the response to the Query API where the logo is a square. Possibility to do individualized calls to the list of available shipping_products. Add information on what is used for customs Added information under <i>Server</i> about future rate-limits that will be implemented to our API:s. Additional states in Tracking API. Support for Norwegian MVA-number Update translations from Swedish to English Updated information in commodity elements that <name> and <description> value shall be understandable in the receiving country customs. Included support for real packing slips with id 14 and clarified id 10 name to "delivery note"
4.5	20230914	New API to check server status. New tag 'server_status' in all API responses indicating temporary server problems such as if we are not in contact with a shipping company. Option to send in EORI number for the recipient. New tag if you want insurance when needed.

		<p>Tags in price results that show any price and VAT on the insurance.</p> <p>Fewer mandatory fields in Webshop-API.</p> <p>More ready-made integrations with different systems.</p> <p>New tag in Shipment API and Order API <message_to> for a short message to you customers that are included in the emails from Fraktjakt to the customer.</p> <p>A third street address line for Saudi Arabia.</p>
4.4	20230101	<p>New tag <shipped> on commodity to determine whether the item should be included in the shipping or not.</p> <p>Support for direction instructions and entry code on addresses with <instructions> and <entry_code> tags.</p> <p>Clarified and standardized errors in the Service Point Locator API.</p> <p>New Test mode for integrations.</p>
4.3.1	20220913	<p>Information about how to display one agent.</p> <p>Information about own layout on the agent selection link.</p>
4.3	20220803	<p>New tag <no_agents> for Order API.</p> <p>New tag <agent_id> in the Order API and Shipment API to select a specific agent.</p> <p>Better documented extra parameters to the link for agent/service point selection.</p> <p>New link in the response from the Order API and Shipment API which, when called, deletes the freight that was just created in the API call. If it is not paid, it will be deleted, otherwise it will be annulled.</p> <p>New Service Point Locator API to find information about the nearest agent to an address and a shipping company.</p> <p>New link in the response from the Order API and Shipment API that shows an example of a call to the Service Point Locator API.</p> <p>New chapter that summarizes how to let the end customer choose an agent for a shipment.</p> <p>New Service Point Selector API to let the end customer choose a representative.</p> <p>More information about shipping services in the answer at https://www.fraktjakt.se/shipping_products/xml_list</p>
4.2	20220503	<p>The ability to submit shipping product_id in the Shipment API instead of the favorite service setting on the integration.</p> <p>New link in the response from the Order API and Shipment API which, when called, creates a return shipping based on the shipping that was just created in the API call.</p>
4.1	20220201	Webshop API rewritten.
4.0.1	20211201	Additional information about the tag <no_prices>.
4.0	20210915	Important new functionality:

		<p>Webhook connection in Query API, Order API and Shipment API. It is a way to tell Fraktjakt that Fraktjakt should send information about whats happening with the shipment.</p> <p>New API for downloading shipping documents - Shipping Documents API.</p> <p>It is possible to get the shipping documents for shipments that has been paid for, but has not yet been sent. The reply comes as a Bas64-encoded field in an XML.</p> <p>New taggs for sender - <tax_id> and <ioss>. Used for shipments in or out of EU.</p> <p>Commodity templates can be used with wildcards to complete submitted information with customs information and translations to customers.</p> <p>The parcel block broken out into its own chapter.</p> <p>Order API now has a tag with a link to select agents. It works as long as the freight is not purchased in Fraktjakt.</p> <p>If you requests a cached freight search in Query API or ReQuery API for a freight where the agent is selected, a result adapted to that agent is displayed instead.</p>
3.10	20210615	<p>Clarification on the value of commodities.</p> <p>Support for eori, voec, gb_vat, rex as new tags in Order API and Shipment API. Located in the <sender> block.</p> <p>The <order_id> tag is no longer used in the Order API and will be removed in the future. For the time being, it does not contain current information.</p> <p>The response from Trace API now also shows the name of the transport company that handles the freight.</p> <p>All shipping documents of a shipment are accessed via the Trace API.</p> <p>Full locale support in the Trace API.</p>
3.9	20200910	<p>Country is not needed in adresses. Set to SE if not entered.</p> <p>Query-api now supports to offer the end customer to choose an agent.</p> <p>Recipient name no longer required in Order-API asnd Shipment-API.</p>
3.8	20200515	<p>Clarification on whether a price shown is with or without VAT.</p> <p>Currency Support.</p> <p>Information about Delivery notes and type of goods.</p> <p>Better message iif Fraktjakt don't have the information about closes delivery agent.</p> <p>New tag in the query api that specifies whether the service has delivery to agents or not.</p> <p>Explanation how notification is chosen in Order-api.</p>
3.7	20200205	<p>New shipping modules for integration.</p> <p>Ireland requires postal code.</p>

		<p>Stret_address_1 no longer required in an address.</p> <p>The error levels in all answers are better explained.</p> <p>New tags in all replies <tracking_link> and <tracking_code> with link to track the shipment.</p> <p>Contact email to customer service.</p> <p>Order API: The <user_notes> tag will not be supported in the future for bookings. It is not needed.</p> <p>Shipment API: Support for the <parcels> tag.</p> <p>Address lines longer than 35 characters are cut.</p> <p>Submitted addresses are matched against the address register and the contact register.</p>
3.6	20190625	<p>The highest of the <value>-tagg and the commodities total value is used for customs handling and to calculate if free-shipment prices is to be displayed in query-api.</p> <p>New tagg for recipient - <tax_id>. Used for tax identification number for shipments in or out of EU.</p>
3.5	20190426	<p>Changed the account type "webshop" to "integration" to corelate with our new account types.</p> <p>Updated layout to the same template as our other user guides and manuals.</p>
3.4	20190310	<p>New tag for specifying sender details <sender> in Order API and Shipment API.</p> <p>Shipment API supports support receipt paid shipping.</p> <p>Clearer how to send pallet, stackable and not stackable.</p>
3.3	20190120	<p>New PHP module.</p> <p>The chapter on how to calculate the package size based on the size of the commodities has been removed, since it is now done server-side in Fraktjakt.</p> <p>Logic for all freight searches is improved. Commodities are the element you should use instead of Parcels. Then a packing algorithm can be used (bin-packing) to instead calculate the appropriate package.</p> <p>The ability to specify commodities instead of parcels in the Query API and Order API.</p> <p>New chapter on the Commodity element and fewer absolute requirements for the content in the elements.</p> <p>The ability to specify Length, Width, and Height in the Commodity element to allow for bin-packing calculations of optimal packages to be used in shipping.</p> <p>Other new tags on commodities: in_own_parcel, article_number, shelf_position</p> <p>New tag for Query API and Shipment API that overrides the settings for how the search is to be sorted.</p> <p>Better error management if you enter an incorrect number in the <shipping_product_id> element in the Order API.</p> <p>Less stringent requirements for input data. If certain information is missing, the call is not</p>

		<p>incorrect, but the missing information can be completed later in Fraktjakt.</p> <p>Better support for calls in English.</p> <p>Various clarifications.</p> <p>Updated layout of the document.</p>
3.2.1	20181108	<p>The recipient's email address is not mandatory in the Shipment API, but it is highly recommended that it be sent in (just a correction of an incorrect text in the latest version).</p> <p>A new element <tracking_code> containing the code required to track the shipment is now included in the response from the Track & Trace API.</p> <p>A new element <tracking_number> containing the shipments tracking number is now included in the response from the Track & Trace API when the shipment is fully processed (purchased).</p>
3.2	20181024	<p>The recipient's email address must not be specified in the Shipment API.</p> <p>New section that describes the new Track & Trace link.</p> <p>A new element <tracking_link> containing the shipments' tracking link is now included in the response from the Track & Trace API.</p>
3.1.1	20180922	<p>The "Change and additions" text describing the new elements <status> and <access_link> added in version 3.1 have been corrected (Requery and Track & Trace were added where missing).</p>
3.1	20180915	<p>New URL's for TEST API and PROD API environments (using https).</p> <p>All displayed Fraktjakt URL's and clickable links now have https:// prefix.</p> <p>Added section "Register a webshop that is linked to your existing company" to cover those cases where the user already has a company registered at Fraktjakt.</p> <p>A new element <status> containing the API call status is now included in the response to the Query, Requery, Track & Trace, Order & Shipment API calls.</p> <p>A new element <access_link> containing a link for managing a shipment is now included in the response to the Query, Requery, Order & Shipment API calls.</p> <p>A new element <amount> containing the cost of the shipment is now included in the response to the Order API calls.</p> <p>Notable additions or changes since the last version of this document are indicated with a yellow background.</p>
3.0	20180611	<p>Synchronized with the Swedish version of this document dated 20180427.</p> <p>All text that has been added since the last version of this document is indicated by a green background.</p> <p>A new API has been added – The Shipment API.</p> <p>Two sections have been added to describe XML element blocks that are common to more than one API - specifically the consignor and address elements and their subelements.</p> <p>A section about integration scenarios has been added.</p>

		<p>New webshop settings that have been introduced in Fraktjakt are described.</p> <p>The relationships between accounts, companies and webshops have been simplified.</p> <p>Descriptions of existing functions have been clarified and improved.</p> <p>New “redirect” parameters have been introduced in order to receive API responses from Fraktjakt in different formats.</p> <p>A shipment can now always be accessed/managed via a single uniform URL - https://www.fraktjakt.se/shipments/show/SHIPMENT_ID&access_code=ACCESS_CODE</p> <p>Changes related to addresses:</p> <p>city_name is no longer required for countries where postal code is required. Fraktjakt determines the city name based on the postal code.</p> <p>country_subdivision has been eliminated. It shall not be included in any API calls.</p> <p>Recipient addresses may now include in country codes.</p> <p>Fraktjakt now supports customised notifications.</p> <p>export_reason is now required for both export and import shipments.</p> <p>Improved language support.</p> <p>Language can now be specified for notifications to receivers of shipments.</p> <p>Appendix 1 and 2 added.</p>
2.91	20150225	<p>New non-Required elements in Consignor field.</p> <p>system_name – name of the platform sending the API call. It can be the name of your webshop platform, i.e. "Magento", Prestashop", "WooCommerce", etc.</p> <p>system_version – version number of the system mentioned in the "system_name" field above.</p> <p>module_version – version number of the Fraktjakt shipping module in use.</p> <p>api_version – version number of the Frakjakt API in use (should be the same number as this documents version number!).</p>
2.9	20140616	<p>Clarification of the API's functionality:</p> <p>1) a parcels length, width and height must be provided (Required: Yes) if the <parcel> element is included in the call to Order API call type 2 [see Order API Call section].</p> <p>2) to create an order from a previous query call, a parcels length, width and height must be provided in the original query call [see Query API Call section].</p>
2.8	20140603	<p>New element <shipper_info> in the Query and Requery API to switch on/off additional information about the shipper (in the Api response) for each returned <i>shipping_product</i>.</p> <p>New element <name> under <shipping_product>, where the shipping product name is returned without also containing the shipper name.</p> <p>Corrected name of element (from <i>time_guarantie</i> to <i>time_guarantee</i>).</p>

2.7	20140307	The elements <i><commodities></i> and <i><parcels></i> are no longer required when calling Order API type 2 to order <i>Fraktbody</i> products (<i>shipping_product_id</i> = 137 – 141).
2.6	20140305	<i>Reason for export</i> must be entered into the Order API for international shipments. Added some additional XML examples for Query and Order API calls.
2.5	20140301	Support for <i>Fraktjakt+</i> and <i>Eget fraktavtal (individual shipping contracts)</i> subscriptions.
2.4	20140225	Fixed bug in multiple multiple shipments handling. Corrected section about <i>"Error messages from the API"</i> . Added more error messages and solutions. Address handling explained under <i>Order API</i> .
2.3	20140224	Added functionality in the <i>Order API</i> , to enable creation of an order without creating a shipment through <i>Query API</i> first. New Fraktjakt API test page Track & Trace API (<i>trace_xml</i>) now requires authentication (include Consignor ID & Key). Fixed bug in MD5 handling, to respond with a faster cached reply, when you have entered the same <i>Query API</i> call (within 24 hours).

Why integrate with Fraktjakt?

We have a simpler API than most transport booking APIs.

Multiple services with one integration. If you write an integration directly with a certain shipping company, there will soon be a demand for an integration with another shipping company. By integrating with us, your customers get access to more choices.

We market your system on our site.

Customers often want a service for 95% of their freight, but a good solution for the remaining freight. Say that a store usually sells within Sweden, but from time to time someone wants it shipped to Japan, or extra fast. It is solved with an integration with Fraktjakt.

Customers get cheaper shipping than if you integrate with a shipping company. Fraktjakt has a built-in packaging algorithm that always ensures that you use the smallest possible packaging and do not pay to send air.

You do not need your own shipping agreements to get started. We have pre-negotiated agreements with most carriers. If the customer has their own freight agreements, they can of course also use these through Freight Hunting.

The service is completely free to use!

By integrating the Fraktjakt order part with your program's sales part, you get a direct connection between your program and different shipping companies. Fraktjakt helps you with all communication towards them, construction of all necessary shipping documents and handling of all bookings and other things related to shipping.

Configuration Possibilities with Fraktjakt

- Selection of which shipping services your customers may choose when they make purchases in your webshop;
- Selection of whether to sort the shipping alternatives by **time** or **price** when they are presented to your customers;
- Selection of whether shipping time will be displayed as **Number of days** or **Arrival time**;
- Selection of **Free shipping** when a customer's purchase exceeds a specific amount;
- Setting a price for picking up a product in person instead of having it shipped;
- What measures the webshop parcels has and how to use it automatically. For example, as Standard Package and Package Templates.
- Definition of optional standard box dimensions, if you want Fraktjakt to use these;
- Selection of which days shippers can pick up products for delivery;
- Definition of pick-up times, if applicable;
- Custom track-and-trace page (with your logo, etc.);
- Custom shipping dispatch notification e-mails to your customers;
- Control over which categories of e-mails you receive from Fraktjakt.
- Packing method (bin-packing algorithm). In what way Fraktjakt should calculate which package should be used.
- Default addresses can be set per company and integration.

Useful functions

- Compare your own shipping contracts with other shipping options to always choose the most optimal.
- Fraktjakt can handle all products in the customer's shopping cart, regardless of whether they are sent as separate parcels or if multiple products are packaged together in a standard box.
- Fraktjakt allows you to define the dimensions of individual products. This provides more exact price calculations.

- Fraktjakt can determine which products in your customer's shopping cart will fit into which Parcel Templates (bin-packing) and base the shipping prices on your Parcel Template's size. With Fraktjakt you will not pay to ship air.
This is normally cheaper than basing the shipping prices on the dimensions of your individual products.
- Fraktjakt estimates delivery times based on your settings, i.e. which days shippers can come and pick up products for delivery, along with any specific pick-up times that you may have defined.
- Your customers only see the shipping alternatives that you want them to be able to choose.
- When your customers choose delivery that is not door-to-door, they are shown the nearest pick-up location and how far it is from their address to that location.
- Fraktjakt provides links to maps that help your customers locate pick-up locations.
- Fraktjakt prepares booking and pick-up of one or more shipments.
- Frakt creates shipping documents for both domestic and international shipments, to be printed out on your own printer.
- Possibility to connect multiple integrations to one account and multiple users to every integration, all on an invoice.

Server

URL: <https://api.fraktjakt.se>

This is where you direct all API calls.

This environment is optimized to provide more efficient searches and better stability and security for your integration.

We have implemented "rate-limiting" on our APIs as follows:

- The limit is 120 requests per minute per consignor key. The speed limit is based on a sliding window algorithm with a period of 5 seconds, so the actual limit is 10 requests per 5 seconds.
- When the number of requests reaches the speed limit, HTTP 429 (Too Many Requests) will be displayed, making it very clear if the speed limit is being enforced.
- Any overuse (burst) during a period will mean that the speed limit starts to apply until enough time has passed to keep the average at 10 requests per 5 seconds.

NOTE: Please do not send API calls to <https://www.fraktjakt.se>. This could seriously slow down your application, provide unpredictable results and make it more difficult for us to troubleshoot any API-related problems you encounter.

NOTE: The <https://www.fraktjakt.se> URL should only be used from your web browser, when you are managing your shipments, printing out your shipping documents, configuring your Fraktjakt account, etc.

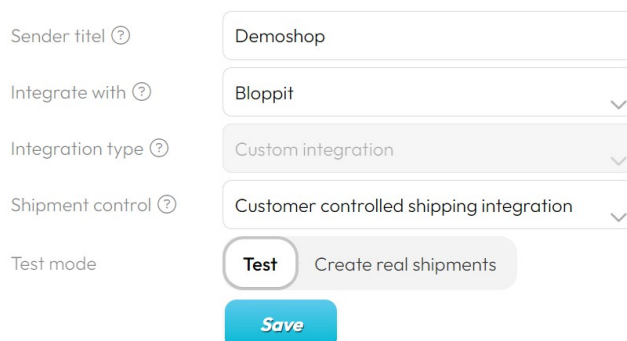
Testing

During the development of an integration against Fraktjakt, it is absolutely necessary to test the code. **Please put the new integration into Test Mode** (see instructions below).

We strongly recommend that you contact api@fraktjakt.se during the entire development process.

Test mode

A newly created integration in Fraktjakt does not start in test mode. However, it is easy to switch to it in the settings for the integration.



Sender titel ?	<input type="text" value="Demoshop"/>
Integrate with ?	<input type="text" value="Bloppit"/>
Integration type ?	<input type="text" value="Custom integration"/>
Shipment control ?	<input type="text" value="Customer controlled shipping integration"/>
Test mode	<input checked="" type="radio"/> Test <input type="radio"/> Create real shipments
<input type="button" value="Save"/>	

If the integration is in Test mode, no sharp shipments are created. Instead, test shipments are created that do not work. It is not possible to send goods with these shipments, all shipping documents are marked with test and unusable and all bookings are test bookings.

Test shipments cannot be purchased automatically, but you do not have to pay for them in the cart.

In this way, you can risk-free test your integration with Fraktjakt. When the development is complete, you simply switch off Test mode and thus immediately start running in production mode.

If there is a need to both be able to further develop and at the same time send in other shipments in production mode, we recommend that you register two integrations in Fraktjakt. One for test calls and one for production calls.

TEST API – the test environment

We have also previously had a dedicated test server. However, we plan to shut this down on June 27, 2025 as it created unnecessary problems with different accounts and discrepancies with different shipping price calculation models and settings between the environments.

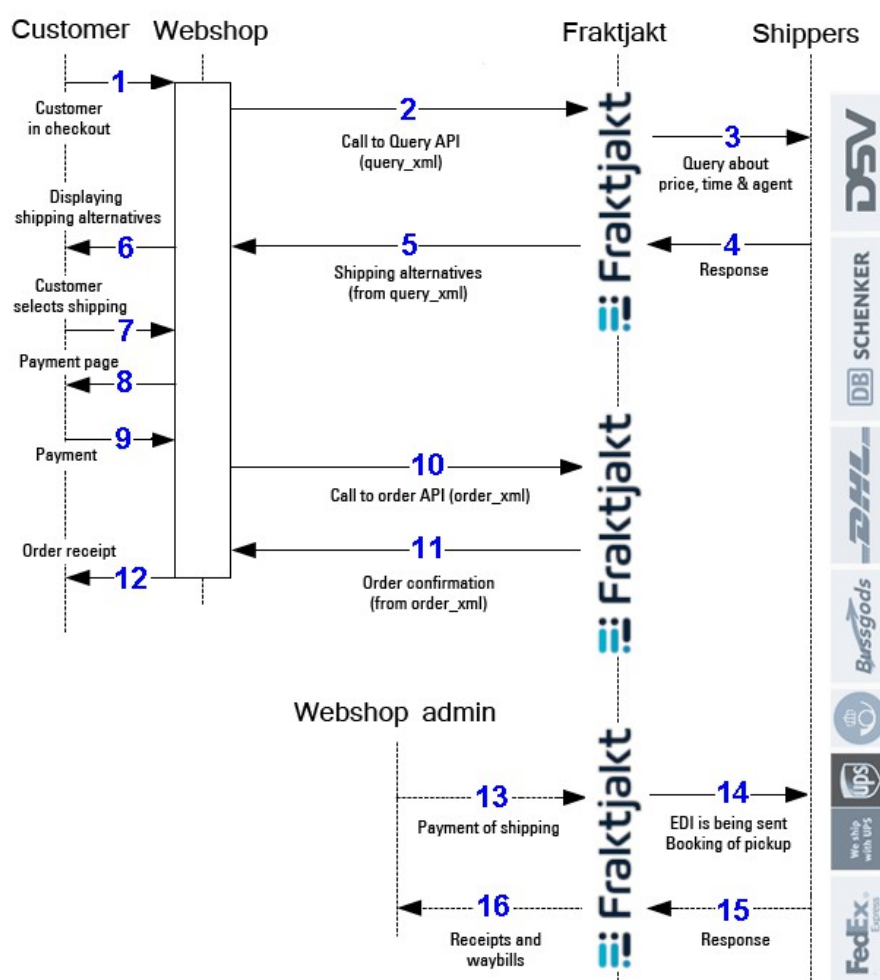
We therefore recommend all developers to switch to the Test Mode setting in the instructions above.

Interaction Diagram

The following diagram depicts the flow of interaction between a webshop customer, the webshop, and Fraktjakt.

The details that are shown of the interaction between the webshop customer and the webshop are just an example. The exact details will vary, depending on the workings of each individual webshop.

Note that the webshop manager needs to manually log on to the Fraktjakt website in order to pay for the shipping services that the webshop's customers have ordered. Multiple shipping orders can be paid for and handled simultaneously using Fraktjakt's transport administration interface.



Integration Scenarios

Listed below are some of the most common scenarios when using the API's.

1. Several shipping services are presented to the customer

Webshops that want to let their customers choose the shipping service can use the Query API in combination with Order API, type 1. The customers choice may be based on the price or maybe instead on the proximity to the agent location where they will pick up their package.

- The webshop performs a price query using Query API.
- The query results are displayed as alternative shipping services in the webshop shopping cart or checkout.
- The customer chooses one of the presented shipping services.
- The webshop creates a shipment order in Fraktjakt by sending the customer selected shipping service in a call to Order API, type 1.
- The webshop administrator manages the new shipment order in Fraktjakt via this link: https://www.fraktjakt.se/shipments/show/SHIPMENT_ID?access_code=ACCESS_CODE

2. One shipping service is presented to the customer

The above query can be narrowed down to just one specific shipping service in the case that the webshop does not want the customer to decide on the shipping service, since it wants to always use just one specific shipping service.

- Retrieve the price of only one shipping service using Query API, see "Example 3 - Search for a specific shipping service (enter 'shipping_product_id') for details.
- The query result is displayed as the only shipping service in the webshop shopping cart or checkout.
- The webshop creates a shipment order in Fraktjakt by sending the shipping service in a call to Order API, type 1.
- The webshop administrator manages the new shipment order in Fraktjakt via this link: https://www.fraktjakt.se/shipments/show/SHIPMENT_ID?access_code=ACCESS_CODE

3. Fixed shipping price, using webshop's pre-determined shipping service

This scenario illustrates a webshop that prefers to offer a fixed shipping price to its customers and wants to always use a specific pre-chosen shipping service to fulfill all of its order deliveries. This can be facilitated by using the Order API, type 2 which can directly create a new shipment order in Fraktjakt without necessitating a price query in advance.

- Find the available shipping services at Fraktjakt using this listing:
https://www.fraktjakt.se/shipping_products/xml_list
- The webshop creates a shipment order in Fraktjakt by sending the shipping service in a call to Order API, type 2.
- The webshop further manages the shipment order via the link:
https://www.fraktjakt.se/shipments/show/SHIPMENT_ID?access_code=ACCESS_CODE

4. Fixed shipping price, webshop chooses shipping service later on

A webshop that wants to offer a fixed shipping price to its customers and wants to easily transfer the order details into a new shipment order in Fraktjakt. The webshop later on manages the shipment order in Fraktjakt, choosing the appropriate shipping service itself.

- Send all the customer information, including delivery address, and the purchased item details to Fraktjakt using the Shipment API.
- The webshop further manages the shipment order via the link:
https://www.fraktjakt.se/shipments/show/SHIPMENT_ID?access_code=ACCESS_CODE

5. Webshop portal searches and orders shipping for a member webshop

This is useful when a webshop portal integrates multiple webshop customers with Fraktjakt.

- The webshop portal does a price query with Query API, possibly searching for individual shipping services by adding a shipping_product_id.
- A customer chooses one of the presented alternatives.
- The webshop portal submits the selected shipment to Fraktjakt using Order API, type 1 (alternatively 2), with the integration credentials of its

member webshop using the `referred_consignor`-element.

- Fraktjakt returns a link by which the shipping purchase can be completed by the member webshop.

6. Show what's happening with a shipment after it's been purchased in Fraktjakt

- Fraktjakt has many settings that allow webshops to create a customized page for customers to track their shipments in a user-friendly manner.
- You can also create your completely own parcel tracking page with the Track & Trace API on page 92.

Registrations

Register a company with a shipping-integration

Before your system can communicate with Fraktjakt, you must have a Consignor ID and a Consignor key.

1. If you already have a Fraktjakt user account, log in to it now.

<https://api.fraktjakt.se/account/login>

2. If you already have a company registered with Fraktjakt, go to the next section "Register an online store that is linked to your company". Otherwise, go to the "Register a Company" page:

<https://api.fraktjakt.se/account/register>

3. Fill out the entire form and indicate that you want a shipping integration (important). Then press the [Register] button at the bottom of the page.
4. Enter the activation code in the appropriate input field. The code should be sent to the email address you provided during registration.
5. Now you should be logged in! (or, still logged in if you started with step 1). Your user name should appear in the web browsers upper right corner.
6. Change the active account to the new integration, in the top right corner of the browser. It appears there as a menu choice under both your private user account and your registered company.
7. Then go to the "Settings" page of your integration:
<https://api.fraktjakt.se/webshops/change>
8. Here you will find the Consignor ID and Consignor Key that you will soon need to submit API calls to Fraktjakt. You can also change settings that affect the API calls and responses.

Register an integration that is linked to your existing company

Before your system can communicate with Fraktjakt, you must have a Consignor ID and a Consignor key.

A company may have several integrations connected to itself and multiple user-accounts.

1. If you already have a Fraktjakt user account, log in to it now.
<https://api.fraktjakt.se/account/login>
2. If you already have a company registered with Fraktjakt, change the active account to your company. It appears in the browser's upper right corner as a menu selection under your private user account.
3. Then go to the "Settings" page of your company:
https://api.fraktjakt.se/enter_shipper/verify
4. Under the Administration section, click on integrations.
5. Then click the button "+ Register integration".
6. Enter which integration system you are using, and then click the "Save" button.
7. Then go to the "Settings" page of your integration by clicking integrations again in the Administration section, and then clicking on the name of the new integration you have just registered.
8. Here you will find the Consignor ID and Consignor Key that you will soon need to submit API calls to Fraktjakt. You can also change your settings that affect the API calls and responses.

About the API calls

General information about the XML in the API calls

The API supports both UTF-8 and ISO-8859-1 encoding, but we recommend UTF-8 for all communication with Fraktjakt. If you are using the PHP programming language, it is best to choose ISO-8859-1, since UTF-8 tends to cause problems in PHP programs.

The calls that you send to Fraktjakt over the API are sent as HTTP requests.

It is possible to use both post and get calls. Get is usually the simplest, but has limitations on how much data can be used in the calls. If there is a large payload to be sent, we recommend post calls instead. We also recommend this if the calls contain many special characters. However, we recommend that you try to limit the amount of special characters, as many shipping companies only accept standard ASCII characters.

All decimals that appear in numeric values must be decimal points – not commas. Example: "3.14"

In this documentation we use indentation to show that an XML element is contained by the element on the previous line of XML.

The XML is sent in as a parameter named xml in the URL to Fraktjakt. It has to be URL-encoded. https://en.wikipedia.org/wiki/Query_string#URL_encoding

A shipment can be specified either by simply entering the goods, with all dimensions stated, or with items without dimensions and with the packages you intend to use with the specified dimensions. The goods are listed in `Commodities` element and packages in `Parcels` element.

If only the goods are specified, Fraktjakt uses an algorithm (bin-packing) to calculate which of the integration's known packages will be used. Packages are retrieved from package templates.

Common to all API calls

All calls to Freight Charge API have some data that will always be sent, regardless of which API you call.

1. The "xml" parameter must be included in all calls made over the API. This parameter defines the XML data that applies to the call. The XML data should only contain standard XML elements written in lower-case. If the data that you need to send over the API is in an XML-file, you can simply remove all of the line breaks from the file and convert it to a long string of text. Then you use this text as the value of the "xml" parameter in your call to Fraktjakt.
2. There is an optional parameter named "md5_checksum". This is the MD5 checksum for the XML data that is defined by the "xml" parameter (See <https://en.wikipedia.org/wiki/Md5>). If you define the "md5_checksum" parameter in a call to Fraktjakt, the checksum will be used to verify the contents of the "xml" parameter. The "md5_checksum" parameter is optional, but including it will help to speed up the caching of API calls.
3. Remember to URL-encode the XML-string according to this: https://en.wikipedia.org/wiki/Query_string#URL_encoding

Common in all API responses

Fraktjakt always sends back four tags that tell you how the call went. These are Status, Code, warning_message and error_message.

A server_status tag is also sent. If we have a temporary server problem, such as no contact with a shipping company, it will be announced here.

There are three levels of error:

0. Ok. Nothing to complain about in the call and everything went well.

1. Warning. Everything is not right in the call, but Fraktjakt has processed the information anyway. For example, the weight may be missing or the street address is too long. The freight has been sent to Fraktjakt and can be supplemented there if necessary.

When integrating, you should review these messages and try to make sure they disappear, but nothing needs to be shown to the end user.

2. Error. The call to Fraktjakt went wrong. What went wrong appears in the error_message tag.

If the error is on the client side, you need to fix it. If the fault lies with Fraktjakt,

it should be notified to customer service so we can fix it.

Consignor element

All calls to the API **must** include the consignor element. This element tells Fraktjakt who is making the call (through the Consignor ID and Consignor Key combination) and provides some other information that is needed to process the call. Without a properly supplied consignor element a call will not succeed.

<code><consignor></code>	Information about the consignor, i.e. the integration hat is sending the order.
<code><id>YOUR_CONSIGNOR_ID</id></code>	<p>The consignor's customer ID in Fraktjakt.</p> <p>Data type: INT</p> <p>Required: Yes</p> <p>Your integrations id in Fraktjakt. Found in the installation tab for the integration in Fraktjakt.</p>
<code><key>YOUR_CONSIGNOR_KEY</key></code>	<p>The consignor's login key in Fraktjakt.</p> <p>Data type: STRING</p> <p>Max chars: 64</p> <p>Required: Yes</p> <p>Found in the installation tab for the integration in Fraktjakt.</p>
<code><currency>EUR</currency></code>	<p>Selected Currency in ISO 4217 notation.</p> <p>Data type: STRING</p> <p>Max chars: 3</p> <p>Required: No</p> <p>Default value: SEK</p> <p>Currency for the purchase in ISO 4217.</p> <p>Fraktjakt will return the price results in the specified currency.</p> <p>Used when an integration wants to show prices in other than SEK.</p> <p>However, the freight will be paid in SEK in Fraktjakt.</p> <p>Most existing currencies are supported.</p>
<code><language>[sv en]</language></code>	<p>Language in ISO 639-1 notation (6)</p> <p>Data type: STRING</p> <p>Max chars: 2</p> <p>Required: No</p> <p>Default: sv</p>

	<p>See Appendix 1 page 124</p> <p>Sets language for the web store's administrator / user. Warnings and Errors are returned in that language. It also means that the user gets that language the next time it logs in.</p>
<code><encoding>[UTF-8 ISO-8859-1]</encoding></code>	<p>The encoding that applies to text in the returned XML data.</p> <p>Data type: STRING Required: No Default: UTF-8</p> <p>UTF-8 is the default and recommended encoding. ISO-8859-1 is also compatible.</p>
<code><system_name>Prestashop</system_name></code>	<p>Name of system calling the API.</p> <p>Data type: STRING Max chars: 32 Required: No</p> <p>Woocommerce, Magento Prestashop etc..</p> <p>This is to make troubleshooting and support easier</p>
<code><system_version>1.2.3</system_version></code>	<p>Version number of system named in <code><system_name></code> element.</p> <p>Data type: STRING Max chars: 16 Required: No</p> <p>This is to make troubleshooting and support easier</p>
<code><module_version>1.5.0</module_version></code>	<p>Version number of the Fraktjakt shipping module in use.</p> <p>Data type: STRING Max chars: 16 Required: No</p>
<code><api_version>4.7.0</api_version></code>	<p>Version number of the API integrationen in use.</p> <p>Data type: STRING Max chars: 16 Required: No</p> <p>Should be the same version as this document!</p> <p>Ensures that the call is always compatible with future versions.</p>
<code></consignor></code>	

Address elements

In order for a shipment to be created, address elements must be provided, such as `<address_to>`, `<address_from>`, etc.

If you are sending from the regular sender address as it is in Fraktjakt, `<address_from>` should not be specified for best functionality in Fraktjakt.

The element content (between the element tags) must be as follows:

<code><address_*> (<address_from>, <address_to>, etc.)</code>	<p>Address type</p> <p>Data type: STRING Max chars: 25 Required:</p> <p>Replace the “*” in “<address_*”, at left, with the address type, e.g. “to”, “from”, ...</p>
<code><street_address_1>Gjuterig 9</street_address_1></code>	<p>Gatuadress (line 1)</p> <p>Data type: STRING Max chars: 35 (Shippers max) Required: No</p> <p>Strings longer than 35 characters will not generate errors or warnings, but will be truncated.</p> <p>If the shipment is to a 'C/O' address should be in this element. Fraktjakt recognizes the keywords 'C/O' and 'Entry code' and will if possible send that information in the way the shipper want it.</p>
<code><street_address_2>Vån 7 C</street_address_2></code>	<p>Gatuadress (line 2)</p> <p>Data type: STRING Max chars: 35 (Shippers max) Required: No</p> <p>Strings longer than 35 characters will not generate errors or warnings, but will be truncated.</p> <p>Supplementary/further shippinginformation can be given within parentheses in this element. Ex: (Has unloading agreements) (Leave at the loading platform)</p>
<code><street_address_3>123467</street_address_3></code>	<p>Gatuadress (line 3)</p> <p>Data type: STRING Max chars: 35 (Shippers max) Required: No</p> <p>For countries that require three address lines, such as Saudi Arabia an Puerto Rico, there is a third address line.</p>

<code><postal_code>55318</postal_code></code>	<p>Postalcode</p> <p>Data type: STRING Max chars: 16 Required: Yes – except for countries not using postal codes, like Gambia, see Appendix 2, page 125,</p> <p>Postal code is also the primary way of finding the name of the city to which the shipment is going. The <code><city_name></code> element will only be used unclear cases.</p>
<code><city_name>Jönköping</city_name></code>	<p>City name</p> <p>Data type: STRING Max chars: 32 Required: Not if the <code><postal_code></code> line contains a unique postal code for that city.</p>
<code><residential>[0 1]</residential></code>	<p>Specifies whether the address is residential or not (private/residential or commercial)</p> <p>Data type: BOOLEAN Required: No Default = 1 (residential)</p>
<code><country_code>SE</country_code></code>	<p>Country - See Appendix 2</p> <p>Data type: STRING Max chars: 2 Required: No. SE is default.</p>
<code><instructions>Keep right</instructions></code>	<p>Directions for the driver</p> <p>Data type: STRING Required: No</p>
<code><entry_code>1234</entry_code></code>	<p>Entry code / Door Code</p> <p>Data type: STRING Required: No</p>
<code><language>sv</language></code>	<p>Receiver language, see Appendix 1 page 124.</p> <p>Language code: ISO 639-1</p> <p>Data type: STRING Max chars: 2</p> <p>Can ONLY be given for <code><address_to></code>-elements.</p> <p>Specifies which language Fraktjakt should use in email and communication with the receiver.</p> <p>For example, in the email to the recipient when the shipment is shipped and on the Tracking page.</p>
<code></address_*> (</address_from>, </address_to>, etc.)</code>	

Address management

After an address is used in an API call, Fraktjakt tries to match it to an address in the Address Book, your addresses or contacts' addresses. If an address is found in the address book, the submitted address is supplemented with the information from the address book in terms of alternate name, directions, door code and opening hours.

This can be used, for example, to send in an <address_from> that goes to an alternative warehouse room. Fraktjakt then complements the shipment information with the premises opening hours and names.

If you are sending from the regular sender address as it is in Fraktjakt, <address_from> should not be specified for best functionality in Fraktjakt.

Another example is that if the address of a customer <address_to> is submitted and it is already in the Address Book, the address is supplemented with directions and entry code for the customer.

Commodity elements

Used in Query-API, Order API and Shipment API.

Content of shipping. Thus the different items/commodities contained in the packages or on the pallet to be sent. Specified per item. Thus, if three copies of the same item are sent, all are listed in a commodity block with the value 3 in the `<quantity>3</quantity>` element.

If the shipment is to be taxed, more elements are required than if it is only domestic freight. However, it does not hurt to always submit all information.

Note that the weight is given as the **total** weight of all the items of the specified type. Thus, you send 3 items weighing 2 kg, 6 kg must be sent in the `<weight>` element.

The length, width and height are given per individual item and centimeter. Not as the total measure.

In some way, Fraktjakt must know how heavy and how large the freight is to be sought. During the implementation, you should consider what information you have.

- Either you enter the contents of the freight in the `<commodities>` element, with the total weight and preferably all other dimensions given, and let Fraktjakt calculate the packages to be sent based selected Packing Method and the existing and active Package Templates.
- Or, enter the packages to be sent directly in the `<parcels>` element. **Note!** If Compability-mode is activated in the integration's settings, one parcel will be generated with the Standard Parcel's measures.
- You can enter both commodities and parcels. The packages and its dimensions are then taken from the `<parcels>` element.
- For the goods and the packages only the weight is needed, but for more correct information, all measurements (length, width and height) of each item are recommended. Unless the volume dimensions are submitted, only Standard Package selected in Fraktjakt can be used.

It is possible to set in Fraktjakt so that submitted Commodities are matched to the Commodity Templates in Fraktjakt (Settings → Content Completion). If a match is found, the submitted information is supplemented with what is stated in Fraktjakt. It thus provides easy maintenance of all necessary customs information and translation of the information to customs and customers. It is also possible to enter wildcards ('*') in the name, article number or special matching -field of the Commodity Templates in Fraktjakt. In this way, a Commodity Template can be used to save customs information on many items in your warehouse.

For example, a Commodity Template named '*shoes' matches both if you submit 'Low shoes' or 'Stiletto heeled shoes'.

If a commodity is dangerous goods, the tag `<dangerous_goods>1</dangerous_goods>` must be specified. The entire shipment will then be marked as containing dangerous goods and only services where a limited amount of dangerous goods can be sent can be selected.

Please note:

- * Permission from Fraktjakt is required to send a limited amount of dangerous goods with us.
- * Only certain types of dangerous goods can be sent with Fraktjakt and then to a limited amount according to ADR-S (see MSB page on Limited amount of dangerous goods)
- * The sender is responsible for ensuring that all labeling is correct, all safety data sheets are attached and that everything is packaged according to all regulations.

Fraktjakt strongly recommends that all measurements and as many data as possible be specified on each call or in a matching Commodity Template.

With all measurements specified, Fraktjakt can calculate the smallest possible package needed for the transport. This provides very large savings on transport costs, something that Fraktjakt is the only one to offer on the transport market. With package optimization, you usually save more money on transport than if you only focus on freight prices.

<code><commodity></code>	A single commodity type.
<code><name>shoes</name></code>	<p>The name of the commodity type.</p> <p>Data type: STRING Max chars: 64 Required: Yes</p> <p>The value shall be understandable in the receiving country customs.</p> <p>Displayed on the Packing List, Customs documents and Delivery Notes.</p> <p>Is what will appear as the product name on CN22. So it must be an article description that is suitable for CN22.</p> <p>Keep in mind that Fraktjakt's Content Templates can help with translations.</p>
<code><quantity>3</quantity></code>	<p>The quantity of the commodity in the unit specified in <code><quantity_units></code></p> <p>Data Type: INTEGER</p>

	<p>Default value: 1 Required: NO - Set to 1 if the value is missing or is 0.</p> <p>Displayed on the Packing List, Customs documents and Delivery Notes.</p>
<code><shipped>[0 1]</shipped></code>	<p>Indicates whether the item should be included in the shipping at all. If not, it is only used to calculate possible free shipping and such.</p> <p>Data type: BOOLEAN Required: No Default = 1</p>
<code><taric>6403999600</taric></code> <p>(see http://tulltaxan.tullverket.se/#!/taric/nomenclature/sbt)</p>	<p>The harmonized customs code for this commodity type (also called TARIC code).</p> <p>Data type: INT</p> <p>Required: No, but when you include this element in international shipping orders, the shipments will normally get through customs much faster.</p>
<code><quantity_units>EA</quantity_units></code> <p>Allowed values: EA = each DZ = dozens L = liter ML = milliliter KG = kilogram</p> <p>Default is EA (each) if no or invalid value is submitted.</p>	<p>The unit of measure for this commodity type.</p> <p>Data type: STRING Max chars: 2</p> <p>Required: No, but it should be included in international shipping orders.</p> <p>Default: EA (each)</p> <p>Allowed values: EA = each DZ = dozens L = liter ML = milliliter KG = kilogram</p> <p>Only the units 'EA' and 'DZ' can be specified if an advanced bin-packing algorithm is to be used.</p> <p>Displayed on the Packing List, Customs documents and Delivery Notes.</p>
<code><description>Black leather shoes</description></code> <p><i>(Attention! 15 characters minimum. The value shall be understandable in the receiving countrys customs.)</i></p>	<p>Description of the commodity type.</p> <p>Data type: STRING Min chars: 15 Max chars: 128</p> <p>Mandatory: For overseas shipments, it should be stated. Otherwise, it must be specified in Fraktjakt.</p> <p>The value shall be understood in the recipient country's customs.</p> <p>Keep in mind that Fraktjakt's Content Templates can help with</p>

	<p>translations and missing descriptions.</p> <p>The best descriptions are based on the 3 W, along with additional information: What is it? What is it made of? What is it used for? Additional information relevant to the product, such as the gender the clothing is intended for, the size of the water bottle, etc.</p> <p>US customs are extra strict on very good descriptions.</p>
<code><country_of_manufacture>US</country_of_manufacture></code>	<p>Country code according to the ISO 3166-1 alpha-2 (5) standard.</p> <p>Data type: STRING Max chars: 2 Default: SE</p> <p>Required: No, but can be given in overseas shipments.</p> <p>See Appendix 2 page 125 for the codes.</p>
<code><weight>13.1</weight></code>	<p>The cumulative weight in kg of all items of this type in the shipment.</p> <p>The least needed to be stated as a measure unless the <code><parcel></code> elements are specified.</p> <p>Tip - Be extra careful with the correct weight for international shipping. Otherwise they can get caught in customs.</p> <p>Data type: FLOAT</p> <p>Required: NO but international shipments will need it.</p>
<code><length>40</length></code>	<p>The length of one item of the commodity.</p> <p>Specified in centimeters (cm)</p> <p>Data type: FLOAT</p> <p>Required: No, but it is required for advanced bin packing.</p>
<code><width>30</width></code>	<p>The width of one item of the commodity.</p> <p>Specified in centimeters (cm)</p> <p>Data type: FLOAT</p> <p>Required: No, but it is required for advanced bin packing.</p>
<code><height>40</height></code>	<p>The height of one item of the commodity.</p> <p>Specified in centimeters (cm)</p>

	<p>Data type: FLOAT</p> <p>Required: No, but it is required for advanced bin packing.</p> <p>Displayed on the Delivery Notes and Customs documents.</p>
<code><unit_price>99.99</unit_price></code>	<p>The value per unit of this commodity type. It is used in customs and for complaint.</p> <p>The value the item was sold for. If you sell with VAT, the VAT must be included in the value and if you sell without VAT, it must not be included.</p> <p>If you have sold at a discount, it must also be included in what is sent in</p> <p>Data type: FLOAT Required: Yes when the shipment is international</p>
<code><currency>USD</currency></code>	<p>Currency for the value in ISO 4217 standard.</p> <p>Data type: STRING Max chars: 3 Mandatory: No Default value: SEK</p> <p>Fraktjakt will interpretate <code><unit_price></code> in this currency. The value will how ever always be displayed in SEK in Fraktjakt or selected currency in the <code><consignor><currency></code> tag.</p> <p>Most existing currencies are supported.</p>
<code><in_own_parcel>[0 1]</in_own_parcel></code>	<p>Indicates if the item has its own parcel and should not be added to another parcel.</p> <p>Data type: BOOLEAN Required: No Default = 0 (Shared parcel)</p>
<code><article_number>K19-53</article_number></code>	<p>Article number. Used in the packing list and Delivery Notes</p> <p>The SKU (Stock Keeping Unit) can be specified here if used.</p> <p>Data type: STRING Max chars: 64 Required: No</p>
<code><shelf_position>Top Left</shelf_position></code>	<p>Where the commodity can be found when packing it.</p> <p>Data type: STRING Max chars: 64 Required: No</p>
<code><dangerous_goods>[0 1]</dangerous_goods></code>	<p>Indicates whether the item is</p>

	<p>dangerous goods.</p> <p>After approval, it is permitted to send certain types of dangerous goods in a limited quantity (LQ) with Fraktjakt.</p> <p>The responsibility for ensuring that everything is packaged and labeled correctly is always the sender's.</p> <p>The sender is also responsible for the necessary documentation.</p> <p>Data type: BOOLEAN Required: No Default = 0 (Inte farligt gods)</p>
<pre><dangerous_goods_un_nr>UN 2908 </dangerous_goods_un_nr></pre>	<p>Used for dangerous goods.</p> <p>Four-digit code indicating the type of dangerous goods.</p> <p>https://en.wikipedia.org/wiki/Lists_of_UN_numbers</p> <p>Data type: STRING Max chars: 16 Required: No</p>
<pre></commodity></pre>	

Parcel-elements

Used in Query-API, Order API and Shipment API.

How the package looks like being transported. Can be specified without commodities, together with commodities or not at all. See above under the Commodity block for information above on how it works. The norm in Fraktjakt is NOT to send in parcels.

NOTE - If the length is 120 cm and the width 80 cm, it is assumed that a pallet is being transported. If the length is 80 cm and the width 60 cm, it is assumed to be a half pallet. In that case, the height determines whether it is stackable or not. Below 125 cm is stackable, otherwise not.

If the length, width or height of the plot is not specified, or if it is specified as 0.0 cm, the measurements are taken from the Default package template that has been set in the settings for the integration.

<code><parcel></code>	Information about a parcel.
<code><weight>3.2</weight></code>	Parcel weight in kg. If no value is given, the weight from the Default package template. Datatype: FLOAT Mandatory: Yes
<code><length>35</length></code>	Parcel length in cm Datatype: FLOAT Mandatory: No The longest side.
<code><width>23.5</width></code>	Parcel width in cm Datatype: FLOAT Mandatory: No The second longest side
<code><height>15</height></code>	Parcel height in cm Datatype: FLOAT Mandatory: No
<code><commodities></code> See the Commodity chapter on page 34. <code></commodities></code>	The types of commodities that are shipped in this parcel. Remember that the weights of the goods should match the weight of the package. If a commodity is listed here, it should not be listed separately again elsewhere in the call.

Fraktjakt Query API

The API calls that are used for sending queries about shipping services to Fraktjakt are collectively called the Fraktjakt Query API. In this document we will simply use the term "Query API".

You can also get information about the nearest agent.

Your system can use the Query API to retrieve information about the best, cheapest and fastest shipping alternatives. This is the same type of information that web users find when they perform searches at <https://www.fraktjakt.se/>; however, the Query API is faster.

XML for Query API calls

In order to search for shipping services, your system must send a definition of the required shipping services to the Query API. The definition that is sent may comprise one or more parcels and the address that they need to be shipped to. The definition must only have one sender address and one receiver address, regardless of how many parcels are included.

The packages to be sent can either be sent in `<parcels>` element or calculated by Fraktjakt based on the goods to be shipped and listed in the `<commodities>` element.

You can choose to only submit the `<commodities>` element, only the `<parcels>` element or both, see note 3.

After the query, the integration's customer or user is expected to choose shipping product for the shipment. The selection is submitted with Order API Call Type 1, page 61.

Or, the shipment is completed via the link returned in the `<access_link>` element.

If the shipping price is distance-based, the shipping address that you entered on your integration configuration page at Fraktjakt will be used by default.

The Query API uses the following URI:

PROD API environment: https://api.fraktjakt.se/fraktjakt/query_xml

If you want to get the price a specific shipping service the `<shipping_product_id>` element can be used. You can retrieve the ID-number from an earlier search in the `<id>` element under the `<shipping_product>` element or via https://www.fraktjakt.se/shipping_products/xml_list.

You can configure Fraktjakt to send `<arrival_time>` as the expected shipping time in days or as the expected time of arrival. This setting is done on your

configuration page at Fraktjakt.se. The arrival time is calculated based on the shipping product's normal shipping time, the available days for pick-up that you have defined in your settings, and the shipper's pick-up schedule.

The result is sorted by the setting for the web store, or by the `<price_sort>` element.

Note that if you within an hour send a query to the Query API that is an exact duplicate of a previous query, the Query API will return the cached results from the first query.

If the end customer has chosen their agent after the original search, the selected agent is displayed as the closest and only shipping services belonging to the same shipping company that has the selected agent are displayed.

When the response is more important than precision.

You can also do a quick search that we cache locally. It will provide an estimate shipping price quickly when your customers do not need the exact final price. This is also useful when you don't have the receiver address yet.

To do this you put the same address in the receiver address field as in the sender field. This will tell Fraktjakt to make a quick search.

NOTE! Do not create an order from this search. You need to do a new search when you want to create the order and buy the shipment. Otherwise the incorrect address will be sent to the delivery company. Then it could be suitable to set the search for a specific `shipping_product_id`.

Other elements used to provide faster response times are: `<no_agents>`, `<no_prices>` and `<agents_in>`.

Showing fixed prices is also always much faster than the actual prices. If you have fixed shipping prices on your orders, you must also set in Fraktjakt so that only fixed prices are displayed.

The fastest way is not to get prices at all. This may be when a webshop wants to display shipping services, but prices are determined locally. For example, when shipping becomes free after using a discount code, or all shipping costs a known amount. Then the `<no_prices> 1 </no_prices>` tag should be submitted. This means that Fraktjakt only checks which freight services work for the recipient. The best package is calculated as usual and the nearest agent are selected.

XML element tag and example of contents	Description
<code><?xml version="1.0" encoding="UTF-8"?></code>	XML header
- <code><shipment></code>	This is the document element, i.e. the root element of the XML data that describes a shipment that is being searched for.

- <consignor>	Information about who's making the call.
See Consignor-element on page 29	
</consignor>	
- <callback_url> https://my.se </callback_url>	<p>The URL of the server that will receive the call from Fraktjakt's webhook.</p> <p>For more information, see the chapter "Webhook callback" and the document for Fraktjakt's webhook.</p> <p>If the tag is specified, it overrides the integration's webhook setting.</p> <p>Tip: In the tag, it is possible to send in a URL with its own security parameter, which is then used in the call from Fraktjakt. For example? Key = abcdef123456</p>
- <insure_default>[0 1]</insure_default>	<p>Should the freight be additionally insured if the shipping company's basic insurance does not cover the value of the goods.</p> <p>If the tag is specified, it overrides your company's setting regarding insurance.</p> <p>Data type: BOOLEAN Required: No</p>
- <price_sort>[0 1]</price_sort>	<p>To override the settings for how searches are to be sorted. 1 if the search is to be sorted by the cheapest price. If 0 the search is sorted on delivery time instead.</p> <p>Datatype: BOOLEAN Obligatorisk: No</p> <p>Default = The settings for the integration</p>
- <value>10</value>	<p>The cumulative value of all items in the shipment.</p> <p>If the total given value for the commodities is higher than this value, that value will be used instead.</p> <p>Use this field unless the value of the goods is shown in commodities. Otherwise it is not used.</p> <p>Used primarily to know whether free shipping should be used in the answer.</p> <p>Data type: FLOAT Required: No</p>
- <currency>USD</currency>	<p>Currency for the value in ISO 4217 standard</p> <p>Data type: STRING</p>

	<p>Max chars: 3 Mandatory: No Default value: SEK Most existing currencies are supported.</p>
- <express>[0 1]</express>	<p>Flag for limiting the search to only find express shipping alternatives.</p> <p>Data type: BOOLEAN Required: No Default = 0 (Search for both express and other types of shipping services.)</p>
- <pickup>[0 1]</pickup>	<p>Flag for limiting the search to only find shipping alternatives with pickup service.</p> <p>Data type: BOOLEAN Required: No Default = 0 (Do not limit the search.)</p>
- <dropoff>[0 1]</dropoff>	<p>Flag for limiting the search to only find shipping alternatives with dropoff service.</p> <p>Data type: BOOLEAN Required: No Default = 0 (Do not limit the search.)</p>
- <green>[0 1]</green>	<p>Flag for limiting the search to only find shipping alternatives that have some type of environmental certification.</p> <p>Data type: BOOLEAN Required: No Default = 0 (Do not limit the search.)</p>
- <quality>[0 1]</quality>	<p>Flag for limiting the search to only find shipping alternatives that have some type of quality certification.</p> <p>Data type: BOOLEAN Required: No Default = 0 (Do not limit the search.)</p>
- <time_guarantee>[0 1]</time_guarantee>	<p>Flag for limiting the search to only find shipping alternatives that have guaranteed delivery time.</p> <p>Data type: BOOLEAN Required: No Default = 0 (Do not limit the search.)</p>
- <cold>[0 1]</cold>	<p>Flag to indicate whether the shipment contains items that require refrigeration. (3)</p> <p>Data type: BOOLEAN Required: No Default = 0 (refrigeration not required)</p>
- <frozen>[0 1]</frozen>	<p>Flag to indicate whether the shipment contains items that require frozen</p>

	<p>transport. (3)</p> <p>Data type: BOOLEAN</p> <p>Required: No</p> <p>Default = 0 (frozen transport not required)</p>
- <shipping_product_id>30</shipping_product_id>	<p>Include this element if you only want to find the price for a specific shipping product.</p> <p>The ID is found from the call with the product list. See page Fel: Det gick inte att hitta referenskällan.</p> <p>NB: This element can be useful if you want to fill in a price using AJAX.</p> <p>Data type: INT</p> <p>Required: No</p>
- <no_agents>[0 1]</no_agents>	<p>If you want to know the nearest agent or not. If you select to not show them you will receive a significantly faster reply, but give less service to your customers.</p> <p>Data type: BOOLEAN</p> <p>Required: No</p> <p>Default = 0 (show agents)</p>
- <no_prices>[0 1]</no_prices>	<p>No prices shown!</p> <p>Provides the fastest search results and then only shows the services that are available at the specified address.</p> <p>Used if only availability is interesting, or the price is determined locally.</p> <p>Data type: BOOLEAN</p> <p>Required: No</p> <p>Default = 0 (prices are shown)</p>
- <agents_in>[0 1]</agents_in>	<p>If you want to show the nearest agent for the sender or not. If you select to show it, the reply will be much slower.</p> <p>Data type: BOOLEAN</p> <p>Required: No</p> <p>Default = 0 (don't show agents)</p>
- <shipper_info>[0 1]</shipper_info>	<p>If you want extra information about the shippers. The reply will then contain the additional element <shipper> with its sub element's <i>id</i>, <i>name</i> and <i>logo_url</i>.</p> <p>Data type: BOOLEAN</p> <p>Required: No</p> <p>Default = 0 (no additional info)</p>
<commodities>	<p>What is to be transported.</p> <p>If not specified, but there are parcel tags, and the integration has a default Commodity Template in Fraktjakt and it</p>

	specified to be used in searches, the default Commodity Template is used to create content. See Note 3
See the Commodity chapter on page 34.	
<code></commodities></code>	
- <code><parcels></code>	The parcels that will be sent. Can be specified if commodities are not known, or if you have really decided how big the packages should be and do not want to use Fraktjakt's package calculations.
See the Parcel-elements on page 40	
<code></parcels></code>	
- <code><address_to></code>	Note that <code><language></code> can be used.
See address-element on page ?	
<code></address_to></code>	
- <code><address_from></code>	Sender address. If this element is missing the default shipping address entered in Fraktjakt will be used. Required: No (see note nr.1)
See Address-element på on page ?	
<code></address_from></code>	
<code></shipment></code>	

Notes:

1. `<address_from>` normally does not need to be included, since the shipping address that you have entered on your settings page at Fraktjakt.se should apply to all of the shipments that you order via Fraktjakt.

The Query API supports addresses anywhere on the globe; however, Fraktjakt does only handle shipments that is to or from Sweden.

If you do not ship the freight from an agent, the address should be where the package is picked up! Thus, not the sender address, but the physical location from which the package is sent.

2. If the elements, `<cold>` and `<frozen>`, are omitted from your XML, or if they do not contain the value "1" or "true", no refrigerated or frozen shipping alternatives will be included in the search results that the Query API returns. Furthermore, refrigerated and frozen shipping alternatives are only shown when these elements contain "1" or "true". It is thus not

possible to get a search result that includes both refrigerated/frozen and non-refrigerated shipping alternatives in the same call to the Query API.

3. In some way or another, Fraktjakt must know how heavy and how big the shipment is that is queried for. When implementing this API, you should ask yourself, do I know which packages I'm going to use, or do I know what the goods I'm going to send look like? From there, select which elements to use.
 - Either enter the contents of the freight in the <commodities> element, with the dimensions specified, and let Fraktjakt calculate the packages to be sent based on the aktive Package Templates.
 - Or, enter the packages to be sent directly in the <parcels> element. (If Compability-mode is activated, one parcel will be generated with the Standard Parcel's measures).
 - You can enter both commodities and parcels. The packages and its dimensions are then taken from the parcels.
 - For the commodities and packages only the weight needs to be stated, but for more accurate data, all the dimensions (length, width and height) of each commodity and the packages, if you send in which packages to use) are recommended. If the volume measures are not submitted, only the standard package selected in Fraktjakt can be used.
 - If neither <parcels> element or the complete dimensions of the commodities are listed, all the goods will assumed be fit in the Standard Package specified in the settings in Fraktjakt for the integration. If you have not specified a Standard Package, the query will fail! Fraktjakt simply does not have the information needed to calculate the shipping price.

Fraktjakt recommends that all measurements and as much data as possible be specified on each call. Otherwise, the shipment needs to be completed in Fraktjakt, and that it is both slower and prevent automation of freight purchases.

4. If you specify dimensions 120 or 60 on length and 80 on width, Fraktjakt interprets the input package as a pallet or half pallet and searches for shipping products for pallets. If the height is below 125 cm, the pallet will be stackable. Higher heights provide non-stackable pallets.

Example 1 - Regular shipping query

A regular search delivers a response with one or more shipping alternatives. It is thought that you want to send five Rubric's cube.

This XML should work if you only replace the consignor id and key to your own.

See Order API call type1, example 1 page 71 for how the response should be handled.

```
<?xml version="1.0" encoding="UTF-8"?>
<shipment xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance">
```

```

<value>199.50</value>
<shipper_info>1</shipper_info>
<consignor>
  <id>YOUR_CONSIGNOR_ID</id>
  <key>YOUR_CONSIGNOR_KEY</key>
  <currency>SEK</currency>
  <language>en</language>
  <encoding>UTF-8</encoding>
  <api_version>4.7.0</api_version>
</consignor>
<no_agents>1</no_agents>
<commodities>
  <commodity>
    <name>Rubriks kub</name>
    <description>Ett kubpussel i plast</description>
    <quantity>5</quantity>
    <unit_price>35.00</unit_price>
    <currency>DKK</currency>
    <weight>1</weight>
    <length>7</length>
    <width>7</width>
    <height>7</height>
    <country_of_manufacture>DK</country_of_manufacture>
  </commodity>
</commodities>
<address_to>
  <street_address_1>Hedenstorp 10</street_address_1>
  <street_address_2></street_address_2>
  <postal_code>33292</postal_code>
  <city_name>Gislaved</city_name>
  <residential>1</residential>
  <country_code>SE</country_code>
  <language>sv</language>
</address_to>
</shipment>

```

Example 2 - Regular shipping query with given parcels

A regular search delivers a response with one or more shipping alternatives.

This does not specify what you want to send but the size of the package you want to send

This XML should work if you only replace the consignor id and key to your own.

```

<?xml version="1.0" encoding="UTF-8"?>
<shipment xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <value>199.50</value>
  <shipper_info>1</shipper_info>
  <consignor>
    <id>YOUR_CONSIGNOR_ID</id>
    <key>YOUR_CONSIGNOR_KEY</key>
    <currency>SEK</currency>
    <language>en</language>
    <encoding>UTF-8</encoding>
    <api_version>4.7.0</api_version>
  </consignor>
  <no_agents>1</no_agents>
  <parcels>
    <parcel>
      <weight>2.8</weight>

```

```

        <length>30</length>
        <width>20</width>
        <height>10</height>
    </parcel>
</parcels>
<address_to>
    <street_address_1>Hedenstorp 10</street_address_1>
    <street_address_2></street_address_2>
    <postal_code>33292</postal_code>
    <city_name>Gislaved</city_name>
    <residential>1</residential>
    <country_code>SE</country_code>
    <language>sv</language>
</address_to>
</shipment>

```

Example 3 - Regular shipping query with multiple packages

One wants to ship three parcels with different dimensions.

This XML should work if you only replace the consignor id and key to your own.

The customer can have free shipping, so the value is given in the XML.

```

<?xml version="1.0" encoding="UTF-8"?>
<shipment xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <value>199.50</value>
    <consignor>
        <id>YOUR_CONSIGNOR_ID</id>
        <key>YOUR_CONSIGNOR_KEY</key>
        <currency>SEK</currency>
        <language>en</language>
        <encoding>UTF-8</encoding>
        <api_version>4.7.0</api_version>
    </consignor>
    <no_agents>1</no_agents>
    <value>150</value>
    <currency>EUR</currency>
    <parcels>
        <parcel>
            <weight>1.2</weight>
            <length>20</length>
            <width>15</width>
            <height>10</height>
        </parcel>
        <parcel>
            <weight>2.8</weight>
            <length>30</length>
            <width>20</width>
            <height>10</height>
        </parcel>
        <parcel>
            <weight>7</weight>
            <length>40</length>
            <width>30</width>
            <height>20</height>
        </parcel>
    </parcels>
    <address_to>
        <street_address_1>Hedenstorp 10</street_address_1>
        <street_address_2></street_address_2>
        <postal_code>33292</postal_code>
    </address_to>
</shipment>

```

```

    <city_name>Gislaved</city_name>
    <residential>1</residential>
    <country_code>SE</country_code>
    <language>sv</language>
  </address_to>
</shipment>

```

Example 4 - Search for a specific shipping service

(enter 'shipping_product_id')

One want to send four used tires with Bussgods Sverigefrakt to Gisslaved. They are not interested in which agent they are sent to.

```

<?xml version="1.0" encoding="UTF-8"?>
<shipment xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance">
  <value>199.50</value>
  <consignor>
    <id>YOUR_CONSIGNOR_ID</id>
    <key>YOUR_CONSIGNOR_KEY</key>
    <currency>SEK</currency>
    <language>en</language>
    <encoding>UTF-8</encoding>
    <api_version>4.7.0</api_version>
  </consignor>
  <no_agents>1</no_agents>
  <shipping_product_id>25</shipping_product_id>
  <commodities>
    <commodity>
      <name>Tire</name>
      <description>Used tires</description>
      <quantity>4</quantity>
      <unit_price>650</unit_price>
      <weight>25</weight>
      <length>35</length>
      <width>35</width>
      <height>10</height>
    </commodity>
  </commodities>
  <address_to>
    <street_address_1>Hedenstorp 10</street_address_1>
    <street_address_2></street_address_2>
    <postal_code>33292</postal_code>
    <city_name>Gislaved</city_name>
    <residential>1</residential>
    <country_code>SE</country_code>
    <language>sv</language>
  </address_to>
</shipment>

```

Example 5 - Regular FOREIGN shipping query

A regular search delivers a response with one or more shipping alternatives.

They want to send two Finnish men's shoes to the United States in a package of 30x20x10 size.

This XML should work if you only replace the consignor id and key to your own.

```
<?xml version="1.0" encoding="UTF-8"?>
<shipment xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance">
  <value>199.50</value>
  <consignor>
    <id>YOUR_CONSIGNOR_ID</id>
    <key>YOUR_CONSIGNOR_KEY</key>
    <currency>USD</currency>
    <language>en</language>
    <encoding>UTF-8</encoding>
    <api_version>4.7.0</api_version>
  </consignor>
  <commodities>
    <commodity>
      <name>Shoes</name>
      <quantity>2</quantity>
      <taric>6403999600</taric>
      <quantity_units>EA</quantity_units>
      <description>Black leather shoes with laces</description>
      <country_of_manufacture>FI</country_of_manufacture>
      <currency>EUR</currency>
      <weight>2.8</weight>
      <length>30</length>
      <width>13</width>
      <height>8</height>
      <unit_price>899.95</unit_price>
    </commodity>
  </commodities>
  <address_to>
    <street_address_1>101 Main Street</street_address_1>
    <street_address_2></street_address_2>
    <postal_code>12345</postal_code>
    <city_name>Schenectady</city_name>
    <residential>1</residential>
    <country_code>US</country_code>
    <language>en</language>
  </address_to>
</shipment>
```

Response

If the exact same shipping search is performed several times close to each other, a saved search result is displayed instead of a new search.

If the end customer has chosen the agent after the original search, the selected agent is displayed as the closest and only shipping services belonging to the same shipping-company that has the selected agent are displayed.

XML element tag and example of contents	Description
<?xml version="1.0" encoding="UTF-8"?>	XML header
- <shipment>	This is the document element, i.e. the root element of the XML data that describes a shipment that is being searched for.
<server_status>No contact with DHL</server_status>	If we have a temporary server problem, it will be announced here.
<status>ok</status>	Status of API call <u>ok</u> = API call succeeded. <u>warning</u> = API call succeeded with warnings. <u>error</u> = API call failed. <i>Sent in response only for API versions 3.1.0 or greater.</i>
<code>1</code>	Error code 0 = OK 1 = Warning 2 = Error
<warning_message>You need to input all measurements</warning_message>	Warning message with information that may be useful, especially when testing your integration. Data type: STRING Max chars: 500 Required: Yes
<error_message></error_message>	Error message. Data type: STRING Max chars: 500 Required: Yes
<currency>SEK</currency>	Currency. Data type: STRING Max chars: 3 Required: Yes
<id>67887</id>	Fraktjakt's SHIPMENT_ID. This is used in all future references to this particular query result. Data type: INTEGER Required: Yes
<access_code>ABC12345</access_code>	A code for accessing and managing the shipment without previously being logged in. See note 1.
<access_link> https://www.fraktjakt.se/shipments/show/163221?access_code=b6dfc12fc04ec98132da2eb1c1739272cc646ed9 </access_link>	A link to the shipment using the above SHIPMENT_ID and ACCESS_CODE. Data type: STRING Required: Yes
<tracking_code>ABC12345</tracking_code>	A code to track this shipment without

	previously being logged in. See note 2.
<pre><tracking_link> https://www.fraktjakt.se/trace/shipment/b6dfc12fc04 </tracking_link></pre>	<p>A link to track the shipment using the above TRACKING_CODE</p> <p>Data type: STRING Required: Yes</p>
<pre><agent_selection_link> https://www.fraktjakt.se/agents/search_closest </agent_selection_link></pre>	<p>Via this link, the recipient of the freight can be offered to request an agent.</p> <p>Use this link unless <agent_link> can be presented for the various shipping products.</p> <p>It shows the nearest agents for all shipping companies that have services with agents in the result.</p> <p>If the recipient does not choose an agent who belongs to the service that is then selected, it will of course not be the chosen agent either.</p> <p>Does not appear if no_agents is specified.</p> <p>If you have your own Layout Settings in the integration settings, these are used on the page. In this way, the page can be easily adapted to your own page.</p> <p>The following parameters can be added to the link:</p> <p>* redirect - Specifies the layout to be used on the page to select agents. See page 90 for details.</p> <p>* orig - Specifies a URL where to redirect after the customer selects an agent. Do not forget to URL encode.</p> <p>Datatype: STRING</p>
- <shipping_products>	The query result is presented as an XML array of shipping products, sorted in the order defined on your configuration page at Fraktjakt.se.
- <shipping_product>	<p>An individual shipping product, i.e. a service that matches the search criteria that were sent in the query.</p> <p>See page Fel: Det gick inte att hitta referensskällan for the shipping services currently possible to get in the response.</p>
<id>15</id>	<p>The shipping product's ID in i Fraktjakt. This can be used for purchasing the shipping product.</p> <p>Data type: INTEGER Required: Yes</p>
<name>Privat</name>	<p>The name of the shipping product.</p> <p>The name presented on the service</p>

	<p>can be changed in the integration settings in Fraktjakt.</p> <p>Data type: STRING Max tecken: 35 Obligatorisk: Ja</p>
<pre><description>Posten AB Postpaket</description></pre>	<p>Description of the shipping product. This can be presented directly to your customers.</p> <p>Data type: STRING Max chars: 35 Required: Yes</p>
<pre><arrival_time>One to three days, before 10 am guaranteed</arrival_time></pre>	<p>The arrival time; either the number of days from now or the expected date/time of arrival.</p> <p>Data type: STRING Max chars: 64 Required: Yes, but may be empty</p>
<pre><price>159.50</price></pre>	<p>The shipping product's total price without VAT, .but with any selected insurance if the company's settings or input determines that insurance should be used if necessary.</p> <p>In the requested currency.</p> <p>Data type: FLOAT Required: Yes</p>
<pre><tax_class>25.00</tax_class></pre>	<p>The VAT percentage that applies to this shipping product including the VAT on any insurance.</p> <p>Data type: FLOAT Required: Yes</p>
<pre><insurance_fee>75.00</insurance_fee></pre>	<p>The cost of any additional insurance. Included in the price tag if insurance is selected by settings or input data.</p> <p>Displayed in the requested currency.</p> <p>Data type: FLOAT Required: Yes</p>
<pre><insurance_tax_class>0</insurance_tax_class></pre>	<p>The insurance's VAT (percentage). As a rule, always zero.</p> <p>Data type: FLOAT Required: Yes</p>
<pre><to_agent>1</to_agent></pre>	<p>If the product is delivered by an agent or not.</p> <p>Data type: BOOLEAN Required: Yes 0 = false. Selivery directly to the</p>

	<p>customer.</p> <p>1 = true. Delivery to an agent.</p> <p>Only sent in the reply for API-version 3.8.0 or later.</p>
<pre><agent_info>Cityterminalen Stockholm apx 2 km in Stockholm</agent_info></pre>	<p>The shipping agent location that is closest to the receiver's address (does not apply to direct delivery shipping products).</p> <p>If Fraktjakt is missing information about nearest agent it is announced with the text "Agent information is missing".</p> <p>Empty for shipping products that don't use agents.</p> <p>Data type: STRING Max chars: 72 Required: Yes, but may be empty</p>
<pre><agent_link> https://www.fraktjakt.se/agents/search_closest/37748 </agent_link></pre>	<p>A link to a map that shows where the In this link, the consignee of the freight can be offered to choose a representative for the service.</p> <p>If presenting this link together with the price does not work, <agent_selection_link> may work.</p> <p>Does not appear if no_agents is specified.</p> <p>Empty for shipping products that don't use agents.</p> <p>If you have your own Layout Settings in the integration settings, these are used on the page. In this way, the page can be easily adapted to your own page.</p> <p>The following parameters can be added to the link:</p> <ul style="list-style-type: none"> * redirect - Specifies the layout to be used on the page to select agents. See page 90 for details. * orig - Specifies a URL where to redirect after the customer selects an agent. Do not forget to URL encode. <p>Required: Yes, but may be empty</p>
<pre><agent_in_info>Fedex Jönköping apx 1 km in Jönköping</agent_in_info></pre>	<p>The shipping agent location that is closest to the sender's address (does not apply to shipping products with pickup).</p> <p>Is only sent if agents_in is true.</p> <p>Data type: STRING Max chars: 72 Required: No</p>
<pre><agent_in_link> https://www.fraktjakt.se/agents/search_closest/37748</pre>	<p>A link to a map that shows where the shipping agent defined in <agent_in_info> is located.</p>

2?type=8&shipper=4 </agent_in_link>	<p>Is only sent if agents_in is true.</p> <p>If you have your own Layout Settings in the integration settings, these are used on the page. In this way, the page can be easily adapted to your own page.</p> <p>The following parameters can be added to the link:</p> <ul style="list-style-type: none"> * redirect - Specifies the layout to be used on the page to select agents. See page 90 for details. * orig - Specifies a URL where to redirect after the customer selects an agent. Do not forget to URL encode. <p>Data type: STRING Max chars: 256 Required: No</p>
<service_point_locator_api> https://www.fraktjakt.se/agents/service_point_locator </service_point_locator_api>	<p>An example of a suitable call to the Service Point Locator API for this service and the recipient address.</p> <p>Data type: STRING Max characters: 256 Required: No</p>
<shipper>	<p>The shipper delivering the shipping product.</p> <p>This element (and its sub elements) is only created if <shipper_info> has been sent in the initial API call with the value "1".</p> <p>Required: No</p>
<id>4</id>	<p>The shipper's ID in Fraktjakt.</p> <p>Data type: INTEGER Required: No</p>
<name>Bussgods</name>	<p>The shipper's name.</p> <p>Data type: STRING Max chars: 35 Required: No</p>
<logo_url> https://www.fraktjakt.se/images/shippers/4.png </logo_url>	<p>The shipper image logo (as a rectangle) url. In case you want to present it in the search result.</p> <p>Data type: STRING Max chars: 72 Required: No</p>
<logo_cube_url> https://www.fraktjakt.se/images/shippers/4.png </logo_cube_url>	<p>The shipper image logo (as a square) url. In case you want to present it in the search result.</p> <p>Data type: STRING Max chars: 72 Required: No</p>
</shipper>	

</shipping_product>	
</shipping_products>	
</shipment>	

Notes:

1. Put the code in the link:
https://www.fraktjakt.se/shipments/show/SHIPMENT_ID&access_code=ACCESS_COD
For more information about the link, see page 102.
2. Put the code in the link:
https://www.fraktjakt.se/trace/shipment/TRACKING_CODE
For more information about the link, see page 102.

Requery API (cached shipment search)

If you've previously called Query API and got a response containing a shipment_id you can call the Requery API to get a cached response which is much quicker.

XML for Requery API calls

Use the following URI to get a cached response from Fraktjakt via the Requery API:

https://api.fraktjakt.se/fraktjakt/requery_xml

(PROD API – live / production environment)

The XML that you send to the Requery API should include all of the standard fields that were sent to the Query API, plus a shipment_id value that the Query API returned.

You can configure Fraktjakt to send <arrival_time> as the expected shipping time in days or as the expected time of arrival. This setting is done on your configuration page at Fraktjakt.se. The arrival time is calculated based on the shipping product's normal shipping time, the available days for pick-up that you have defined in your settings, and the shipper's pick-up schedule.

XML element tag and example of contents	Description
<?xml version="1.0" encoding="UTF-8"?>	XML header
- <shipment>	This is the document element, i.e. the root element of the XML data that describes a shipment that is being searched for.
- <value>10.50</value>	Value of content in the shipment. Data type: FLOAT Required: No
- <shipper_info>[0 1]</shipper_info>	Include this element if you want to receive extra information about the shipping providers in the response. The response will then contain a <shipper> element containing id, name and logo_url elements.

	Data type: BOOLEAN Required: No Default = 0 (No extra info)
- <consignor>	Information about who's making the call.
See Consignor-element on page 29	
</consignor>	
<shipment_id>167887</shipment_id>	Fraktjakt's shipment ID. This is used in all future references to this particular query result. Data type: INTEGER Required: Yes
</shipment>	

Example

```
<?xml version="1.0" encoding="UTF-8"?>
<shipment>
  <value>10</value>
  <consignor>
    <id>YOUR_CONSIGNOR_ID</id>
    <key>YOUR_CONSIGNOR_KEY</key>
    <currency>SEK</currency>
    <language>en</language>
    <encoding>UTF-8</encoding>
    <api_version>4.7.0</api_version>
  </consignor>
  <shipment_id>167887</shipment_id>
</shipment>
```

Response

Same as from Query API, see above.

If the end-customer has chosen the agent after the original search, the selected agent is displayed as the closest and only shipping services belonging to the same shipping company that has the selected agent are displayed.

Order API - make a shipment ready to be purchased

There are two different types of Order API calls.

In a common integration where you have first submitted a Query API call to present shipping options for the online store customer, the web shop will communicate a second time with Fraktjakt to place an order. Then you use Order API call type 1 (page 61).

In a web site integration, it may be appropriate that this happens after the web store's customer has completed the payment for the goods purchased in the online store.

If you know exactly which shipping service you want to buy, and do not want to call the Query API to get shipping options and shipping prices in advance, then you can use Order API Call Type 2 (see page 62). The advantage is that you create an order with just a single call to Fraktjakt.

Use the following URI to send an order to Fraktjakt via the Order API:

https://api.fraktjakt.se/orders/order_xml (PROD API – live production environment)

The setting called 'Incoming shipment connection' in the integration's settings in Fraktjakt indicates whether an email from Fraktjakt to the Web Administrator should be created for each call to this API. This way, the administrator can easily manage the administration if it rarely comes calls.

Calltype 1 – Create an order from a previously created shipment

Usually the previously created shipment was created using the Query API.

The incoming XML contains the element, <shipment_id>, which identifies a previously created shipment from the Query-API-call. A search via the Query API is saved for five days. For another five days after that, existing <shipment_id> can be used, but a new shipment is created instead.

Do not use the element <parcels>, for this call type. The previously created original shipment (identified with <shipment_id>) should already have dimensions (length, width and height) for each package in it. Either from entering the <parcels> element or from the packages being calculated based on specified commodities.

However, you may need to submit the `<commodities>` element if the package contents not were specified when doing the shipment query. If the content is already known, the new `<commodities>` element is ignored. So do not send with commodities if you have already sent it in the Query call.

TIP: If you send a call to Order API containing a `shipment_id` from a previous paid shipment, a new shipment will be created with a new `shipment_id`. It will look the same as the previous shipment. This makes it easy to do multiple purchases of identical shipments with the same addresses. This is a quick way to copy a shipment.

Order API divides a shipment in to multiple shipments when the original shipment contains more than one package and the shipping service is one where the receiver collects the shipment at a an agent.

It's possible to send addresses with the elements, `<address_from>` and `<address_to>`, but these will not be used here. The address will always be from the given shipment.

Call type 2 – Create and order directly without a previously created shipment.

This call type allows you to directly place a shipping-order in Fraktjakt.'s shopping cart.

The incoming XML does not contain any `<shipment_id>` element. Instead, the `<commdoties>` element is used and if you want to you may use the `<parcels>` element. Therefore, `<commodities>` needed and `<parcels>` can be submitted if desired.

If you only submit `<commodities>`, the packages are calculated as in the Query API, see page 41, if the weights are specified.

If there are `<parcels>` but not `<commodities>`, they must be completed later in the shopping cart.

The alternatives for the `<shipping_product_id>` element can be found by calling https://www.fraktjakt.se/shipping_products/xml_list.

Address management:

- Sender address – If the sender address is sent with the element, `<address_from>`, it will be used, otherwise the registered delivery address will be used.

Receiver address - If the receiver address is sent with the element, `<address_to>`, it will be used as the receiver address. Otherwise the sender address will be used. In most cases this is not desirable.

XML for Order API calls

XML element tag and example of contents	Description
<code><?xml version="1.0" encoding="UTF-8"?></code>	XML header
- <code><OrderSpecification></code>	This is the document element, i.e. the element that contains the entire order that is being sent to Fraktjakt. It contains one shipment. Required: Yes
<code><value>10.2</value></code>	[Currently not used] This is the value of all items in the order, i.e. the reimbursement amount in case of loss in transport. It should normally be the same as the shipment's value (See below.). Data type: FLOAT Required: No
- <code><consignor></code>	Information about the consignor, i.e. the webshop or other entity that is sending the order.
See consignor element on page 16.	
<code></consignor></code>	
<code><referred_consignor></code>	Use this element if the shipment order will be finished and paid for by someone other than the consignor. Enter the name of the The response will then also include the element <code><payment_link></code> that will link to the payment-page. Data type: INT Required: No
<code><id>12121</id></code>	The payer's customer ID in Fraktjakt. Data type: INT Required: Yes, if referred_consignor is used
<code><key>12121212121212121212121212121</key></code>	The payer's customer ID in Fraktjakt. Data type: STRING Max chars: 64 Required: Yes, if referred_consignor is used
<code></referred_consignor></code>	
<code><shipment_id>67887</shipment_id></code>	Fraktjakt's shipment ID This must be an ID that you have received in a response from the Query API. If you have already used

	<p>the same shipment ID in a previous call to the Order API, Fraktjakt will create a new copy of the existing shipment record with a new shipment ID (but with the same shipment details).</p> <p>For call type 2 this field should not be used.</p> <p>It is this field that determines whether to use call type 1 or 2.</p> <p>Data type: INT Required: Yes, for call typ 1 No for call type 2.</p>
- <callback_url> https://my.se </callback_url>	<p>The URL of the server that will receive the call from Fraktjakt's webhook.</p> <p>If the tag is specified, it overrides the integration's webhook setting.</p> <p>For more information, see the chapter "Webhook callback" and the document for Fraktjakt's webhook.</p> <p>Tip: In the tag, it is possible to send in a URL with its own security parameter, which is then used in the call from Fraktjakt. For example? Key = abcdef123456</p>
- <insure_default>[0 1]</insure_default>	<p>Should the freight be additionally insured if the shipping company's basic insurance does not cover the value of the goods.</p> <p>If the tag is specified, it overrides your company's setting regarding insurance.</p> <p>Data type: BOOLEAN Required: No</p>
<shipping_product_id>15</shipping_product_id> <i>(Should be one of the shipping product IDs from the response to a previous query_xml call.)</i>	<p>The shipping product's ID in Fraktjakt.</p> <p>For call type 1: This must be an ID that you have received in a response from the Query API.</p> <p>Call Type 2 uses a number from the list of products.</p> <p>The id can be found with the call https://www.fraktjakt.se/shipping_products/xml_list</p> <p>If an incorrect number is entered, the shipment must be corrected in Fraktijakt.</p> <p>Data type: INT Required: Yes</p>
<agent_id>1236</agent_id>	<p>If the end customer chooses a representative, the selected</p>

	<p>representative is stated here.</p> <p>See further on page 112, Customers agent selection</p> <p>Data type: INT Required: NO</p>
<reference>Order id 191</reference>	<p>Free text that refers to the order in Fraktjakt. This element may contain your own system's order ID, an identifying text, or some other. This text will appear on your shipping labels.</p> <p>Data type: STRING Max chars: 50 Required: No</p>
<p><export_reason>SALE</export_reason></p> <p>Valid values:</p> <ul style="list-style-type: none"> • SALE – Any shipment being sold or sent free of charge, from one business entity to another, for the purpose of being re-sold or otherwise consumed in the course of carrying out its business. • GIFT – Any shipment containing articles to be given as an unsolicited gift, from one private individual to another private individual, which are not being shipped by a business nor consigned to a business. • SAMPLE – Any shipment containing articles which are sent free of charge that have been marked or mutilated or otherwise made unsuitable for sale or use except as commercial samples, being shipped with the view of soliciting an order or as a mock-up to furthering an existing order from the foreign entity. • RETURN – The cargo that was bought from the receiver • REPAIR – Articles being shipped for repair or articles previously exported for repairs which are being returned following the repair. • PERSONAL EFFECTS – Any shipment containing used personal articles such as unaccompanied baggage and household goods being shipped for relocation. 	<p><i>Reason for export</i> tells the purpose of the shipment.</p> <p>Data type: STRING Max chars: 16 Default: SALE Required: Recommended for International shipments</p>
- <no_agents>[0 1]</no_agents>	<p>If you want to know the nearest agent or not. If you select to not show them you will receive a significantly faster reply, but give less service to your customers.</p> <p>Data type: BOOLEAN Required: No Default = 0 (show agents)</p>
<commodities>	<p>The shipments content.</p> <p>The different commodities the shipment.</p> <p>For call type 1, specify, if it is not previously specified for the freight in the Query API. That is recommended</p>

	<p>to do.</p> <p>Must be entered for call type 2. If not specified but parcels are, Default Commodity Template is used if it is also specified to be used in searches.</p>
See the Commodity chapter on page 34.	
</commodities>	
- <parcels>	<p>Packages to be sent.</p> <p>Always omitted at call type 1</p> <p>May be omitted at call type 2 if commodities are given by weight and for (shipping_product_id = 137-141). See Note 3.</p>
- <parcels>	<p>The parcels that will be sent.</p> <p>Can be specified if commodities are not known, or if you have really decided how big the packages should be and do not want to use Fraktjakt's package calculations.</p>
Se the Parcel-elemnts on page 40	
</parcels>	
- <address_from>	<p>Sender address</p> <p>Required: Yes for Call type 2. Is not used in Call type 1.</p> <p>If not entered the shipment address from the settings will be used.</p>
See address element on page 30	
</address_from>	
- <address_to>	<p>Receiver address</p> <p>Alternative name is 'address'</p> <p>Required: yes, with call type 2</p> <p>Is not used in Call type 1.</p>
See address element on page 30	Know that <language> can be used
</address_to>	
- <sender>	<p>The Sender</p> <p>Required: No, but for receiver paid it should be given.</p>
<company_from>Köp och sälj AB</company_from>	Sending company name

	Data type: STRING Max chars: 32 Required: No
<name_from> Sven Andersson </name_from>	Name of sending contact Data type:STRING Max chars: 32 Required: No
<telephone_from> 036190220 </telephone_from>	Senders telephone number Data type: STRING Max chars: 32 Required: No
<email_from> sven@webbutikenab.se </email_from>	Senders email address Data type:: STRING Max chars: 64 Required: No, <i>and is not used if the tagg <sender_email> is given.</i>
<eori>SE5561234711</eori>	Senders EORI-nummer for export out off EU. Data type: STRING Max chars: 32 Required: No. If it is not specified, the value in Fraktjakt is used.
<rex>My REX</rex>	Sender's REX number for country of origin declaration. Data type: STRING Max chars: 32 Required: No. If it is not specified, the value in Fraktjakt is used.
<voec>123456789</voec>	Sender's VOEC number for export to Norway. Data type: STRING Max chars: 32 Required: No. If it is not specified, the value in Fraktjakt is used.
<gb_vat>My REX1234567890</gb_vat>	Sender's UK VAT number for export to Great Britain. Data type: STRING Max chars: 10 Required: No. If it is not specified, the value in Fraktjakt is used.
<mva_num>123456789MVA</mva_num>	Sender's norwegian MVA, vat number, for export to Norway Datatyp: STRING Max tecken: 12 Required: No. If it is not specified, the value in Fraktjakt is used.

<code><ioss>SE1234569</ioss></code>	<p>Senders IOSS nummer for import to EU.</p> <p>Datotyp: STRING</p> <p>Max tecken: 32</p> <p>Obligatorisk: <i>No. If it is not specified, the value in Fraktjakt is used.</i></p>
<code><tax_id>1234567890</tax_id></code>	<p>The sender's tax identification number (TIN).</p> <p>Data type: STRING</p> <p>Max chars: 18</p> <p>Required: No, but used in shipments from or to EU.</p>
<code></sender></code>	
<code>- <recipient></code>	<p>The receiver of the shipment.</p> <p>Required: No</p> <p>If not given for shipments the receiver pays, the information from the integration is used.</p>
<code><company_to>Buy and Sell Ltd</company_to></code>	<p>The recipient's company name (if the recipient is a company).</p> <p>Data type: STRING</p> <p>Max chars: 32</p> <p>Required: No</p>
<code><name_to>Sven Andersson</name_to></code>	<p>The recipient's name (if the recipient is not a company).</p> <p>Data type: STRING</p> <p>Max chars: 32</p> <p>Required: No</p>
<code><telephone_to>0730710220</telephone_to></code>	<p>The recipient's telephone number.</p> <p>Must be from the same country as the recipient. Otherwise it will not work.</p> <p>Data type: STRING</p> <p>Max chars: 32</p> <p>Required: No – but we recommend you to send it</p> <p>See Note 7 for how the notification is set.</p>
<code><mobile_to>0730710220</mobile_to></code>	<p>The recipient's cell phone number.</p> <p>Must be from the same country as the recipient. Otherwise it will not work.</p> <p>Data type: STRING</p> <p>Max chars: 32</p> <p>Required: No</p> <p>See Note 7 for how the notification is set.</p>
<code><email_to>sven@webbutikenab.se</email_to></code>	<p>The recipient's e-mail address.</p>

	<p>Data type: STRING Max chars: 64 Required: Should be entered so that email notification works, including the tailor made notification from Fraktjakt.</p> <p>See Note 7 for how the notification is set.</p>
<eori>SE5561234711</eori>	<p>Receivers EORI-nummer for export out off EU.</p> <p>Data type: STRING Max chars: 32 Required: No. If it is not specified, the value in Fraktjakt is used.</p>
<tax_id>1234567890</tax_id>	<p>The recipient's tax identification number (TIN).</p> <p>Data type: STRING Max chars: 18 Required: No, but used in shipments from or to EU.</p>
<message_to>Hello Customer</message_to>	<p>A message to the receiver. Can be added in the Notification email to your customers with a special tag in the layout branding settings.</p>
</recipient>	
<sender_email>info@example.info</sender_email>	<p>[Only applies to pre-paid shipping] The e-mail address of the person who will be handling the shipment, if not the consignor.</p> <p>This will generate a sender_email_link element in the reply and an email to that address when the order is paid.</p> <p>Data type: STRING Max chars: 64 Required: No</p>
<booking>	<p>Booking information. Shipments are always booked from the sender's address. If no booking address is included, Fraktjakt will use the address information that you entered on the integrations configuration page at Fraktjakt.se.</p> <p>Required: No</p>
<driving_instruction>Here and there</driving_instruction>	<p>Instructions on how to find the place where the shipment is to be picked up for delivery.</p> <p>Data type: String Required: No</p>
<pickup_date>2019-12-10</pickup_date>	<p>The date when the shipment is to be</p>

	picked up. This must be a date in the future. Data type: Date (YYYY-MM-DD) Required: No
</booking>	
</OrderSpecification>	

Notes:

1. A shipment contains one or more parcels.
2. En shipment contains one or more commodity types.
3. If you specify dimensions 120 or 60 on length and 80 on width, Fraktjakt interprets the input package as a pallet or half pallet and searches for shipping products for pallets. If the height is below 125 cm, the pallet will be stackable. Higher heights provide non-stackable pallets.
4. At least one contact method must be included for the shipping recipient. It is recommended that the recipient's e-mail address is always included.
5. https://en.wikipedia.org/wiki/ISO_3166-1_alpha-2
6. <user_notes> will not be supported in the future. Put the text in 'driving_instructions' instead.
7. The notification to the recipient is determined as follows:
In the first place, SMS notification is used, if the selected shipping product supports it. The selected number is <telephone_to> if it is a mobile number or <mobile_to> if it is entered. Many platforms only support one field for the telephone and it should then be sent into <telephone_to>. Then Fraktjakt takes care of the logic based on what the user enters for the type of number.
Alternatively, as second prio, telephone notification is used, if the selected shipping product supports it, where you call the recipient. Provided <telephone_to> is specified.
Third, email notification is selected if supported by the the selected shipping product and if <email_to> is submitted.
Ultimately, letter notification is used if the selected shipping product supports it.

Once Fraktjakt receives an order via the Order API, the order is "completed" internally in Fraktjakt. This entails that the order is associated with your Fraktjakt account, pending payment. The shipment can be found in the shopping cart.

Example – Call type 1 Domestic

[Create an order from an earlier created shipment \(usually created through Query API \).](#)

In this case, the content of the shipping has been submitted to the Query API for calculating which packages are used and id = 87566 returned. See Query API Example 1 page 47. Bussgods Sverigefrakt is selected.

This XML will work if you replace consignor id and key to your own integration values, as well as entering an existing *shipment_id*.

```
<?xml version="1.0" encoding="UTF-8"?>
<OrderSpecification>
  <consignor>
    <id>YOUR_CONSIGNOR_ID</id>
    <key>YOUR_CONSIGNOR_KEY</key>
    <currency>SEK</currency>
    <language>en</language>
    <encoding>utf-8</encoding>
    <api_version>4.7.0</api_version>
  </consignor>
  <referred_consignor>
    <id>REFERRED_CONSIGNOR_ID</id>
    <key>REFERRED_CONSIGNOR_KEY</key>
  </referred_consignor>
  <shipment_id>87566</shipment_id>
  <shipping_product_id>25</shipping_product_id>
  <reference>My brothers shoes</reference>
  <recipient>
    <name_to>Olle Klint</name_to>
    <company_to>Hanson & Greve AB</company_to>
    <telephone_to>036190220</telephone_to>
  </recipient>
  <booking>
    <pickup_date>2019-12-10</pickup_date>
    <driving_instruction>Drive up on the alley to your right.</driving_instruction>
    <user_notes>Port code 1112</user_notes>
  </booking>
</OrderSpecification>
```

Example – Call type 1 International

Create an order from an earlier created INTERNATIONAL shipment (usually created through Query API).

In this case, the query API has only specified the package and not the content. The weight was 2.8 kg.

This XML will work if you replace consignor id and key to your own integration values, as well as entering an existing *shipment_id*.

```
<?xml version="1.0" encoding="UTF-8"?>
<OrderSpecification>
  <consignor>
    <id>YOUR_CONSIGNOR_ID</id>
    <key>YOUR_CONSIGNOR_KEY</key>
    <currency>EUR</currency>
    <language>en</language>
    <encoding>utf-8</encoding>
```

```

    <api_version>4.7.0</api_version>
  </consignor>
  <shipment_id>694556</shipment_id>
  <shipping_product_id>119</shipping_product_id>
  <reference>My brothers shoes</reference>
  <export_reason>GIFT</export_reason>
  <commodities>
    <commodity>
      <name>Skor</name>
      <quantity>2</quantity>
      <taric>0123456789</taric>
      <quantity_units>EA</quantity_units>
      <description>Black leather shoes with laces</description>
      <country_of_manufacture>FI</country_of_manufacture>
      <weight>2.8</weight>
      <unit_price>99.99</unit_price>
      <currency>EUR</currency>
    </commodity>
  </commodities>
  <recipient>
    <name_to>Olle Klint</name_to>
    <company_to>Hanson & Greve AB</company_to>
    <telephone_to>036190220</telephone_to>
  </recipient>
  <booking>
    <pickup_date>2019-12-10</pickup_date>
    <driving_instruction>Drive up on the alley to your right.</driving_instruction>
    <user_notes>Port code 1112</user_notes>
  </booking>
</OrderSpecification>

```

Example – Call type 2 Domestic

Create an order directly (without having created a shipment in an earlier query call).

This XML will work if your replace consignor id and key to your own integration values, as well as entering an existing shipment_id.

```

<?xml version="1.0" encoding="UTF-8"?>
<OrderSpecification>
  <consignor>
    <id>YOUR_CONSIGNOR_ID</id>
    <key>YOUR_CONSIGNOR_KEY</key>
    <currency>SEK</currency>
    <language>en</language>
    <encoding>utf-8</encoding>
    <api_version>4.7.0</api_version>
  </consignor>
  <shipping_product_id>25</shipping_product_id>
  <reference>Ordernr 123456</reference>
  <commodities>
    <commodity>
      <name>Jeans</name>
      <quantity>1</quantity>
      <description>Bluejeans size 33x32</description>
      <weight>1.3</weight>
    </commodity>
  </commodities>
  <address_from>
    <street_address_1>Gjuterigatan 9</street_address_1>
    <street_address_2></street_address_2>
    <postal_code>55318</postal_code>
  </address_from>

```

```

    <city_name>Jönköping</city_name>
    <residential>0</residential>
    <country_code>SE</country_code>
  </address_from>
  <address_to>
    <street_address_1>Hedenstorp 10</street_address_1>
    <street_address_2></street_address_2>
    <postal_code>33292</postal_code>
    <city_name>Gislaved</city_name>
    <residential>1</residential>
    <country_code>SE</country_code>
    <language>sv</language>
  </address_to>
  <recipient>
    <name_to>Olle Klint</name_to>
    <telephone_to>036190220</telephone_to>
  </recipient>
  <booking>
    <pickup_date>2019-04-06</pickup_date>
    <driving_instruction>Drive up on the alley to your right.</driving_instruction>
    <user_notes>Port code 1112</user_notes>
  </booking>
</OrderSpecification>

```

Example – Call type 2 International

Create an order directly (without having created a shipment in an earlier query call).

This XML will work if your replace consignor id and key to your own integration values, as well as entering an existing shipment_id.

```

<?xml version="1.0" encoding="UTF-8"?>
<OrderSpecification>
  <consignor>
    <id>YOUR_CONSIGNOR_ID</id>
    <key>YOUR_CONSIGNOR_KEY</key>
    <currency>SEK</currency>
    <language>en</language>
    <encoding>utf-8</encoding>
    <api_version>4.7.0</api_version>
  </consignor>
  <shipping_product_id>119</shipping_product_id>
  <reference>Ordernr 123456</reference>
  <export_reason>GIFT</export_reason>
  <commodities>
    <commodity>
      <name>Jeans</name>
      <quantity>2</quantity>
      <taric>0123456789</taric>
      <quantity_units>EA</quantity_units>
      <description>Levis blue jeans, size 33x32</description>
      <country_of_manufacture>FI</country_of_manufacture>
      <weight>1.5</weight>
      <unit_price>99.99</unit_price>
      <currency>EUR</currency>
    </commodity>
  </commodities>
  <address_from>
    <street_address_1>Gjuterigatan 9</street_address_1>
    <street_address_2></street_address_2>
    <postal_code>55318</postal_code>
    <city_name>Jönköping</city_name>
  </address_from>

```

```
<residential>0</residential>
<country_code>SE</country_code>
</address_from>
<address_to>
  <street_address_1>101 Main Street</street_address_1>
  <street_address_2></street_address_2>
  <postal_code>12345</postal_code>
  <city_name>Schenectady</city_name>
  <residential>1</residential>
  <country_code>US</country_code>
  <country_subdivision_code>NY</country_subdivision_code>
  <language>en</language>
</address_to>
<recipient>
  <name_to>Jack Martins</name_to>
  <telephone_to>(732) 555-1212</telephone_to>
</recipient>
<booking>
  <pickup_date>2019-04-06</pickup_date>
  <driving_instruction>Drive up on the alley to your right.</driving_instruction>
  <user_notes>Port code 1112</user_notes>
</booking>
</OrderSpecification>
```

Response

XML element tag and example of contents	Description
<code><?xml version="1.0" encoding="UTF-8"?></code>	
- <code><result></code>	
<code><server_status>No contact with DHL</server_status></code>	If we have a temporary server problem, it will be announced here.
<code><status>ok</status></code>	<p>Status of API call</p> <p><u>ok</u> = API call succeeded. <u>warning</u> = API call succeeded with warnings. <u>error</u> = API call failed.</p> <p><i>Sent in response only for API versions 3.1.0 or greater.</i></p>
<code><code>1</code></code>	<p>0=OK 1=warning 2=Error</p>
<code><warning_message>You need to define all measurements</warning_message></code>	<p>Warning message with information that may be useful, especially when testing your integration.</p> <p>Data type: STRING Max chars: 500 Required: Yes</p>
<code><error_message></error_message></code>	<p>Error message.</p> <p>Data type: STRING Max chars: 500 Required: Yes</p>
<code><shipment_id>45654</shipment_id></code>	<p>Fraktjakt's shipment ID, see note 1</p> <p>Data type: INT Required: Yes</p>
<code><access_code>ABC12345</access_code></code>	A code for accessing and managing the shipment without previously being logged in. (note 2)
<code><access_link></code> https://www.fraktjakt.se/shipments/show/163221?access_code=b6dfc12fc04ec98132da2eb1c1739272cc	A link to the shipment using the above SHIPMENT_ID and ACCESS_CODE.

646ed9 </access_link>	Data type: STRING Required: Yes
<return_link> https://www.fraktjakt.se/shipments/xml_create_return/163221?access_code=b6dfc12fc04ec98132da2eb1c1739272cc646ed9 </return_link>	Available from version 4.2.0. A link to create a return shipping based on this shipping. If there is a pre-selected return product, it will be used, otherwise the best according to the settings for Automatic selection of shipping service. The answer to calling this link will look the same as the answer from the Shipment API except for a new <return_link> tag.
<cancel_link> https://www.fraktjakt.se/shipments/xml_cancel/163221?access_code=b6dfc12fc04ec98132da2eb1c1739272cc646ed9 </cancel_link>	Available from version 4.3.0. A link to delete the shipping created in this call. If the shipping is not paid, it will be deleted immediately, otherwise it will be annulled. In case of annulation, not all shipping companies allow you to unbook the pickup automatically. Then you yourself have to make a annulation by phone directly to the shipping company.
<tracking_code>ABC12345</tracking_code>	A code to track this shipment without previously being logged in. See note 2.
<tracking_link> https://www.fraktjakt.se/trace/shipment/b6dfc12fc04 </tracking_link>	A link to track the shipment using the above TRACKING_CODE Data type: STRING Required: Yes
——<order_id>67887</order_id>	Fraktjakt's order ID Data type: INT Required: Yes Deprecated and will be removed
<amount>1066.33</amount>	The cost of the shipment, excluding VAT. Data type: FLOAT Required: Yes
<currency>SEK</currency>	Currency corresponding to the shipment cost. Data type: STRING Max chars: 3 Required: Yes

<pre><payment_link></pre> <p>https://www.fraktjakt.se/orders/finish/1631?code=b6dfc12fc04ec98132da2eb1c1739272cc646ed9</p> <pre></payment_link></pre>	<p>This element contains a link to the payment page If a referred_consignor element is submitted for the paying user.</p> <p>Data type: STRING Required: No</p>
<pre><sender_email_link></pre> <p>https://www.fraktjakt.se/orders/confirmation/1671?key=cabec52762637412652e5023deb5e5c2</p> <pre></sender_email_link></pre>	<p>A link to handle the shipment after the order is paid. The element is added if sender_email is sent in the order-call.</p> <p>Data type: STRING Required: No</p>
<pre><agent_info>Cityterminalen Stockholm apx 2 km in Stockholm</agent_info></pre>	<p>The shipping agent location that is closest to the receiver's address (does not apply to direct delivery shipping products).</p> <p>If Fraktjakt is missing information about nearest agent it is announced with the text "Agent information is missing".</p> <p>Empty for shipping products that don't use agents.</p> <p>Data type: STRING Max chars: 72 Required: Yes, but may be empty</p>
<pre><agent_link></pre> <p>https://www.fraktjakt.se/agents/search_closest/37748</p> <pre></agent_link></pre>	<p>A link to a map that shows where the In this link, the consignee of the freight can be offered to choose a representative for the service.</p> <p>Valid as long as the shipment is not purchased in Fraktjakt.</p> <p>Empty for shipping products that don't use agents.</p> <p>If you have your own Layout Settings in the integration settings, these are used on the page. In this way, the page can be easily adapted to your own page.</p> <p>The following parameters can be added to the link:</p> <ul style="list-style-type: none"> * redirect - Specifies the layout to be used on the page to select agents. See page 90 for details. * orig - Specifies a URL where to redirect after the customer selects an agent. Do not forget to URL encode. <p>Required: Yes, but may be empty</p>
<pre><service_point_locator_api></pre> <p>https://www.fraktjakt.se/agents/service_point_locator</p> <pre></service_point_locator_api></pre>	<p>An example of a suitable call to the Service Point Locator API for this service and the recipient address.</p>

	Data type: STRING Max characters: 256 Required: No
</result>	

Notes:

1. This is not necessarily the same shipment_id that was returned when creating the order, since that SHIPMENT_ID can be reused and changed if the shipment is split into several new shipments.

This SHIPMENT_ID can be used to track the shipment with one of these links:

(just substitute SHIPMENT_ID with the actual shipment_id number that was returned)

https://www.fraktjakt.se/trace/list_shipment/SHIPMENT_ID

https://www.fraktjakt.se/trace/xml_trace?

[consignor_id=YOUR_CONSIGNOR_ID&consignor_key=YOUR_CONSIGNOR_KEY&shipment_id=SHIPMENT_ID](https://www.fraktjakt.se/trace/xml_trace?consignor_id=YOUR_CONSIGNOR_ID&consignor_key=YOUR_CONSIGNOR_KEY&shipment_id=SHIPMENT_ID)

2. Put the code in link:

https://www.fraktjakt.se/shipments/show/SHIPMENT_ID&access_code=ACCESS_CODE

For more information about the link, see page 102

3. Put the code in the link:

https://www.fraktjakt.se/trace/shipment/TRACKING_CODE

For more information about the link, see page 102.

Shipment API

(Merchant controlled)

The Shipment API is used to create a preliminary shipment in Fraktjakt. No price search is performed by the call, so the integration administrator must later on complete the price search and shipment purchase after logging in to Fraktjakt.

In Fraktjakt it is possible to set whether or not an automatic search should be done. It is also possible to set how the search should then choose freight service.

The Shipment API is usually used by an integration to make a call to Fraktjakt after the webshop's customer has completed the payment for the goods purchased in the online store. It's the integration administrator who sends out the call by clicking a link in the online store's admin console.

The shipment must have Sweden as either sender country or recipient country, or both.

Prepare a shipment in Frakjakt by calling Shipment API with the following URL:

https://api.fraktjakt.se/shipments/shipment_xml (PROD API – production environment)

After the call to this API, Fraktjakt starts to automatically find the best freight based on the information in the call. How that search should work and what result to find can be set in the settings for the Integration in Fraktjakt. This way, your freight handling can go extremely fast and with few manual interventions.

If you want help with the settings, do not hesitate to contact Fraktjakt's customer service, see sid 123.

The shipment can be found using the following URL:

https://www.fraktjakt.se/shipments/show/SHIPMENT_ID&access_code=ACCESS_CODE

The setting for email from Fraktjakt called 'Incoming shipment connection' in the integration's settings in Fraktjakt determines if a mail from Fraktjakt to the integration admin should be sent for every call to this API.

Call

XML element tag and example of contents	Description
---	-------------

<?xml version="1.0" encoding="UTF-8"?>	XML-header
- <CreateShipment>	The call to Fraktjakt Required: Yes
- <consignor>	Information about who's making the call.
See Consignor-element on page 29	
</consignor>	
- <referred_consignor>	Use this element if someone other than the consignor will be paying for the transportation. The element should contain the name of the person who will be handling the shipment. The response will then also include the element, <payment_link>, that will link to the payment-page. Data type: INT
<id>REFERRED_CONSIGNOR_ID</id>	The payer's customer ID in Fraktjakt. Data type: INT Required: Yes, if referred_consignor is used
<key>REFERRED_CONSIGNOR_KEY</key>	The payer's customer key in Fraktjakt. Data type: STRING Max chars: 64 Required: Yes, if referred_consignor is used
</referred_consignor>	
<shipping_product_id>15</shipping_product_id>	Favorite service ID in Fraktjakt. Used instead of Favorite service in the integration setting for Automatic selection of shipping service. The ID can be found with the call https://www.fraktjakt.se/shipping_products/xml_list See page Fel: Det gick inte att hitta referenskällan The difference between sending with shipping_product_id in this call instead of using Shipment API is that this is the selected service, not the favorite service that can be changed depending on the settings for the Integration. Datatyp: INT

	Obligatorisk: No
<code><agent_id>1236</agent_id></code>	<p>If the end customer chooses a representative, the selected representative is stated here.</p> <p>See further on page 112</p> <p>Data type: INT Required: No</p>
- <code><callback_url>https://my.se</callback_url></code>	<p>The URL of the server that will receive the call from Fraktjakt's webhook.</p> <p>If the tag is specified, it overrides the integration's webhook setting.</p> <p>For more information, see the chapter "Webhook callback" and the document for Fraktjakt's webhook.</p> <p>Tip: In the tag, it is possible to send in a URL with its own security parameter, which is then used in the call from Fraktjakt. For example? Key = abcdef123456</p>
- <code><insure_default>[0 1]</insure_default></code>	<p>Should the freight be additionally insured if the shipping company's basic insurance does not cover the value of the goods.</p> <p>If the tag is specified, it overrides your company's setting regarding insurance.</p> <p>Data type: BOOLEAN Required: No</p>
<code><reference>Orderid 1236</reference></code>	<p>Individual field for referencing the shipment in Fraktjakt. It could be your own order_id, help text or any value you choose. Also visible on all shipping notes.</p> <p>Data type: STRING Max chars: 50</p> <p>Only digits, letters A-Z and space are allowed..</p> <p>Required: No</p>
<code><price_sort>[0 1]</price_sort></code>	<p>To override the settings for how searches will be sorted.</p> <p>1 if the search is to be sorted by the cheapest price. If 0 the search is sorted on delivery time instead.</p> <p>Datatype: BOOLEAN Required: No Default = The settings for the integration</p>

- <address_from>	Sender address Required: No If not entered the address from the settings will be used.
See address element on page 30	
</address_from>	
- <address_to>	Receiver address Required: No If not entered, the sender address will be used. Which is usually not something you want.
See address element on page 30	Know that <language> can be used
</address_to>	
<export_reason>SALE</export_reason> Valid values: <ul style="list-style-type: none"> • SALE – Any shipment being sold or sent free of charge, from one business entity to another, for the purpose of being re-sold or otherwise consumed in the course of carrying out its business. • GIFT – Any shipment containing articles to be given as an unsolicited gift, from one private individual to another private individual, which are not being shipped by a business nor consigned to a business. • SAMPLE – Any shipment containing articles which are sent free of charge that have been marked or mutilated or otherwise made unsuitable for sale or use except as commercial samples, being shipped with the view of soliciting an order or as a mock-up to furthering an existing order from the foreign entity. • RETURN – The cargo that was bought from the receiver • REPAIR – Articles being shipped for repair or articles previously exported for repairs which are being returned following the repair. • PERSONAL EFFECTS – Any shipment containing used personal articles such as unaccompanied baggage and household goods being shipped for relocation. 	Reason for export tells the purpose of the shipment. Data type: STRING Max chars: 16 Default: SALE Required: International shipments
<sender_email>info@example.info</sender_email>	[Only applies to pre-paid shipping] The e-mail address of the person who will be handling the shipment, if not the consignor. This will generate a sender_email_link element in the reply and an email to that address when the order is paid.

	Data type: STRING Max chars: 64 Required: No
- <sender>	The Sender Required: No, but for receiver paid it should be given.
<company_from> Köp och sälj AB </company_from>	Sending company name Data type: STRING Max chars: 32 Required: No
<name_from> Sven Andersson </name_from>	Name of sending contact Data type: STRING Max chars: 32 Required: No
<telephone_from> 036190220 </telephone_from>	Senders telephone number Data type: STRING Max chars: 32 Required: No
<email_from> sven@webbutikenab.se </email_from>	Senders email address Data type: STRING Max chars: 64 Required: No, and is not used if the tagg <sender_email> is given.
<eori> SE5561234711 </eori>	Senders EORI-nummer for export out off EU. Data type: STRING Max chars: 32 Required: No. If it is not specified, the value in Fraktjakt is used.
<rex> My REX </rex>	Sender's REX number for country of origin declaration. Data type: STRING Max chars: 32 Required: No. If it is not specified, the value in Fraktjakt is used.
<voec> 123456789 </voec>	Sender's VOEC number for export to Norway. Data type: STRING Max chars: 32 Required: No. If it is not specified, the value in Fraktjakt is used.
<gb_vat> My REX1234567890 </gb_vat>	Sender's UK VAT number for export to Great Britain. Data type: STRING Max chars: 10

	Required: No. If it is not specified, the value in Fraktjakt is used.
<mva_num>123456789MVA</mva_num>	Sender's norwegian MVA, vat number, for export to Norway Datatyp: STRING Max tecken: 12 Required: No. If it is not specified, the value in Fraktjakt is used.
<ioss>SE1234569</ioss>	Senders IOSS nummer for export from EU. Datatyp: STRING Max tecken: 32 Obligatorisk: <i>No. If it is not specified, the value in Fraktjakt is used.</i>
<tax_id>1234567890</tax_id>	The sender's tax identification number (TIN). Data type: STRING Max chars: 18 Required: No, but used in shipments from or to EU.
</sender>	
- <recipient>	The receiver of the shipment. Required: Yes
<company_to>Köp och sälj AB</company_to>	The recipient's company name (if the recipient is a company). Data type: STRING Max chars: 32 Required: No
<name_to>Sven Andersson</name_to>	The recipient's name (if the recipient is not a company). Data type: STRING Max chars: 32 Required: No
<telephone_to>036190220</telephone_to>	The recipient's telephone number. Must be from the same country as the recipient. Otherwise it will not work. Data type: STRING Max chars: 32 Required: No - but should be specified for Email Notification to work as intended, including the customized notification from Fraktjakt. See Note 1
<mobile_to>0709999999</mobile_to>	The recipient's cell phone number. Must be from the same country as

	<p>the recipient. Otherwise it will not work.</p> <p>Data type: STRING Max chars: 32 Required: No See Note 1</p>
<code><email_to>sven@webbutikenab.se</email_to></code>	<p>The recipient's e-mail address.</p> <p>Data type: STRING Max chars: 64 Required: No, but it is highly recommended that it be sent in so that email notification works, including the tailor made notification from Fraktjakt.</p>
<code><eori>SE5561234711</eori></code>	<p>Receivers EORI-nummer for export out off EU.</p> <p>Data type: STRING Max chars: 32 Required: No. If it is not specified, the value in Fraktjakt is used.</p>
<code><tax_id>1234567890</tax_id></code>	<p>The recipient's tax identification number (TIN).</p> <p>Data type: STRING Max chars: 18 Required: No, but used in shipments from or to EU.</p>
<code><message_to>Hello Customer</message_to></code>	<p>A message to the receiver. Can be added in the Notification email to your customers with a special tag in the layout branding settings.</p>
<code></recipient></code>	
<code><ommodities></code>	<p>The content in the shipment The commodity types in the shipment. At least one item must be entered. MinOccur = 1</p>
See Commodity-element on page 34	<p>If at least the total weight is specified on each Commodity, Fraktjakt can prepare the shipment even more to calculate packages to be used and even prepare a faster freight search.</p>
<code></commodities></code>	
<code>- <parcels></code>	<p>The parcels that will be sent. Can be specified if commodities are not known, or if you have really decided how big the packages should be and do not want to use Fraktjakt's package calculations.</p>

Se the Parcel-elemts on page 40	
</parcels>	
</CreateShipment>	

Notes

1. Primarily <mobile_to> is saved and secondly <telephone_to>. It is used for notification to the recipient if the shipping product supports that type of notification.

Example – Domestic

```
<?xml version="1.0" encoding="UTF-8"?>
<CreateShipment>
  <consignor>
    <id>YOUR_CONSIGNOR_ID</id>
    <key>YOUR_CONSIGNOR_KEY</key>
    <currency>SEK</currency>
    <language>en</language>
    <encoding>UTF-8</encoding>
    <system_name>Workpress</system_name>
    <module_version>1.0</module_version>
    <api_version>4.7.0</api_version>
  </consignor>
  <reference>Test shipment</reference>
  <address_to>
    <street_address_1>Brunnsgatan 16</street_address_1>
    <street_address_2></street_address_2>
    <postal_code>55317</postal_code>
    <residential>0</residential>
    <country_code>SE</country_code>
    <language>sv</language>
  </address_to>
  <recipient>
    <company_to>Fraktjakt AB</company_to>
    <name_to>Customer Service</name_to>
    <telephone_to>0709-399981</telephone_to>
    <email_to>testmail@fraktjakt.se</email_to>
  </recipient>
  <commodities>
    <commodity>
      <name>Cell phone</name>
      <quantity>3</quantity>
      <taric>01234567</taric>
      <quantity_units>EA</quantity_units>
      <description>Apple iPhone X</description>
      <country_of_manufacture>US</country_of_manufacture>
      <weight>0.6</weight>
      <unit_price>2395</unit_price>
      <currency>USD</currency>
    </commodity>
  </commodities>
</CreateShipment>
```

Response

XML element tag and example of contents	Description
<?xml version="1.0" encoding="UTF-8"?>	
- <result>	
<server_status>No contact with DHL</server_status>	If we have a temporary server problem, it will be announced here.
<status> ok </status>	Status of API call <u>ok</u> = API call succeeded. <u>warning</u> = API call succeeded with warnings. <u>error</u> = API call failed. <i>Sent in response only for API versions 3.1.0 or greater.</i>
<code> 1 </code>	0=OK 1=warning 2=error
<warning_message>Du behöver ange alla mått</warning_message>	Warning message with information that may be useful, especially when testing your integration. Data type: STRING Max chars: 500 Required: Yes
<error_message></error_message>	Error message. Data type: STRING Max chars: 500 Required: No
<shipment_id>45654</shipment_id>	Fraktjakt's shipment ID Data type: INT Required: Yes
<access_code>ABC12345</access_code>	A code for reaching and managing the shipment without being logged in.
<access_link> https://www.fraktjakt.se/shipments/show/163221?access_code=b6dfc12fc04ec98132da2eb1c1739272cc646ed9 </access_link>	A link to the shipment using the above SHIPMENT_ID and ACCESS_CODE. Data type: STRING Required: No
<return_link> https://www.fraktjakt.se/shipments/xml_create_return/163221?	Available from version 4.2.0. A link to create a return shipping based on this shipping.

access_code=b6dfc12fc04ec98132da2eb1c1739272cc646ed9 </return_link>	<p>If there is a pre-selected return product, it will be used, otherwise the best according to the settings for Automatic selection of shipping service.</p> <p>The answer to calling this link will look the same as the answer from the Shipment API except for a new <return_link> tag.</p>
<cancel_link> https://www.fraktjakt.se/shipments/xml_cancel/163221?access_code=b6dfc12fc04ec98132da2eb1c1739272cc646ed9 </cancel_link>	<p>Available from version 4.3.0.</p> <p>A link to delete the shipping created in this call.</p> <p>If the shipping is not paid, it will be deleted immediately, otherwise it will be annulled.</p> <p>In case of annulation, not all shipping companies allow you to unbook the pickup automatically. Then you yourself have to make a annulation by phone directly to the shipping company.</p>
<tracking_code> ABC12345 </tracking_code>	<p>A code to track this shipment without previously being logged in. See note 1.</p>
<tracking_link> https://www.fraktjakt.se/trace/shipment/b6dfc12fc04 </tracking_link>	<p>A link to track the shipment using the above TRACKING_CODE</p> <p>Data type: STRING Required: Yes</p>
</result>	

After the call **you must** to access the new shipment using this link:

https://www.fraktjakt.se/shipments/show/SHIPMENT_ID&access_code=ACCESS_CODE

Notes:

- Put the code in the link:
https://www.fraktjakt.se/trace/shipment/TRACKING_CODE
For more information about the link, see page 102.

Customs Handling

For international shipments, certain customs information must be submitted to ensure smooth processing in the destination country's customs. Some of these details are recommended to be pre-filled in your company's settings, so they do not need to be included in every request.

More information about customs declarations can be found at:

https://www.fraktjakt.se/services/customs_declaration

Below is an excerpt of the fields used to submit customs information:

Address, Sender, Recipient Block (Sender and Recipient Address)

Much of this information can be stored in the company's settings, eliminating the need to include it in each request.

For customs handling, certain information must be included in the sender's and recipient's address:

- `<address_from>` / `<address_to>`
 - `<country_code>` - [Country code](#) for sender/recipient.
- `<sender>/<recipient>`
 - `<tax_id>` - Sender's/Recipient's VAT registration number if applicable.
 - `<eori>` - Sender's/Recipient's EORI number for exports from the EU.
 - `<ioss>` - Sender's/Recipient's Import One-Stop Shop number (for exports from the EU).
 - `<voec>` - Sender's/Recipient's VOEC number (for exports to Norway).
 - `<gb_vat>` - Sender's/Recipient's VAT registration number for the UK.
 - `<mva_num>` - Sender's/Recipient's MVA number for Norway.
 - `<rex>` - Sender's/Recipient's REX number for the origin declaration.

Commodity Block (Product Information)

The [commodity block](#) is used to describe the items included in the shipment.

Note that commodity templates

(https://www.fraktjakt.se/services/commodity_templates) can help with this information.

For customs purposes, the following fields must be specified:

- <name> - Item name.
- <description> - A clear description of the item (at least 15 characters). It must be understandable by the destination country's customs authorities. Different countries have different regulations on how descriptions should be written. Examples for the USA and the EU:
 - USA: <https://www.cbp.gov/trade/basic-import-export/e-commerce/examples-unacceptable-vs-acceptable-cargo-descriptions>
 - EU customs union: https://taxation-customs.ec.europa.eu/system/files/2021-03/guidance_acceptable_goods_description_en.pdf
- <quantity> - Number of units of the item.
- <quantity_units> - Unit of quantity, e.g., EA (each), KG (kilogram), L (liter).
- <weight> - Total weight (in kg) for all units of the item.
- <length>, <width>, <height> - Dimensions per single item in cm.
- <taric> - Taric code (customs classification according to the EU nomenclature). Required for international shipments.
- <country_of_manufacture> - Country of origin according to ISO 3166-1 alpha-2 (e.g., SE for Sweden, US for the USA).
- <unit_price> - Value per unit of the item (including VAT if sold with VAT).
- <currency> - Currency code according to ISO 4217 (e.g., SEK, USD, EUR).

Order API and Shipment API (General Customs Information)

To ensure correct customs declaration for exports, the following fields are used:

- `<export_reason>` - Reason for export:
 - SALE - Sale
 - GIFT - Gift
 - SAMPLE - Sample
 - RETURN - Return
 - REPAIR - Repair
 - PERSONAL EFFECTS - Personal belongings

Recommendations

To avoid customs delays or extra costs:

- Always submit all relevant customs information in the API request.
- Ensure that customs codes and descriptions are clear and properly formatted.
- Specify weight and dimensions accurately.
- If customs information is missing or incorrect in the request, the shipment may be stopped or require manual completion in Fraktjakt. This may also result in penalty fees.

By following these guidelines, international shipments can be handled more smoothly and with fewer issues during customs clearance.

Track & Trace API

In Fraktjakt's settings for companies and integrations there's a tool for creating your own custom shipment tracking page. Here is an example of how a custom tracking page could look:



A webshop can give its customers access to their custom tracking page through the following link:

https://www.fraktjakt.se/trace/shipment/TRACKING_CODE&locale=LOCALE

where TRACKING_CODE is the <tracking_code> returned from a request to Track&Trace API, and where LOCALE specifies the language to be used when rendering the page. Valid locales are currently SV for Swedish and EN for English.

There is a <tracking_link> element returned from a request to Track&Trace API which contains the tracking link including the TRACKING_CODE but without the LOCALE.

Also, there is a <tracking_number> element returned from a request to Track&Trace API which contains the shipments tracking number, which can be used to manually track the shipment from Fraktjakt's tracking page <https://www.fraktjakt.se/trace> or from the shipping company's tracking page.

Do you want to create your own page instead?

If our customization settings don't suit your demands you could create your own tracking page or otherwise retrieve and display tracking information directly to your site using the Track&Trace API.

A request to the Track&Trace API returns an XML response containing status information, which Fraktjakt has previously retrieved from the shipping companies' web services.

This call also returns links to the shipping documents that belong to the shipping. They are downloadable after the shipping is purchased and before it is sent.

The Call

Please note that no XML is needed for the call since we wanted to make the request as simple as possible. Track a shipment in Fraktjakt by calling Track & Trace API with the following URL:

https://www.fraktjakt.se/trace/xml_trace?consignor_id=YOUR_CONSIGNOR_ID&consignor_key=YOUR_CONSIGNOR_KEY&shipment_id=SHIPMENT_ID&locale=LOCALE

SHIPMENT_ID is the shipment_id returned from the Order API or Shipment API. It is not the same shipment_id that is returned from the Query API. Those two id's might be different.

LOCALE is sv or en for Swedish or English.

Also be aware that Fraktjakt might split one shipment into several shipments and that each one of these might have a different status. This is done for all shipping products that require the customer to pick up the package at a service point.

Response

XML element tag and example of contents	Description
<?xml version="1.0" encoding="iso-8859-1"?>	
- <result>	
<server_status>No contact with DHL</server_status>	If we have a temporary server problem, it will be announced here.
<status> ok </status>	Status of API call <u>ok</u> = API call succeeded. <u>warning</u> = API call succeeded with warnings. <u>error</u> = API call failed. <i>Sent in response only for API versions 3.1.0 or greater.</i>
<code> 0 </code>	0=OK 1=warning 2=Error
<warning_message></warning_message>	Warning message with information that may be useful, especially when testing your integration Data type: STRING Max chars: 500 Required: Yes
<error_message></error_message>	Error message. Data type: STRING Max chars: 500 Required: Yes
<shipping_states>	The status of the different shipments It is more than one shipment if the original shipping is divided into several shipments. MinOccur = 1
<shipping_state>	Status for one of the shipments MinOccur = 0 (If the wrong shipment_id is sent)
<shipment_id> 204025 </shipment_id>	Id for the shipment. Might be a different one than the one sent in the call. Data type: INTEGER Required: Yes
felaktig behörighet <name> Not payed </name>	Status as text.

	Data type: STRING Max chars: 500 Required: Yes
<id>0</id>	Status as a number (1). Data type: INTEGER Required: Yes
<fraktjakt_id>8</fraktjakt_id>	Fraktjakt's internal status as a number Data type: INTEGER Required: Yes
</shipping_state>	
</shipping_states>	
<tracking_code> b6dfc12fc04ec98132da2eb1c1739272cc646ed9 </tracking_code>	A code needed to track the shipment. Data type: STRING Required: Yes
<tracking_link> https://www.fraktjakt.se/trace/shipment/b6dfc12fc04ec98132da2eb1c1739272cc646ed9&locale=en </tracking_link>	A link to track the shipment using TRACKING_CODE. Data type: STRING Required: Yes
<tracking_number>BG9700003016</tracking_number>	A code needed to track the shipment. Data type: STRING Required: No – returned only after the shipment has been purchased.
<shipping_company>DB Schenker</shipping_company>	The name of the transport company that performs the shipping. Data type: STRING Required: No - only has a value when it has been determined by the freight buyer.
<shipping_documents>	Any shipping documents associated with this shipment. Required: No - Available only when the shipment is purchased in Fraktjakt and before it is sent.
<shipping_document>	Link to a shipping document on Fraktjakt. Note that some shipping documents are shared by several shipments, such as the manifest that belongs to a booking for one or more shipments. It is sufficient to print these documents once.

	Data type: STRING Required: No - Available only when the shipment is purchased in Fraktjakt and before it is sent.
</shipping_documents>	
</result>	

Status Codes

The status of a shipment is seen in the *fraktjakt_id* field that corresponds to the *shipping_state* of the shipment identified by *shipment_id*.

The following table shows the meanings of the numbers in <shipping_state> elements.

<fraktjakt_id>	<id>	<name>	Comments
0	0	Not paid	If a shipment_id from a Query API response was sent and an Order API call did not send it.
1	0	Preliminary	A shipment_id was sent with an Order API call and a preliminary order has been created.
2	0		Not in use
3	0	Paid	<p>The order for the shipment has been paid for. This means that one or more of the following things happened, depending on the shipping service:</p> <ul style="list-style-type: none">• Shipping documents have been created• EDI have been sent to the carrier• Pickup at the sender is booked• Email sent to the sender.
12	0	Handled by the carrier	The shipment order is present in Fraktjakt but the payment will be made externally at the carrier. An example of this is when you buy Schenker Privpaks services.
4	1	Sent	The carrier has received the shipment either by pickup at the sender or if the sender left it at the terminal or the delivery agent.
5	2	Delivered	The Shipment was delivered to the receiver or to a delivery point close to the receiver, depending on the shipping service purchased.
6	3	Signed	The receiver has signed the delivery of the shipment.
7	4	Returned	The shipment has been returned to the sender for some reason.
8	0	In cart	The shipment has been processed, but not purchased. It is in the shopping cart.
17	0	Edited	The shipment is corrected in Fraktjakt.
18	0	On hold	The shipment was created through order import and is awaiting processing.
19	0	Searching	The shipment is processed in Fraktjakt.

20	0	Waiting confirmation	The shipment is waiting to be confirmed by the end customer.
----	---	----------------------	--

Shipping Documents API

For shipments that has been paid for in Fraktjakt, but has not yet been sent, it is possible to get the shipping documents with a simple call that looks much like the call in the Track & Trace API.

The documents for that shipment will be returned as one of the fields in the XML. The field is a file that is Base64 encoded.

Other fields in the answer are there to make it easier to sort and filter the answer locally.

Call

Please note that no XML is needed for the call since we wanted to make the request as simple as possible.

PROD

API

[https://api.fraktjakt.se/shipping_documents/xml_get?consignor_id=YOUR_CONSIGNOR_ID&consignor_key=YOUR_CONSIGNOR_KEY&shipment_id=SHIPMENT_ID&locale=\[sv|en\]](https://api.fraktjakt.se/shipping_documents/xml_get?consignor_id=YOUR_CONSIGNOR_ID&consignor_key=YOUR_CONSIGNOR_KEY&shipment_id=SHIPMENT_ID&locale=[sv|en])

SHIPMENT_ID is the shipment_id returned from the Order API or shipment API. It is not the same shipment_id that is returned from the Query API. Those two id's might be different.

LOCALE is sv or en for Swedish or English.

Response

XML element tag and example of contents	Description
<?xml version="1.0" encoding="UTF-8"?>	
- <result>	
<server_status>No contact with DHL</server_status>	If we have a temporary server problem, it will be announced here.
<status> ok </status>	Status of API call <u>ok</u> = API call succeeded. <u>warning</u> = API call succeeded with warnings. <u>error</u> = API call failed. <i>Sent in response only for API versions 3.1.0 or greater.</i>
<code> 0 </code>	0=OK 1=warning 2=Error
<warning_message></warning_message>	Warning message with information that may be useful, especially when testing your integration Data type: STRING Max chars: 500 Required: Yes
<error_message></error_message>	Error message. Data type: STRING Max chars: 500 Required: Yes
<shipping_documents>	Any shipping documents associated with this shipment. Required: No - Only available when the shipment is bought in Fraktjakt and before it is sent.
<shipping_document>	The information about a document. Note that some shipping documents are shared by several shipments, such as manifest-lists that belong to a booking for one or more shipments. It is sufficient to print these documents once. Datatype: STRING Mandatory: No, available only after purchase and before shipping is shipped.

<code><name>PATH/Filename.pdf</name></code>	<p>Link to a shipping document, on Fraktjakt.</p> <p>Can be used both to download the shipping document and decide what it should be called in the local system.</p>
<code><type_id>1</type_id></code>	<p>Id for the type of document this is. Can be used for local filtering of the document.</p> <p>Some types of documents, such as Manifests, are shared by multiple shipments and are not needed once for each shipment.</p>
<code><type_name>Shipping label</type_name></code>	<p>The name of the type in plain text. Translated.</p>
<code><type_description>TEXT</type_description></code>	<p>Explanation and instructions for the type. Translated.</p> <p>Can be printed directly to the end customer for assistance.</p>
<code><state_name>Ready</state_name></code>	<p>Status of the document. Translated.</p>
<code><state_description>TEXT</state_description></code>	<p>Description of the status. Translated.</p>
<code><format_name></code> Labels 105 x 220 mm <code></format_name></code>	<p>An explanation of the status. Translated.</p>
<code><file>Base64-coded string</file></code>	<p>Entire Base64 encoded file.</p> <p>Save the string, Base64 decode it and save the file locally.</p>
<code></shipping_document></code>	
<code></shipping_documents></code>	
<code></result></code>	

The document file is a Base64 encoded string.

Sample code to take care of it:

```
file = Base64.decode(Base64 string)
```

```
file.save
```

The file is in pdf-format.

Dokument types

<id>	<name>
1	Pro Forma Invoice
2	Commercial Invoice
3	Shipping label
4	Waybill
5	Manifest
10	Delivery note
11	CN22
12	CN23
13	Security declaration
14	Packing slip

Links to the shipment

When a shipment has been created through an API it can always be accessed on Fraktjakt.se if you have an access_code connected to that shipment. It's retrieved from the Order-API and Shipment-API response.

The shipment can be accessed with this link:

https://www.fraktjakt.se/shipments/show/SHIPMENT_ID?access_code=ACCESS_CODE

If the shipment search isn't complete you're directed to a page where you can complete it. If it's not purchased you're directed to the cart and if it is purchased you get to a page with the shipment or an order confirmation if you're not logged in.

Shipping can be deleted or canceled with the link:

https://www.fraktjakt.se/shipments/xml_cancel/SHIPMENT_ID?access_code=ACCESS_CODE

(There may be a charge for cancellations that occur after shipping is purchased and shipping documents are created.)

A return on shipping can be quickly created with the link:

https://www.fraktjakt.se/shipments/xml_create_return/SHIPMENT_ID?access_code=ACCESS_CODE

Another link points to the tracking page for the shipment. It can be presented directly in your application to the end customer or sent to the addressee.

As soon as the freight is purchased in Fraktjakt, tracking information will be displayed on the page to which the link goes. A good way for the receiver to know where the shipment is.

Tracking link:

https://www.fraktjakt.se/trace/shipment/TRACKING_CODE

Webshops API

(create a user and a webshop)

Contact Fraktjakt if you want to use this API. (api@fraktjakt.se)

The Webshop API makes it possible for an e-commerce platform to easily offer its e-merchants flexible shipping management through Fraktjakt.

The e-commerce platform uses the Webshop API to send information about an e-merchant to Fraktjakt. This creates a user, a company and an integration in Fraktjakt. The e-merchant can then log in with his user in Fraktjakt to be able to buy/book freight for his company.

If the e-commerce platform has also integrated with Fraktjakt Query API, Order API and/or Shipment API then the e-merchants' order details can be automatically transferred to Fraktjakt to simplify the e-merchant's freight handling even more.

Good to know when integrating with the Webshop API

The request to the Webshop API should normally create a user, a company and an integration in Fraktjakt. The company is linked to the user through a role that gives the user authorization to administer the company in the Fraktjakt system. The user may also administer the integration.

The Relationships between Users, Companies and Integrations in Fraktjakt

- * a User may be authorized to administer one or more Companies (and Company Integrations)
- * a Company may have one or more Integrations and be administered by one or more Users.
- * an Integration belongs to a Company

The user is created based on the submitted user's email address sent in the <email> tag.

For new Users in Fraktjakt, the following applies:

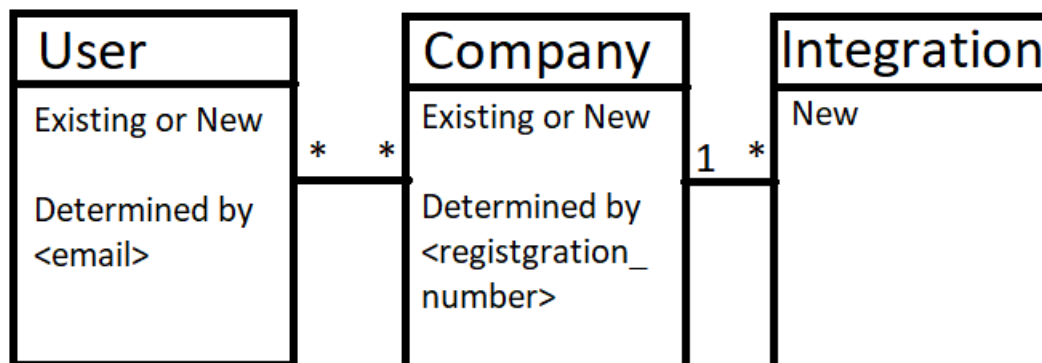
If <password> is not specified, Fraktjakt creates a random password for the user. The password is returned in the response.

If the submitted organization number in the tag <registration_number> has already been registered to a Company in Fraktjakt, the Webshop API will not create a new Company in Fraktjakt. The e-commerce platform is referred to Fraktjakt's customer service to find out how to proceed.

For existing Users in Fraktjakt, the following applies:

The correct email address and password must be entered.

If the submitted organization number in the <registration_number> tag has already been registered to a Company in Fraktjakt, the User must have permission to administer the Company in order for an Integration to be created.



The API has the following URL:

https://api.fraktjakt.se/webshops/create_xml

(PROD API – production environment)

IMPORTANT No user-accounts can be used before they are activated. We strongly recommend that all settings for the new integration be reviewed. Basic settings are retrieved from calling integration, but most must be set after the integration is created.

Since version 4.1, this API has been rewritten and simplified. The old version will not work any more.

If the calling Integration is in test-mode no real data will be created. This way you can test your integration without any problems until you are ready to go live.

We don't want you to create incorrect accounts during development because our customer service then checks the creditworthiness and if all the information matches real existing companies. If you do your testing by creating sharp accounts, you cause a lot of extra work for us behind the scenes.

XML for webshop API calls

XML element tag and example of contents	Description
<?xml version="1.0" encoding="iso-8859-1"?>	
- <integration>	
- <consignor>	Information about the consignor, i.e. the integration that is creating

	another integration.
See consignor-element on page 16	
</consignor>	
<email>name@email.se</email>	<p>Users email</p> <p>Used to specify the correct user in Fraktjakt, or create a new one.</p> <p>Data type: STRING</p> <p>Required: Yes</p>
<user_name>Olle Klint</user_name>	<p>Users name (not login).</p> <p>Both first name and surname.</p> <p>Only used if this is a new user in Fraktjakt.</p> <p>Data type: STRING</p> <p>Required: Yes, if new user</p> <p>Default: Consignor-users value</p>
<password>secret</password>	<p>Users password</p> <p>If an existing user is entered, the password MUST be entered and be correct.</p> <p>Data type: STRING</p> <p>Required: No</p> <p>Default: Random, for new user.</p>
<telephone>036123456</telephone>	<p>The integrations telephone number</p> <p>Data type: STRING</p> <p>Required: Yes, if new user</p>
<company_name>Joppes import</company_name>	<p>The name for the new company.</p> <p>The real company name linked to your <registration_number>.</p> <p>Used only for new companies.</p> <p>If <registration_number> specifies an existing company to which the user is authorized, the field is NOT used.</p> <p>If the field is not specified for new companies, <integration_name> is used instead</p> <p>Data type: STRING</p> <p>Required: Yes, if new company</p>
<integration_name>Joppes</integration_name>	<p>The name for the new integration.</p> <p>Displayed on shipping documents to the end-customer.</p> <p>Not a unique name. Several</p>

	<p>integrations can have the same name.</p> <p>Data type: STRING</p> <p>Required: Yes</p>
<code><url>https://kottegott.se</url></code>	<p>URL the the integration</p> <p>Data type: STRING</p> <p>Required: No</p>
<code><registration_number>650616-7235</registration_number></code>	<p>The company's registration-number (organization number).</p> <p>Either a company that is NOT registered in Fraktjakt or an existing company to which the specified user is authorized.</p> <p>If the field is not specified for an existing user, the user's first company is taken.</p> <p>Data type: STRING</p> <p>Required: Yes, if new company</p>
<code><sender_address></code>	<p>Sender address for the webshop and the user.f. Used in alla fright-queries.</p> <p>Required: Yes, for new accounts</p>
See Address-element on page ?	
<code></sender_address></code>	
<code><invoice_address></code> Address-element on page ? <code></invoice_address></code>	<p>Invoice-address. The sender_address will be used If it isn't provided, or if that isn't provided, the invoice-address for the consignor-user.</p> <p>Required: No</p>
<code><postal_address></code> See Address-element on page ? <code></postal_address></code>	<p>Postal-address. The sender_address will be used If it isn't provided, or if that isn't provided, the postal-address for the consignor-user.</p> <p>Required: No</p>
<code></integration></code>	

Response

XML element tag and example of contents	Description
<?xml version="1.0" encoding="iso-8859-1"?>	
- <result>	
<server_status>No contact with DHL</server_status>	If we have a temporary server problem, it will be announced here.
<status>ok</status>	<p>Status of API call</p> <p><u>ok</u> = API call succeeded. <u>warning</u> = API call succeeded with warnings. <u>error</u> = API call failed.</p> <p><i>Sent in response only for API versions 3.1.0 or greater.</i></p>
<code>1</code>	<p>0=OK 1=warning 2=Error</p>
<warning_message></warning_message>	<p>Warning message with information that may be useful, especially when testing the integration.</p> <p>Data type: STRING Max chars: 500</p>
<error_message></error_message>	<p>Error message.</p> <p>Data type: STRING Max chars: 500</p>
<consignor_id>NEW_CONSIGNOR_ID</consignor_id>	<p>Users ID – Used in future call to Fraktjakt</p> <p>Data type: STRING</p>
<consignor_key>NEW_CONSIGNOR_KEY</consignor_key>	<p>Users key – Used in future call to Fraktjakt</p> <p>Data type: STRING</p>
</result>	

Service Point Locator API

API for searching for agents / service points

To retrieve information about the nearest agent, there is a link. It returns the nearest agent based on a more or less well-specified address. It is possible to call without having an account with us, even if the results are a little better then.

It is intended to work regardless of which other API you use with us.

You call it by a single URL. It may look like these examples:

- https://api.fraktjakt.se/agents/service_point_locator?locale=en&consignor_id=32830&consignor_key=6f18d7afffbba8135733bd57cc4c8781fe22eeb9&country=se&city=Jönköping&street=Östra storgatatan 49 &postal_code = 55321
- https://api.fraktjakt.se/agents/service_point_locator?locale=en&consignor_id=32830&consignor_key=6f18d7afffbba8135733bd57cc4c8781fe22eeb9&country=se&city=J%C3%B6nk%C3%B6ping&street=%C3%96stra%20storgatan%2072&postal_code=55321
- https://api.fraktjakt.se/agents/service_point_locator?locale=sv&consignor_id=32830&consignor_key=6f18d7afffbba8135733bd57cc4c8781fe22eeb9&country=se&city=Solna&postal_code=111111
- https://api.fraktjakt.se/agents/service_point_locator?locale=sv&consignor_id=32830&consignor_key=6f18d7afffbba8135733bd57cc4c8781fe22eeb9&country=se&city=Solna

It thus starts with https://api.fraktjakt.se/agents/service_point_locator, then follows a number of parameters.

Examples of links are returned in the response in the Query API and Order API.

The various parameters that can be specified are:

- * locale - The language. Selectable is 'sv' or 'en'.
- * consignor_id - Indicates which integration you have. Gives a result that is adapted to the settings you have in Fraktjakt.
- * consignor_key - The integration's password. Downloaded from the Installation tab for the integration in Fraktjakt.
- * An address with the parameters: country, city, street, postal_code. The more parameters you specify, the better the result. At least the city should be stated. The country is specified by ISO 3166 code, see page 103. If no country is specified, we assume that it applies to Sweden.

* A limiter in the form of shipper_id, vendor (name of shipping company) or shipping_product_id. The latter is excellent if you use the API in connection with the Query API.

The answer is Json-formatted and also contains some documentation of what the call should look like. Example:

```
"agents": [
  {
    "shipper": "DHL Freight",
    "agent_operation_hours": null,
    "html_info": "Cash on delivery / COD <br /> DHL Express delivery / pickup
<br /> DHL Express delivery / dropoff",
    "latitude": 57.866177,
    "longitude": 14.117463,
    "name": "BANKERYDS BIL (INGO)",
    "shipper_id": 5,
    "id": 78035,
    "distance": "5.850653384970596",
    "address": {
      "country_code": "SE", "postal_code": "56435",
      "city_name": "Bankeryd", "street": "Kortebovägen 6"
    }
  },
  {
    "notification": {
      "hint": "" Always enter shipping_product_id, shipper_id or vendor best and
most accurate results. Address can be specified with the country, city,
postal_code and street parameters. "
      "available_shipper_selections": [
        {"vendor": "bussgods", "shipper_id": 4},
        {"vendor": "dhl", "shipper_id": 5},
      ],
      "available_shipping_products": [
        {"shipping_product_id": 99, "name": "Service Point", "shipper_id": 5},
      ]
    }
  }
]
```

The answer is an array of agents, possible error messages and help texts.

'Shipper': Shipping company name

'Shipper_id': ID of the shipping company. Can be used when you want to search only for that shipping company.

'name': Agent name

'address':

 'street': Street address

 'postal_code': Postal code

 'city': Place name

 'country': IANA code for the country

'html_info': Extra information about agents in HTML code

'latitude': latitude

'longitude': longitude,

'distance': How far it is from the address you searched from.

'agent_operation_hours': A large element that tries to present the agent's opening hours. Not all shipping companies present opening hours to their

agents:

Exempel {'default':[],'overrides':{'weekday':{'wed':
[{'open':'09:00','close':'18:00'}], 'sat':[{'open':'09:00','close':'15:00'}], 'fri':
[{'open':'09:00','close':'18:00'}], 'tue':[{'open':'09:00','close':'18:00'}], 'mon':
[{'open':'09:00','close':'18:00'}], 'sun':[{'open':'10:00','close':'15:00'}], 'thu':
[{'open':'09:00','close':'18:00'}]}}}

'default': [Openinghours]

'overrides' : Hash with weekdays and / or dates.

Every day has an array of opening hours. More times may occur for eg lunch closing.

Opening hours can be 'always_open', 'closed' or {'open': Time, 'close': Time}

In addition to the array of proxies, there are also the fields

'notification': Various tips to get a better answer.

'status': Status of the API call. ok = API call succeeded. warning = API call succeeded with warnings. error = The API call failed.

'code': 0=OK, 1=Warning, 2=Error

'error_message': If something has gone wrong, the error is also in plain text here in the selected language. The field does not exist if everything went well.

HTTP status codes in the answer: 200 if everything is ok. 400 if something is wrong with the call.

The result of the call must never be saved for longer than one day. Agents can get new IDs every night.

The most important field in the answer is the ID of each agent. This is the ID to be used when selecting agents in the <agent_id>-tag in the Order API and Shipment API or via Service Point Selector API. See page 111 for information on how it works.

It can also be used in a separate call to the small Service Point Selector API

Show the agent to the customers

The agent can also be displayed in a customer-friendly way with the link

https://www.fraktjakt.se/agents/show/AGENT_ID?redirect=ecommerce

See page 119 for more information about the link.

Service Point Selector API

If the shipping is in Fraktjakt and has not yet been purchased, it is possible to choose an agent with this small API.

It may be easier to implement it directly instead of having to rewrite existing API implementations.

It is easy to use. Just call the URL

https://api.fraktjakt.se/agents/service_point_selector?locale=SV&shipment_id=ID&access_code=ACCESS_CODE&agent_id=AGENT_ID

shipment_id and ACCESS_CODE are obtained from the answers in Fraktjakt's various APIs.

Agent_id is obtained from the Service Point Locator API.

HTTP status codes in the answer: 200 if everything is ok. 400 if something is wrong with the call.

Shipping must not be paid in Freight Hunt for this to work.

How to let customers choose agents

One of the most requested functions when you want to implement one of our APIs is to let customers choose which agent they want the shipping to go to. In Fraktjakt, there are several ways to solve it.

Important - Once a freight has been purchased in Fraktjakt, you can no longer select or change agents.

The reason is that the shipping company is then notified of the shipping and where it is to go. The shipping label also states to which agent the shipping is to be made.

Option 1 - Agent Selection Link

The easiest way to let customers choose agents has Fraktjakt for Query API or Order API.

In the answer from the Query API there is a link per shipping product in the answer's `<agent_link>` tag, see page 52.

It goes to a website where the customer can choose an agent who works for the shipping product with which it is associated.

To the link you can add the formatting parameter "redirect" with the options "ecommerce", "clean" or "raw", depending on how you want to present the agent selection page to your customers. Ecommerce is suitable if the page is displayed in an iFrame, clean for a minimal layout and raw if you scrape the page with, for example, curl and present the result yourself, see page 119.

In cases where it is not possible to present a proxy selection link per shipping product, there is also a general link for proxy selection in the response from the Query API in the `<agent_selection_link>` tag. It shows options for all the shipping products shown in the Query API response. You can also add the redirect parameter to that link.

The response from the Order API also has a `<agent_link>` tag that works just like the `<agent_link>` tags in the response from the Query API, with redirect parameters.

With that link, the end customer can have the opportunity to choose a representative for the shipping until it is purchased in Fraktjakt. After that, the link only shows information about the agent you pick up the package from.

Option 2 - Service Point Locator API and <agent_id> tag

We have a special API for finding agents. It is called the Service Point Locator API and is described on the page 108.

Via a simple call from a link, you get a json with a list of the nearest agents. The agents are geocoded and have a distance to the specified address. With the information in the answer, it is possible to either present the agents on any map service or present an agent list.

The end customer can choose an agent there.

In the Order API and Shipment API, you can then add the <agent_id> tag to send with the customer's agent choice.

In the Order API, the selected agent is used for the freight if it belongs to the same company as the shipping product you choose and the shipping product is to an agent. Otherwise we'll throw it away.

Please note, the choice of agent must be made before the freight is purchased in Fraktjakt.

In the Shipment API, the agent selection will control the automatic shipping selection, so that only services that work with the selected agent will be able to be selected. This means that if you let the customer choose an agent before sending the shipment to Fraktjakt, the shipping service you will receive will work with the agent selection. If you do not want the shipping to go to an agent, you should not send in an agent selection.

Option 3 - Service Point Locator API and Service Point Selector API

Instead of sending the new <agent_id> tag in the API calls, you can use the new Service Point Selector API see page 111.

It is simply an easy call to a special URL.

To show that you have the correct authority to handle a selected shipment, use the Access code returned in Query API, Order API and Shipment API calls.

Shipping Products API

We have a simple call to get a list of all shipping services currently supported in Fraktjakt. Simply call the URL:

https://api.fraktjakt.se/shipping_products/xml_list

(Currently we only support locale='sv' on this call.)

The response itself is an XML that contains the tag <shipping_products> filled with several <shipping_product> tags.

Example of a <shipping_product> tag:

```
<shipping-product>
  <cold>false</cold>
  <courier>false</courier>
  <description>Euroline &#xE4;r DHL Freight's frakttj&#xE4;nst f&#xF6;r
st&#xF6;rre partifrakter inom Europa. Med ett stort n&#xE4;tverk i &#xF6;rver 40
l&#xE4;nder f&#xE5;r du en smidig och p&#xE5;litlig transportl&#xF6;sning som
kombinerar internationell r&#xE4;ckvidd med lokal expertis. DHL Freight erbjuder
flexibla l&#xF6;sningar anpassade efter dina behov, oavsett s&#xE4;song. DHL:s
p&#xE5;litliga transportflotta och m&#xF6;jlighet att kombinera olika
transport&#xE4;tt g&#xF6;r det b&#xE5;de billigare och mer h&#xE5;llbart att
frakta ditt gods.</description>
  <dropoff-included>true</dropoff-included>
  <express>false</express>
  <iced>false</iced>
  <id>234</id>
  <individual-contract>true</individual-contract>
  <international>true</international>
  <name>DHL - Euroline (export)</name>
  <pickup-included>true</pickup-included>
  <shipper-id>5</shipper-id>
  <terms>https://www.dhl.com/content/dam/dhl/local/se/dhl-
freight/documents/pdf/sv/se-freight-general-terms-and-conditions-sv.pdf</terms>
</shipping-product>
```

If you want to get a list of the shipping services your integration has access to, simply send consignor_id and consignor_key in the call URL.

https://api.fraktjakt.se/shipping_products/xml_list?consignor_id=MY_ID&consignor_key=MY_KEY

Status API

We have a simple call to check if our servers are working as they should. The answer should be as fast as possible.

Simply call the URL:

<https://api.fraktjakt.se/application/status?locale=sv>

(We have support for locale='en' too)

If you get an answer, there is something that answers in some way.

The response itself is a json with two fields:

```
{  
  "status": "ALIVE",  
  "server_status": "DHL: No contact"  
}
```

"Status" 'ALIVE' if everything is up

 'DEAD' if Fraktjakt's system is down in some way.

"server_status" Are minor problems, such as a shipping company's server being down.

Testing using the API Test Page

You may use the forms found at https://www.fraktjakt.se/xml/query_test to test your XML. You can find this page in the left-menu in the integration settings called 'API Test page'.

Here you can try the following test forms for your calls:

- Query API
- Requery API
- Order API, type 1
- Order API, type 2
- Shipment API
- Track & Trace API
- Webshops API
- Shipping Documents API

You can also test a webhook call.

There are several additional xml examples prepared with your Consignor ID & Key already in them (If you're logged in and you've selected your registered integration), just click submit to try them out.

The above forms are also available in the PROD API live production environment, in case you have reason to test your XML documents there.

Be aware that any Order API documents that you enter in the PROD API environment will generate real shipping orders.

To test another XML document, copy and paste the entire contents of the XML document directly in the form at one of the above links.

Then press the *"Submit"* button. This will cause the same results to be displayed in your browser that you would have received from the Fraktjakt API if you had sent the XML document via the API.

Tips when planning a parcel module integration

Create a fall-back in case of problems

Although we strive to maintain a stable environment for our system, there still are circumstances beyond our control which can affect the stability and access to the system. Examples could be power outage, loss of broadband connection, hardware failure, etc.

We therefore recommend that you try and create a fall back in case you are not able to reach our system to retrieve parcel rates.

As an example, we recommend referring to simplified shipping rates in case your system is unable to retrieve parcel rates from us.

Remember to set your system to automatically send Error reports to us so that we can fix problems in a timely manner.

Get faster response times

Everyone loves fast responses. Here is a list of things you can implement to get faster responses from Fraktjakt.

One method to increase response times is to configure the parcel module to only retrieve and display a minimum amount of shippers. The less data communication (retrieval requests) the quicker the response times.

Our system is also able to provide parcel rates for those e-commerce platforms that require parcel rates before a customer indicates the destination of which the product will be sent (such as filling out their postal address during purchase). If you set your company's address as the receiving address we are able to indicate a parcel rate before the purchase and provide an adjusted rate when customers fill in their address during purchase.

If parcel prices are not important however whether or not a certain parcel service is available to a certain address or not, you should send a `<no_prices>` element with the search.

If you are only going to use one shipping method, for example, if you use an AJAX query to update prices on a specific shipping method, then you should set your parcel module not to search for other shipping methods. Send instead `shipping_product_id` in the search xml.

If you think that providing or showing shipping agents (places where the customer can pick up their parcel shipments) then you should send the `<no_agents>` element. It quickens up the response times.

Webhook Callback

By entering the <callback_url> tag in the Query API, Order API or Shipment API, Fraktjakt will activate its webhook for that shipment.

This means that as soon as the shipment is purchased in Fraktjakt, Fraktjakt will send information to the specified URL in the form of a JSON that contains lots of information about the freight, including where the package is and all tracking information.

Every time Fraktjakt discovers that something is happening with the freight, a call is sent. This can either be because its status changes in Fraktjakt, or because the freight company reports something new in its tracing information.

Exactly what the call looks like and how to receive it is stated in the document Fraktjakt_Webhook_en.pdf

If no one answers the calls from Fraktjakt correctly, Fraktjakt reserves the right to turn off this functionality for your Integration.

Help and information can be obtained if you contact Fraktjakt at api@fraktjakt.se

Tip: In the tag, it is possible to send in a URL with its own security parameter, which is then used in the call from Fraktjakt. For example? Key = abcdef123456

Other integration possibilities

If you'd like, you can let your users save their Fraktjakt user names and passwords in your application. This will enable you to open an external browser window or an iFrame using the following URL:s:

Common parameters to all links

You can add the format-parameter "redirect" to the URL:s

- redirect=ecommerce is a reserved parameter that you may optionally include in order to eliminate layout information from the Fraktjakt page that your user sees (in case you want to apply a different layout locally). Suitable when used in an iFrame.
- redirect=clean removes our wrapper with sidehead, side footer and menu and replaces them with "Powered by Fraktjakt" at the bottom. Suitable in an iFrame which you want to easily layout to fit the webshop.
- redirect=raw removes all layout and returns parts of a webpage.

It is also possible to specify which language is to be used with the "locale" parameter, which can have the values "sv" or "en".

View information about an individual agent

https://www.fraktjakt.se/agents/show/AGENT_ID?redirect=ecommerce

The AGENT_ID comes from the response from the Service Point locator API, and is the same ID that is used in agent selection.

Change password

https://www.fraktjakt.se/account/change_email?redirect=ecommerce

Administration of the integration associated with the logged-in user

<https://www.fraktjakt.se/webshops/change?redirect=ecommerce>

Administration of addresses

https://www.fraktjakt.se/account/change_addresses?redirect=ecommerce

Track a shipment based on tracking number and shipping company

Here is a way to track a shipment regardless of whether the shipment is purchased through Fraktjakt or not. To do this you'll need the shipments

tracking number (TRACKING_NUMER) and the shipping company's id number in Fraktjakt (SHIPPER_ID), and these should be sent using the following link.

https://www.fraktjakt.se/trace/create?tracking_number=TRACKING_NUMBER&shipper_id=SHIPPER_ID

The shipping company's id number in Fraktjakt (SHIPPER_ID) is as follows.

DHL - 5, FedEx - 1015, PostNord - 1, Schenker - 2, UPS – 1036, Ntex – 8, DSV -2453, Bring - 3

Information on all shipping services currently supported via Fraktjakt

Use the Shipping Products API to retrieve an XML listing of all shipping services currently supported via Fraktjakt. It displays fields for service description, terms and conditions and other things that may be of interest to shipment buyers.

If you use the Order API, call type 2 or the Shipment API with a specified favorite service, this call will tell you which shipping_product_ids are available to enter.

NOTE! Before using a shipping service in the Order API, call type 2, you must ensure that that particular service serves the sender and recipient address and that the other product conditions are met.

If you want to get a list of the shipping services your integration has access to, simply send the consignor_id and consignor_key in the Shipping Products API call URL.

More information on search results

By copying the shipping product ID from the `<shipping-product><id>` element that you receive in the XML from Fraktjakt's API, you can generate links to more information about the shipping product.

```
- <shipping_product>
```

```
<id>NR</id>
```

You can generate links to the following interesting pages on Fraktjakt using this shipping product ID (called NR below) :

Terms and conditions for the shipping product

https://www.fraktjakt.se/shipping_products/show_terms/NR

Other information about the shipping product

https://www.fraktjakt.se/shipping_products/show/NR

Common problems

- **No response from Fraktjakt**

This problem is typically caused by a communication error. The most common sources of these errors are firewalls at web hotels. Verify that the firewall is permitting all traffic between your integration and Fraktjakt.

The problem may also be caused by a setting in your server that prohibits cURL (libcurl) from working properly. Try changing the communication method in your administration interface to "fopen".

Another possible cause of this problem is that Fraktjakt is experiencing a temporary communication problem. In order to eliminate this as the cause of the problem, verify that you can connect to <https://api.fraktjakt.se> in a web browser.

If you are experiencing problems ordering international shipping, the culprit could be an incorrect recipient address. Verify that your system has sent a valid recipient address to Fraktjakt.

The sender's or recipient's address (or both) must be located in Sweden.

- **You receive an XML file that contains the text, "503 Service Temporarily Unavailable".**

There is probably a problem in the Fraktjakt server, and we are fixing the problem.

- **Orders from my OsCommerce integration are not reaching Fraktjakt.**

If you have installed the OsCommerce module for Fraktjakt, you must make sure that both files in the installation package are installed. The first file only makes it possible to present shipping prices in your integration. The second module (The Fraktdata module) allows orders to be sent to Fraktjakt.

- **My XML file does not seem to work, whatever I do!**

Keep in mind that the file must be URL-encoded before you can send it. https://en.wikipedia.org/wiki/Query_string#URL_encoding

- **Not all shipping products are shown in the results that are supposed to be shown.**

Check your settings for your webshop if you are searching as a private person or as a company(swedish: "företag").

Error reporting

If you encounter problems when using the Fraktjakt API, you may send the entire XML document along with the response you received from the API to the following e-mail address: 'api@fraktjakt.se'.

We will promptly analyse the problem and respond to the reply-to address of the e-mail.

Feel free to automatically generate mails when errors occur, but please ensure that there is someone available to receive our responses to such error reports.

Support

Contact us

https://www.fraktjakt.se/om_fraktjakt/kontakt

Fraktjakt AB
Gjuterigatan 9
55318 Jönköping
SWEDEN

VAT number: SE556751901101

E-post: api@fraktjakt.se

Webchat: https://www.fraktjakt.se/om_fraktjakt/kontakt

[Customer Service: info@fraktjakt.se](mailto:info@fraktjakt.se)

New versions of that document are announced on the mailing list

<https://groups.google.com/d/forum/fraktjakt-api>

We strongly recommend that you subscribe to that mailinglist.

Customer support office hours

Monday – Thursday: 8.00 – 16.15 CET (Central European Time)

Friday: 9:00 – 16:15

Closed for lunch: 11:30 - 12:30

Saturday and Sunday: Closed

Swedish national holidays: Closed

Appendix 1 - Language

The following languages are now supported in Fraktjakt:

Code	Language
sv	Swedish
en	English

The above language codes follow ISO 639-1. Se details in Wikipedia:
https://en.wikipedia.org/wiki/List_of_ISO_639-1_codes.

<language> elements are included in two parts of the XML for API calls:

- in the <consignor> element, so Fraktjakt knows which language to respond to the sender of the API call (identified directly or indirectly by the consignor block);
- in <address_to> elements, so Fraktjakt knows which language to use when communicating with the recipients of shipments (normally customers of the consignor).

Support for English is currently only partial in Fraktjakt's responses to API calls, but it is complete in all correspondence with the receivers of shipments. If <sender_email> elements are included in Order-API calls, the senders of the shipments also receive all of their communication in the specified language.

Fraktjakt's branding and customization features are fully functional in both Swedish and English

Appendix 2 – Countries in Fraktjakt

The following countries can be selected in Fraktjakt API calls by submitting their codes (as shown in the table) in the appropriate address elements (See [Address element](#)). Note, however, that most shipping services do not serve all of these countries.

Fraktjakt can be used for shipping to or from Sweden, but either the originating address (<address_from>), the destination address (<address_to>), or both must be in Sweden.

The country codes follow ISO 3166-1 alpha-2 (See https://en.wikipedia.org/wiki/ISO_3166-1_alpha-2).

Name	Code	Has Postal Codes
Aaland Islands	AX	YES
Afghanistan	AF	YES
Albania	AL	YES
Algeria	DZ	YES
American Samoa	AS	YES
Andorra	AD	YES
Angola	AO	YES
Anguilla	AI	YES
Antarctica	AQ	NO
Antigua and Barbuda	AG	NO
Argentina	AR	YES
Armenia	AM	YES
Aruba	AW	NO
Ascension Island	AC	YES
Australia	AU	YES
Austria	AT	YES
Azerbaijan	AZ	YES
Azores (Portuguese autonomous region)	PT	YES
Bahamas	BS	NO
Bahrain	BH	YES
Bangladesh	BD	YES

Barbados	BB	YES
Belarus	BY	YES
Belgium	BE	YES
Belize	BZ	NO
Benin	BJ	NO
Bermuda	BM	YES
Bhutan	BT	YES
Bolivia	BO	NO
Bosnia and Herzegovina	BA	YES
Botswana	BW	NO
Bouvet Island	BV	NO
Brazil	BR	YES
British Indian Ocean Territory	IO	YES
Brunei Darussalam	BN	YES
Bulgaria	BG	YES
Burkina Faso	BF	NO
Burundi	BI	NO
Cambodia	KH	YES
Cameroon	CM	NO
Canada	CA	YES
Canary Islands	IC	NO
Cape Verde	CV	YES
Cayman Islands	KY	YES

Central African Republic	CF	NO
Ceuta and Melilla	EA	NO
Chad	TD	YES
Chile	CL	YES
China	CN	YES
Christmas Island	CX	YES
Clipperton Island	CP	NO
Cocos (Keeling) Islands	CC	YES
Colombia	CO	YES
Comoros	KM	NO
Congo	CG	NO
Congo, Democratic Republic of the	CD	YES
Cook Islands	CK	NO
Costa Rica	CR	YES
Croatia	HR	YES
Cuba	CU	YES
Curacao	CW	NO
Cyprus	CY	YES
Czech Republic	CZ	YES
Denmark	DK	YES
Diego Garcia	DG	NO
Djibouti	DJ	NO
Dominica	DM	NO
Dominican Republic	DO	YES
Ecuador	EC	YES
Egypt	EG	YES
El Salvador	SV	YES
Equatorial Guinea	GQ	NO
Eritrea	ER	NO
Estonia	EE	YES
Ethiopia	ET	YES
Falkland Islands (Malvinas)	FK	YES
Faroe Islands	FO	YES
Fiji	FJ	NO
Finland	FI	YES
France	FR	YES
France, Metropolitan	FX	NO
French Guiana	GF	YES
French Polynesia	PF	YES
French Southern Territories	TF	NO
Gabon	GA	YES

Gambia	GM	NO
Georgia	GE	YES
Germany	DE	YES
Ghana	GH	NO
Gibraltar	GI	YES
Greece	GR	YES
Greenland	GL	YES
Grenada	GD	NO
Guadeloupe	GP	YES
Guam	GU	YES
Guatemala	GT	YES
Guernsey	GG	YES
Guinea	GN	YES
Guinea-Bissau	GW	YES
Guyana	GY	NO
Haiti	HT	YES
Heard Island and McDonald Islands	HM	YES
Honduras	HN	YES
Hong Kong	HK	NO
Hungary	HU	YES
Iceland	IS	YES
India	IN	YES
Indonesia	ID	YES
Iran, Islamic Republic of	IR	YES
Iraq	IQ	YES
Ireland	IE	YES
Isle of Man	IM	YES
Israel	IL	YES
Italy	IT	YES
Ivory Coast	CI	NO
Jamaica	JM	YES
Japan	JP	YES
Jersey	JE	YES
Jordan	JO	YES
Kazakhstan	KZ	YES
Kenya	KE	YES
Kiribati	KI	NO
Korea, Democratic People's Republic of	KP	NO
Korea, Republic of	KR	YES
Kosovo (use Serbia)	XK	YES
Kuwait	KW	YES

Kyrgyzstan	KG	YES
Lao People's Democratic Republic	LA	YES
Latvia	LV	YES
Lebanon	LB	YES
Lesotho	LS	YES
Liberia	LR	YES
Libyan Arab Jamahiriya	LY	YES
Liechtenstein	LI	YES
Lithuania	LT	YES
Luxembourg	LU	YES
Macao	MO	NO
Macedonia	MK	YES
Madagascar	MG	YES
Malawi	MW	NO
Malaysia	MY	YES
Maldives	MV	YES
Mali	ML	NO
Malta	MT	YES
Marshall Islands	MH	YES
Martinique	MQ	YES
Mauritania	MR	NO
Mauritius	MU	NO
Mexico	MX	YES
Micronesia, Federated States of	FM	YES
Moldova	MD	YES
Monaco	MC	YES
Mongolia	MN	YES
Montenegro	ME	YES
Montserrat	MS	YES
Morocco	MA	YES
Mozambique	MZ	YES
Myanmar	MM	YES
Namibia	NA	YES
Nauru	NR	NO
Nepal	NP	YES
Netherlands	NL	YES
Netherlands Antilles	AN	NO
New Caledonia	NC	YES
New Zealand	NZ	YES
Nicaragua	NI	YES

Niger	NE	YES
Nigeria	NG	YES
Niue	NU	NO
Norfolk Island	NF	YES
Northern Mariana Islands	MP	YES
Norway	NO	YES
Oman	OM	YES
Pakistan	PK	YES
Palau	PW	YES
Palestinian National Authority	PS	NO
Panama	PA	YES
Papua New Guinea	PG	YES
Paraguay	PY	YES
Peru	PE	YES
Philippines	PH	YES
Pitcairn	PN	YES
Poland	PL	YES
Portugal	PT	NO
Puerto Rico	PR	YES
Qatar	QA	NO
Réunion	RE	YES
Romania	RO	YES
Russia	RU	YES
Rwanda	RW	NO
Saint Barthélemy	BL	YES
Saint Helena	SH	YES
Saint Kitts and Nevis	KN	NO
Saint Lucia	LC	YES
Saint Martin (French part)	MF	YES
Saint Pierre and Miquelon	PM	YES
Saint Vincent and the Grenadines	VC	YES
Samoa	WS	NO
San Marino	SM	YES
Sao Tome and Principe	ST	NO
Saudi Arabia	SA	YES
Senegal	SN	YES
Serbia	RS	YES
Serbia and Montenegro (ended 2006)	CS	NO
Seychelles	SC	NO
Sierra Leone	SL	NO

Singapore	SG	YES
Slovakia	SK	YES
Slovenia	SI	YES
Solomon Islands	SB	NO
Somalia	SO	NO
South Africa	ZA	YES
South Georgia and the South Sandwich Islands	GS	YES
Spain	ES	YES
Sri Lanka	LK	YES
Sudan	SD	YES
Suriname	SR	NO
Svalbard and Jan Mayen	SJ	YES
Swaziland	SZ	YES
Sweden	SE	YES
Switzerland	CH	YES
Syrian Arab Republic	SY	NO
Taiwan, Province of China	TW	NO
Tajikistan	TJ	YES
Tanzania, United Republic of	TZ	NO
Thailand	TH	YES
Timor-Leste	TL	NO
Togo	TG	NO
Tokelau	TK	NO
Tonga	TO	NO
Trinidad and Tobago	TT	YES

Tristan da Cunha	TA	NO
Tunisia	TN	YES
Turkey	TR	YES
Turkmenistan	TM	YES
Turks and Caicos Islands	TC	YES
Tuvalu	TV	NO
Uganda	UG	NO
Ukraine	UA	YES
United Arab Emirates	AE	NO
United Kingdom	GB	YES
United States	US	YES
United States Minor Outlying Islands	UM	NO
Uruguay	UY	YES
Uzbekistan	UZ	YES
Vanuatu	VU	NO
Vatican City State	VA	YES
Venezuela	VE	YES
Viet Nam	VN	YES
Virgin Islands, British	VG	YES
Virgin Islands, U.S.	VI	YES
Wallis and Futuna	WF	YES
Western Sahara	EH	NO
Yemen	YE	NO
Zambia	ZM	YES
Zimbabwe	ZW	NO