This document describes available API endpoints for XBS. For each command there will be a description on what data the API exposes, the parameters and an example request and response.

XBS API manual

API Documentation

API DOCUMENTATION

Contents

Change log

Change date	Change description
2021-10-28	Added Tracking Webhook in chapter 2.2.1 (page 14)
2021-10-04	Added CreateGroup and VoidGroup command
2021-08-30	Added GetServices command
2021-06-01	Added IOSS fields & explanation: o ConsignorAddress.NlVat o ConsignorAddress.EuEori o ConsignorAddress.Ioss
2020-11-23	- Added DDP integration chapter
2020-07-27	- Added ShippingValue to OrderShipment/OrderShipmentInfo command
2020-07-14	- Added OrderShipmentInfo command (Chapter 2.9)
2020-05-11	- Added XBS BACK export fields: o OrderReference o OrderDate o ExportCarrierName o ExportAwb o Products[x].ImgUrl o Products[x].Weight o Products[x].NonReturnable o Products[x].DaysForReturn
2020-04-15	- Added CZ to PPTT destinations - Brazil Tax ID required in ConsigneeAddress.Vat for destination BR - Norway VOEC number required in ConsignorAddress.Vat for destination NO
2020-01-10	Removed Labelformat from TrackShipment command.
2019-12-23	Added collect service. GetLocations command and widget. New optional field in OrderShipment: PudoLocationId
2019-10-10	Added GetProofOfExportReport command. Restructured order of commands.
2019-07-17	Added Eori and some new services.
2019-04-02	• Added the following services to the “Service” parameter of the OrderShipment command: “HEHDS” – Hermes Home – Sign “CPHD” – Collis Prive – No Sign “CPHDS” – Collis Prive – Sign • Added the GetshipmentPoeLabel command to generate a “Proof of Export” label
2019-02-20	Restructured the API requests. • Added the parameters InvoiceNumber and DeclarationType to the OrderShipment command. • Added “2.2 Label validations” table • Added “2.3 Service validations” table and country ISO value lists for services PPTT and PPLEU
2019-01-11	Added the following service codes to the OrderShipment API: "TRCK" – Spring Tracked "SIGN" – Spring Signature "UNTR"– Spring Untracked A short explanation on these services was also added.
2018-12-20	"CarrierTrackingUrl" parameter added to the OrderShipment, GetShipmentLabel, GetShipmentInvoice and TrackShipment command. Example: "CarrierTrackingUrl" : "http://tracking.carrier.com/LX210720682NL"

1 API Description

1.1 Introduction

The Spring XBS API allows users to retrieve and submit data. You can generate labels, reprint labels, void labels and request the tracking status of items. It is implemented using the REST principles. This document lists the available API endpoints and describes the resource endpoints, parameters and responses.

1.2 Environment URLs

An HTTPS connection is required. Access to the web service is limited to authorized clients only. PostNL will provide an API key for each customer which is submitted in the body of the request. An acceptance environment is available for integration purposes. Once the requests on the acceptance environment have been verified by the customer integration team, an API key for the production environment can be obtained.

Environment	Resource URL	Method
Acceptance	https://mtapi.net/?testMode=1	POST
Production	https://mtapi.net/	POST

1.3 JSON formatting

All information is send using JSON formatting. JSON is an industry standard documented under ECMA-404 and is documented on https://JSON.org/. The JSON header should look similar to:

```
{
   "Content-Type":" text/json"
}
```

1.4 Error Level Code List

The API response will contain a parameter called ErrorLevel. The following table gives a description of these error levels.

ErrorLevel	Error description
0	Command successfully completed without any errors
1	Command completed with errors. E.g., for OrderShipment, if shipment have been created with errors (in Error status)
10	Fatal error, command is not completed at all

1.5 Label changes

The label design and the data it contains has been agreed upon with many different postal parties. It is therefore not allowed to change the label provided by our API in any way. This means you may not change the .pdf or .zpl file or make changes to the physical label after it has been printed. Using the untampered label provided by our API will make sure your label complies with international agreements and not cause delays or issues in processing your shipment. If you have suggestions or wishes to change the label in any way please specify this request to your Spring contact person.

2 Resources

2.1 OrderShipment command

This resource can be used to generate labels. All value formats can be declared as a String.

Body Parameters

Name field	Description	Mandatory
Apikey	Apikey provided by Spring	M
Command	OrderShipment	M
Shipment Object		
LabelFormat	PDF (default) PNG ZPL300 – ZPL 300 dpi ZPL200 – ZPL 203 dpi ZPL – alias for ZPL300 EPL – EPL 203 dpi	O
ShipperReference	Reference number for the shipment. Only unique values accepted.	M
OrderReference	Order number as communicated to the consumer. Mandatory for commercial returns (BACK).	O
OrderDate	Date order was placed. Format: *yyyy-mm-dd. Optional for* commercial returns (BACK).	O
DisplayId	Field that can be used for an internal reference. Will also generate a 2nd barcode on the label if populated.	O
InvoiceNumber	Number that will be displayed on the commercial invoice	O
Service	See chapter **2.10 GetServices command** in this API manual for a list of possible and enabled services or ask your Spring customer service which service codes to use. Example value: *TRCK*	M
Weight	Total weight of the order	M
WeightUnit	kg is the default value. Accepted values: kg, lb	O
Length	Length dimension of the order	O
Width	Width dimension of the order	O
Height	Height dimension of the order	O
DimUnit	Accepted value is: cm	O
Value	Total value of the order	M/O
ShippingValue	Shipping costs charged to the receiver of the parcel	O
Currency	Currency of the value on the CN22/23 information. Default EUR	O
CustomsDuty	Accepted values: DDU, DDP. Default: DDU	O
Description	Description of the order	O
DeclarationType	Accepted values: SaleOfGoods, Documents, Gift, ReturnedGoods and CommercialSample. Default: SaleOfGoods	O
DangerousGoods	Accepted values: Y, N	
ExportCarrierName	Only applicable for service BACK, IMPRT and NOLABEL	O
ExportAwb	Only applicable for service BACK, IMPRT and NOLABEL	O
ConsignorAddress Object		
Name	Ship from name. Printed on label as signer of CN23.	M/O
Company	Ship from company name. Printed on label in FROM part.	O
AddressLine1	Ship from address line 1	O
AddressLine2	Ship from address line 2	O
AddressLine3	Ship from address line 3	O
City	Ship from city	O
State	Ship from state (2/3 letter ISO code)	O
Zip	Ship from zipcode (mandatory for countries using postal code system)	O
Country	Ship from country (2 letter ISO code)	O
Phone	Ship from telephone number	O
Email	Ship from Email	O
Vat	Ship from VAT number: - For destination Norway: Fill with VOEC number. - Used in Spring CLEAR iOSS+ & Union Oss solution	M/O
Eori	Ship from Eori.	M/O

	- For destination GB non-DDP: Fill with the GB123… Eori number. - Used in Spring CLEAR iOSS+ & Union Oss solution	
NlVat	Ship from NL VAT number (Used in Spring CLEAR iOSS+ & Union Oss solution)	M/O
EuEori	Ship from EU Eori number (Used in Spring CLEAR iOSS+ & Union Oss solution)	M/O
Ioss	Ship from IOSS number (Used in Spring CLEAR iOSS+ & Union Oss solution)	M/O
ConsigneeAddress Object		
Name	Ship to name	M
Company	Ship to company name	O
AddressLine1	Ship to address line 1	M
AddressLine2	Ship to address line 2	O
AddressLine3	Ship to address line 3	O
City	Ship to city	M
State	Ship to state (2/3 letter ISO code, mandatory for USA, Canada and Australia)	M/O
Zip	Ship to zipcode (mandatory for countries using postal code system)	M/O
Country	Ship to country (2 letter ISO code)	M
Phone	Ship to telephone number	M
Email	Ship to Email	M
Vat	Ship to VAT number (Mandatory for destination BR)	M/O
PudoLocationId	Id received from GetLocations command. Used for Collect services.	O
Products Array		
Description	Description of the product	M
Sku	Stock Keeping Unit	O
HsCode	Harmonized System Codes. 6-10 characters	M
OriginCountry	Origin country of the product (2 letter ISO code)	O
ImgUrl	URL to image of the product. Mandatory for commercial returns (BACK)	O
PurchaseUrl	URL to webshop.	O
Quantity	Quanity of the product	M
Value	Total value of the product(s) (Single Item Value * Quantity)	M
Weight	Total weight of the product(s) (Single Item Weight * Quantity)	M/O
DaysForReturn	Only applicable for commercials returns (BACK). Number of days the item can be returned by consumers. Defaults to customer setting if not specified.	O
NonReturnable	Only applicable for commercials returns (BACK). Accepted values: Y, N. Default is N. Specify Y if product is not eligible for a commercial return.	O

Notes concerning the response

- The **CarrierTrackingNumber** is always **TrackingNumber**.
- In case of non-fatal error (ErrorLevel=1) LabelType contains value “TEMP” and the LabelImage the contains a temporary label.

Example Request

```
{
  "Apikey": "<your-apikey-here>",
  "Command": "OrderShipment",
  "Shipment": {
    "LabelFormat": "PDF",
    "ShipperReference": "Reference_001",
    "DisplayId": "123450000",
    "InvoiceNumber": "678900000",
    "Service": "SIGN",
    "Weight": "0.85",
    "WeightUnit": "kg",
    "Length": "20",
    "Width": "10",
    "Height": "10",
    "DimUnit": "cm",
    "Value": "20",
    "Currency": "EUR",
    "CustomsDuty": "DDU",
    "Description": "CD",
    "DeclarationType": "SaleOfGoods",
    "DangerousGoods": "N",
    "ConsignorAddress": {
      "Name": "Jan Janssen",
      "Company": "Webshop JAN"
    },
    "ConsigneeAddress": {
      "Name": "John Doe",
      "Company": "",
      "AddressLine1": "20 W 34th St",
      "AddressLine2": "",
      "AddressLine3": "",
      "City": "New York",
      "State": "NY",
      "Zip": "10001",
      "Country": "US",
      "Phone": "+1-123-456-7890",
      "Email": "john@doe.com",
      "Vat": ""
    },
    "Products": [
      {
        "Description": "CD: The Postal Service - Give Up",
        "Sku": "CD1202",
        "HsCode": "852349",
        "OriginCountry": "GB",
        "ImgUrl": "http://url.com/cd-thepostalservicegiveup.jpg",
        "Quantity": "2",
        "Value": "20",
        "Weight": "0.8"
      }
    ]
  }
}
```

Example Response

```
{
"ErrorLevel" : 0,
"Error" : "",
"Shipment" : {
        "TrackingNumber" : "LX210720682NL",
        "ShipperReference" : "Reference_001",
        "DisplayId” : "123450000",
        "Service'" : "PPTT",
        "Carrier" : "PostNL Packet Tracked",
        "CarrierTrackingNumber" : "LX210720682NL",
        "CarrierLocalTrackingNumber" : "LX210720682NL",
        "CarrierTrackingUrl" : "http://tracking.carrier.com/LX210720682NL",
        "LabelFormat" : "PDF",
        "LabelType" : "Label",
        "LabelImage" : "<base64-encoded-string>",
        }
}
```

2.1.1 Label character limit and service validations

To produce good labels we have **soft** guidelines and **hard** requirements for some fields. Below table shows the maximum number of characters advised to send for each field and service. Carrier labels have limit space. Our system does not technically force these guidelines. We save all the data in our system but the label might not print all information if the guidelines are not followed. Actual numbers depends on the size of the characters. Requirements that are forced by our system or the carrier system are specified in **red**. N/A means the field is not used for the carrier.

Key/Service	PPLEU	PPLGE/GU	RM24/48(S)	PPTT	PPTR/NT	SEND(2)	ITCR	HEHDS	CPHD(S)	SC***	PPND/HD(S)
ConsignorCompany	30	60	25	30	30	NotPrinted	NotPrinted	40	25	NotPrinted	30
Name	35	50	30 (35Max)	30	30	20	60	20	35	35	35
Company	35	30	30 (35Max)	30	30	35	60	20	35	35	35
AddressLine1	35	50	30 (35Max)	30	30	35	60	Validation	35	35	35
AddressLine2	35	50	30 (35Max)	30	30	35	60	NoValidation	35	35	35
AddressLine3	35	50	30 (35Max)	30	30	35	60	NoValidation	35	35	35
City	35	50	30	30	30	35	60	Validation	35	35	35
State - Validation for	35 US,CA,AU	50 US,CA,AU	30 NoValidation	30 US,CA,AU	30 US,CA,AU	35 NoValidation	50 NoValidation	NotPrinted NoValidation	35 NoValidation	2 US	35 NoValidation
Zip - Validation for	20 NoValidation	20 NoValidation	20 NoValidation	20 NoValidation	20 NoValidation	6 Validation	5 Validation	5 Validation	20 NoValidation	20 NoValidation	20 NoValidation
ConsigneeCountry	PPLEU LIST	AllDest	GB	PPTT LIST	AllDest	ES, PT	IT	DE	FR	US	NL, BE
Weight	30kgMax*	30kgMax*	20kgMax	2kgMax	2kgMax	20kgMax	20kgMax	20kgMax	20kgMax	10.5kgMax	30kgMax
DisplayId	15	15	15	15	15	15	15	15	15	15	15
Description	NotPrinted	105	NotPrinted	20	20	NotPrinted	60	NotPrinted	NotPrinted	NotPrinted	NotPrinted
HsCode	NotPrinted	6	NotPrinted	25	25	NotPrinted	NotPrinted	NotPrinted	NotPrinted	NotPrinted	NotPrinted
Phone	15	15	NotPrinted	15	15	15	15	15	NotPrinted	NotPrinted	15
Email	NotPrinted	NotPrinted	NotPrinted	NotPrinted	NotPrinted	NotPrinted	NotPrinted	NotPrinted	NotPrinted	NotPrinted	NotPrinted

*is less for some ConsigneeCountry

Supported ConsigneeCountry lists

Service	ISO 2-letter country code LIST
PPTT	AU;AT;BE;BG;BR;BY;CA;CH;CN;CY;CZ;DK;DE;EE;ES;FI;FR;GB;GF;GI;GP;GR;HK;HR;HU;ID;IE;IL;IS;IT;JP;KR; LB;LT;LU;LV;MQ;MT;MY; NL;NO;NZ;PL;PT;RE;RO;RS;RU;SA;SE;SG;SI;SK;TH TR;US;
PPLEU	AT;BE;BG; CY;CZ;DK;DE;EE;FI;FR;GR;HR;HU;IE;LT;LU;LV;MT;NL;PL;RO;SE;SI; SK;

2.1.2 Routed Services

The services ‘TRCK’, ‘SIGN’, ‘UNTR’ & ‘CLLCT’ are services that route to the best end carrier services based on characteristics like destination country and weight.

2.1.3 Service IMPRT

Service IMPRT can be used to only import the order data for further processing at a later stage. No carrier is selected and no shipping label is generated. IMPRT will route to service NOLABEL.

2.1.4 Platform source

If your integration can be used for more than one of our customers, you are required to provide a value for the parameter Shipment.Source. Examples of this type of integration are platforms like Metapack, ShippyPro and MyParcel.com. By adding the source, our customer service can identify easily how the customer is integrated during IT related questions and resolve queries faster.

```
{
  "Apikey": "<your-apikey-here>",
  "Command": "OrderShipment",
  "Shipment": {
    "LabelFormat": "PDF",
    "Source": "<your-platform-name-here>",
    "ShipperReference": "Reference_001",
etc…
```

2.1.5 DDP integration

DDP (Delivery Duty Paid) is where the shipper pre-pays the customs duty and taxes that the consumer otherwise would have to pay on receipt of the shipment. DDP must be activated for your account. When a DDP shipment is requested we calculate the Taxes and Duties based on the provided data. It is therefore important that the provided data is correct.

With a DDP shipment the following fields are important in the OrderShipment API command:

- **Shipment.CustomsDuty** must be set to DDP. With every other value than DDP the shipment reverts to the default; DDU.
- **Shipment.ShippingValue** is the shipping costs charged to the consumer.
- **Shipment.ConsignorAddress.NlVat** is the shipper’s NL VAT number
- **Shipment.ConsignorAddress.EuEori** is the shipper’s EU Eori number
- **Shipment.ConsignorAddress.Ioss** Is the shipper’s IOSS number

The following fields are also required for a correct Taxes and Duties calculation.

- **Shipment.Products[x].HsCode**. An HS code with a minimum of 6 digits is required.
- **Shipment.Products[x].Description**
- **Shipment.Products[x].OriginCountry**
- **Shipment.Products[x].Quantity**
- **Shipment.Products[x].Value** Please note that this is the total value of this specific product. A Quantity of 5 and an single item product value of 10 results in Shipment.Products[x].Value of 50
- **Shipment.Products[x].Weight**

Spring will charge the taxes and duties to the shipper and, usually, the shipper charges the consumer during the order process. As a shipper, you can calculate the taxes and duties before generating a label. This is done by integrating the OrderShipmentInfo command in chapter 2.9 of this API manual. Integrating 2.9 OrderShipmentInfo is optional. Spring also calculates the taxes and duties during OrderShipment label generation. The tax and duties calculation is returned in the response of both the OrderShipment and OrderShipmentInfo command. The response block can be found at **Shipment.DdpInfo** A full description of this response block can be found at chapter 2.9 OrderShipmentInfo

Example Request – DDP destination EU

```
{
  "Apikey": "<your-apikey-here>",
  "Command": "OrderShipment",
  "Shipment": {
    "LabelFormat": "PDF",
    "ShipperReference": "MyDDPshipment01",
    "Service": "TRCK",
    "Weight": "0.8",
    "WeightUnit": "kg",
    "Length": "20",
    "Width": "10",
    "Height": "10",
    "DimUnit": "cm",
    "Value": "25",
    "Currency": "EUR",
    "CustomsDuty": "DDP",
    "Description": "CDs",
    "DeclarationType": "SaleOfGoods",
    "ShippingValue": 5,
    "ConsignorAddress": {
      "Name": "John Doe",
      "Company": "Webshop JOHN",
      "Vat": "123456789",
      "Eori": "GB123456789",
      "NlVat": "NL000099998B57",
      "EuEori": "NL123456789",
      "Ioss": "IM123456789"
    },
    "ConsigneeAddress": {
      "Name": "Kees Konsument",
      "Company": "",
      "AddressLine1": " Waldorpstraat 3",
      "City": "Den Haag",
      "Zip": "2521 CA",
      "Country": "NL",
      "Phone": "+31-123456789",
      "Email": "kees@example.com"
    },
    "Products": [
      {
        "Description": "CD: The Postal Service - Give Up",
        "HsCode": "852349",
        "OriginCountry": "GB",
        "Quantity": "2",
        "Value": "20",
        "Weight": "0.8"
      }
    ]
  }
}
```

Example Request – DDP destination GB

```
{
  "Apikey": "<your-apikey-here>",
  "Command": "OrderShipment",
  "Shipment": {
    "LabelFormat": "PDF",
    "ShipperReference": "MyDDPshipment02",
    "Service": "TRCK",
    "Weight": "0.85",
    "WeightUnit": "kg",
    "Length": "20",
    "Width": "10",
    "Height": "10",
    "DimUnit": "cm",
    "Value": "25",
    "Currency": "EUR",
    "CustomsDuty": "DDP",
    "Description": "CDs",
    "DeclarationType": "SaleOfGoods",
    "ShippingValue": 5,
    "ConsignorAddress": {
      "Name": "Jan Janssen",
      "Company": "Webshop JAN",
      "Vat": "123456789000"
    },
    "ConsigneeAddress": {
      "Name": "John Doe",
      "Company": "",
      "AddressLine1": "Brushfield St 1",
      "City": "London",
      "Zip": "E1 6AN",
      "Country": "GB",
      "Phone": "+44-123456789",
      "Email": "john@doe.com"
    },
    "Products": [
      {
        "Description": "CD: The Postal Service - Give Up",
        "HsCode": "852349",
        "OriginCountry": "NL",
        "Quantity": "2",
        "Value": "20",
        "Weight": "0.8"
      }
    ]
  }
}
```

2.1.6 BACK commercial returns integration

The Spring XBS system can generate return labels. The return labels can be used by consumers to return (part of) their order to the shipper / webshop. The consumer puts the return label on the original box and drops off the parcel at the designated carrier drop-off point.

Note: The return labels you receive from Spring XBS are produced in a separate system, dedicated to returns, called Spring BACK. The Spring XBS system can provide return labels and tracking with your existing XBS integration, so if that is all you need we recommend this easy integration with Spring XBS. If, however, you would like us to create a returns web portal where consumer can register their returns, we recommend you contact your Spring support team to receive instructions for a different integration.

Integration notes:

- The return service needs to be added to your account before you can start using it. This activation will be done by your Spring support team.
- Test mode is not available for this service. During integration use endpoint https://mtapi.net/?testMode=0 and not https://mtapi.net/?testMode=1.
- In the OrderShipment API request use "Service": "**BACK**",
- If you want to get both a forward and a return shipping label you need to perform the OrderShipment Command twice: One time with the forward Service and one time with Service BACK. This is the case when you, for example, want to put the return label inside the box when you ship the order.
- The ConsignorAddress is the shipper / webshop address and ConsigneeAddress is the consumer address. This is the same as with a forward shipping label.
- It is mandatory to provide a value for either ShipperReference or the OrderReference field. This value is usually printed on the return label. When OrderReference is provided we use OrderReference, else ShipperReference. You can also provide both. In case you provide both you can use ShipperReference in the system for tracking while OrderReference is printed on the return label. ShipperReference needs to be unique for every label generated by our Spring XBS system so if you are generating both a forward and a return label the ShipperReference still needs to be unique even though it can be for the same order.
- All forward shipping labels are approximately A6 size (10 cm by 15 cm) but most return labels are A4 size (21cm by 30cm). The reason is that in some cases the return label is printed by the consumers on a home printer for which the default is A4.
- For our return service we use a variety of carriers. Some of these carriers have a longer response time to generate the return label. Therefore, we advise to set the time-out on your API request to at least 100 seconds.
- The TrackShipment Command can be used to track the return shipment. Please consider these settings in your operations.
- We will keep adding countries, but currently our system can create return labels for consumers in these countries: ['AT', 'BE', 'CZ', 'DE', 'DK', 'EE', 'ES', 'FI', 'FR', 'GR', 'HU', 'HR', 'IE', 'IT', 'LT', 'LU', 'LV', 'MT', 'NL', 'PL', 'PT', 'RO', 'SE', 'SI', 'SK']
- If you want to create a link between the forward and return label you can take the Shipment.TrackingNumber value from the OrderShipment response of the forward label and put it in the OrderShipment request field Shipment.ExportAwb of the return label.

Example Request

```
{
  "Apikey": "<your-apikey-here>",
  "Command": "OrderShipment",
  "Shipment": {
    "LabelFormat": "PDF",
    "ShipperReference": "",
    "OrderReference": "Order12345678",
    "DisplayId": "123450000",
    "InvoiceNumber": "678900000",
    "Service": "BACK",
    "Weight": "0.85",
    "WeightUnit": "kg",
    "Length": "20",
    "Width": "10",
    "Height": "10",
    "DimUnit": "cm",
    "Value": "20",
    "Currency": "EUR",
    "CustomsDuty": "DDU",
    "Description": "CD",
    "DeclarationType": "SaleOfGoods",
    "DangerousGoods": "N",
    "ExportCarrierName": "PostNL Packet Tracked",
    "ExportAwb": "LX210720682NL",
    "ConsignorAddress": {
      "Name": "Lukas Müller",
      "Company": "Webshop Müller"
    },
    "ConsigneeAddress": {
      "Name": "Kees Konsument",
      "Company": "",
      "AddressLine1": " Waldorpstraat 3",
      "City": "Den Haag",
      "Zip": "2521 CA",
      "Country": "NL",
      "Phone": "+31-123456789",
      "Email": "kees@example.com"
    },
    "Products": [
      {
        "Description": "CD: The Postal Service - Give Up",
        "Sku": "CD1202",
        "HsCode": "852349",
        "OriginCountry": "DE",
        "ImgUrl": "http://url.com/cd-thepostalservicegiveup.jpg",
        "Quantity": "2",
        "Value": "20",
        "Weight": "0.8"
      }
    ]
  }
}
```

2.2 TrackShipment command

This resource can be used to track shipments. See chapter 2.2.1 if you prefer a Webhook integration for tracking.

Body Parameters

Name field	Description	Mandatory
Apikey	Apikey provided by Spring	M
Command	TrackShipment	M
Shipment Object		
TrackingNumber	Only one element is allowed: TrackingNumber or ShipperReference	O/M
ShipperReference	Only one element is allowed: TrackingNumber or ShipperReference	O/M

TrackShipment Example Request

```
{
  "Apikey": "<your-apikey-here>",
  "Command": "TrackShipment",
  "Shipment": {
    "TrackingNumber": "",
    "ShipperReference": "Reference_001"
  }
}
```

TrackShipment Example Response

```
{
  "ErrorLevel": 0,
  "Error": "",
  "Shipment": {
    "TrackingNumber": "LX210720682NL",
    "ShipperReference": "ref123456789",
    "DisplayId": "123450000",
    "Service'": "PPTT",
    "Carrier": "PostNL Packet Tracked",
    "CarrierTrackingNumber": "LX210720682NL",
    "CarrierLocalTrackingNumber": "LX210720682NL",
    "CarrierTrackingUrl": "http://tracking.carrier.com/LX210720682NL",
    "Weight": "0.85",
    "WeightUnit": "kg",
    "ConsigneeAddress": {
      "Name": "John Doe",
      "Company": "John Doe’s Company",
      "City": "New York",
      "Zip": "10001",
      "Country": "US"
    },
    "Events": [
      {
        "DateTime": "2018-01-10 12:00:00",
        "Country": "US",
        "Code": 21,
        "Description": "IN TRANSIT",
        "CarrierCode": "B01",
        "CarrierDescription": "Accepted"
      },
      {
        "DateTime": "2018-01-11 12:00:00",
        "Country": "US",
        "Code": 100,
        "Description": "DELIVERED",
        "CarrierCode": "I01",
        "CarrierDescription": "Delivered, expl. recorded"
      }
    ]
  }
}
```

2.2.1 Tracking Webhook

Next to the TrackShipment API, the Spring XBS system also offers a Webhook tracking solution. This allows for automated callbacks whenever new tracking events on parcels occur. To set up a Webhook connection, the following information is required:

- **URL –** The URL that we will send the callbacks to.
- **Format –** Format in which the data will be sent to URL. *Options: JSON (default), x-www-form-urlencoded, query*

The webhook cannot be configured via the API. Please contact Spring to have it configured and share the details as explained above.

Webhook Example Request in JSON Format

```
{
    "trackingNumber": "LT580553684NL",
    "event": "parcel_new_tracking_event",
    "data": {
        "time": "1611754011",
        "date": "2021-01-27",
        "location": {
            "address1": "Via Manzoni, 9",
            "address2": "",
            "city": "PARCEL SCANNED",
            "state": "ITALY",
            "zip": "",
            "country": "IT",
            "companyName": "Spring GDS - IT"
        },
        "action": "IN TRANSIT",
        "code": "E03"
    }
}
```

Webhook Example Request in x-www-form-urlencoded and Query Format

```
trackingNumber=LT580553684NL0&event=parcel_new_tracking_event&data%5Btime%5D=1611754011&data
%5Bdate%5D=2021-01-
27&data%5Blocation%5D%5Baddress1%5D=Via%20Manzoni%2C%209&data%5Blocation%5D%5Baddress2%5D=&d
ata%5Blocation%5D%5Bcity%5D=PARCEL%20SCANNED&data%5Blocation%5D%5Bstate%5D=ITALY&data%5Bloca
tion%5D%5Bzip%5D=&data%5Blocation%5D%5Bcountry%5D=IT&data%5Blocation%5D%5BcompanyName%5D=Spr
ing%20GDS%20-%20IT&data%5Baction%5D=IN%20TRANSIT&data%5Bcode%5D=E03
```

2.3 VoidShipment command

This resource can be used to cancel the generated labels.

Body Parameters

Name field	Description	Mandatory
Apikey	Apikey provided by Spring	M
Command	VoidShipment	M
Shipment Object		
TrackingNumber	Only one element is allowed: TrackingNumber or ShipperReference	O/M
ShipperReference	Only one element is allowed: TrackingNumber or ShipperReference	O/M

Example Request

```
{
  "Apikey": "<your-apikey-here>",
  "Command": "VoidShipment",
  "Shipment": {
    "TrackingNumber": "",
    "ShipperReference": "Reference_001"
  }
}
```

Example Response

```
{
  "ErrorLevel": 0,
  "Error": "",
  "Shipment": {
    "TrackingNumber": "LX210720682NL",
    "ShipperReference": "ref123456789"
  }
}
```

2.4 GetShipmentLabel command

This resource can be used to reprint labels.

Body Parameters

Name field	Description	Mandatory
Apikey	Apikey provided by Spring	M
Command	GetShipmentLabel	M
Shipment Object		
LabelFormat	**PDF** (default) **PNG** **ZPL300** – ZPL 300 dpi **ZPL200** – ZPL 203 dpi **ZPL** – alias for ZPL300 **EPL** – EPL 203 dpi	O
TrackingNumber	Only one element is allowed: TrackingNumber or ShipperReference	O/M
ShipperReference	Only one element is allowed: TrackingNumber or ShipperReference	O/M

Example Request

```
{
  "Apikey": "<your-apikey-here>",
  "Command": "GetShipmentLabel",
  "Shipment": {
    "LabelFormat": "PDF",
    "TrackingNumber": "",
    "ShipperReference": "Reference_001"
  }
}
```

Example Response

```
{
  "ErrorLevel": 0,
  "Error": "",
  "Shipment": {
    "TrackingNumber": "LX210720682NL",
    "ShipperReference": "Reference_001",
    "DisplayId": "123450000",
    "Service'": "PPTT",
    "Carrier": "PostNL Packet Tracked",
    "CarrierTrackingNumber": "LX210720682NL",
    "CarrierLocalTrackingNumber": "LX210720682NL",
    "CarrierTrackingUrl": "http://tracking.carrier.com/LX210720682NL",
    "LabelFormat": "PDF",
    "LabelType": "Label",
    "LabelImage": "<base64-encoded-string>",

  }
}
```

2.5 GetShipmentInvoice command

This resource can be used to generate the shipment invoice.

Body Parameters

Name field	Description	Mandatory
Apikey	Apikey provided by Spring	M
Command	GetShipmentInvoice	M
Shipment Object		
LabelFormat	**PDF** (default) **PNG** **ZPL300** – ZPL 300 dpi **ZPL200** – ZPL 203 dpi **ZPL** – alias for ZPL300 **EPL** – EPL 203 dpi	O
TrackingNumber	Only one element is allowed: TrackingNumber or ShipperReference	O/M
ShipperReference	Only one element is allowed: TrackingNumber or ShipperReference	O/M

Example Request

```
{
  "Apikey": "<your-apikey-here>",
  "Command": "GetShipmentInvoice",
  "Shipment": {
    "LabelFormat": "PDF",
    "TrackingNumber": "",
    "ShipperReference": "Reference_001"
  }
}
```

Example Response

```
{
  "ErrorLevel": 0,
  "Error": "",
  "Shipment": {
    "TrackingNumber": "LX210720682NL",
    "ShipperReference": "Reference_001",
    "DisplayId": "123450000",
    "Service'": "PPTT",
    "Carrier": "PostNL Packet Tracked",
    "CarrierTrackingNumber": "LX210720682NL",
    "CarrierLocalTrackingNumber": "LX210720682NL",
    "CarrierTrackingUrl": "http://tracking.carrier.com/LX210720682NL",
    "LabelFormat": "PDF",
    "LabelType": "INVOICE",
    "LabelImage": "<base64-encoded-string>"
  }
}
```

2.6 GetShipmentPoeLabel command

This resource can be used to generate a “Proof of Export” label.

Body Parameters

Name field	Description	Mandatory
Apikey	Apikey provided by Spring	M
Command	GetShipmentPoeLabel	M
Shipment Object		
LabelFormat	PDF (default) PNG ZPL300 – ZPL 300 dpi ZPL200 – ZPL 203 dpi ZPL – alias for ZPL300 EPL – EPL 203 dpi	O
TrackingNumber	Only one element is allowed: TrackingNumber or ShipperReference	O/M
ShipperReference	Only one element is allowed: TrackingNumber or ShipperReference	O/M

Example Request

```
{
  "Apikey": "<your-apikey-here>",
  "Command": "GetShipmentPoeLabel",
  "Shipment": {
    "LabelFormat": "PDF",
    "TrackingNumber": "",
    "ShipperReference": "Reference_001"
  }
}
```

Example Response

```
{
  "ErrorLevel": 0,
  "Error": "",
  "Shipment": {
    "TrackingNumber": "LX210720682NL",
    "ShipperReference": "Reference_001",
    "DisplayId": "123450000",
    "Service'": "PPTT",
    "Carrier": "PostNL Packet Tracked",
    "CarrierTrackingNumber": "LX210720682NL",
    "CarrierLocalTrackingNumber": "LX210720682NL",
    "CarrierTrackingUrl": "http://tracking.carrier.com/LX210720682NL",
    "LabelFormat": "PDF",
    "LabelType": "POE",
    "LabelImage": "<base64-encoded-string>"
  }
}
```

2.7 GetProofOfExportReport command

This resource can be used to track shipments.

Body Parameters

Name field	Description	Mandatory
Apikey	Apikey provided by Spring	M
Command	GetProofOfExportReport	M
Shipment Object		
Report.DateRange	Array of start and end date, "**YYYY-MM-DD**" format	M
DateRangeType	**c** – create date (default) **a** – accept date **e** – export date	
Country	If empty all countries will be returned. ISO 2-letter country code. Example: US	O
ReportFormat	**CSV** (default) **XLS** **PDF**	O

Example Request

```
{
  "Apikey": "<your-apikey-here>",
  "Command": "GetProofOfExportReport",
  "Report": {
    "DateRange": [
      "2019-09-01",
      "2019-09-02"
    ],
    "DateRangeType": "c",
    "Country": "",
    "ReportFormat": "XLS"
  }
}
```

Example Response

```
{
  "ErrorLevel": 0,
  "Error": "",
  "Report": {
    "DateRangeType": "c",
    "DateRange": [
      "2019-09-01",
      "2019-09-02"
    ],
    "Country": "",
    "ReportFormat": "XLS",
    "ReportContent": "base64-encoded-report-data",
    "ReportRows": [
      ,
      {
        "TrackingNumber": "LX210720682NL",
        "ShipperReference": "ref123456789",
        "Weight": "0.45",
        "WeightUnit": "kg",
        "Value": "10",
        "Currency": "EUR",
        "ConsignorAddress": {
          "Name": "Xxxx1 Xxxxxx1",
          "Company": "Xxxx1 LTD",
          "AddressLine1": "Address1 XXX 1",
          "AddressLine2": "apt1 1X",
          "AddressLine3": "",
          "City": "Jersey",
          "State": "NJ",
          "Zip": "10002",
          "Country": "US",

        },
        "ConsigneeAddress": {
          "Name": "Xxxx Xxxxxx",
          "Company": "Xxxx LTD",
          "AddressLine1": "Address XXX 1",
          "AddressLine2": "apt 1X",
          "AddressLine3": "",
          "City": "New York",
          "State": "NY",
          "Zip": "10001",
          "Country": "US",

        },
        "CreateDateTime": "2018-01-10 12:00:00",
        "AcceptDateTime": "2018-01-11 12:00:00",
        "ExportDateTime": "2018-01-12 12:00:00"
      },
      ...
    ]
  }
}
```

2.8 GetLocations command

This resource can be used to get Collect locations (PUDO points).

- Currently supported destinations: NL, BE, DE.
- From the API response "Id" and "Service" of chosen location should be transferred to "OrderShipment" API command in the fields Shipment.ConsigneeAddress.PudoLocationId and Shipment.Service.

Body Parameters

Name field	Description	Mandatory
Apikey	Apikey provided by Spring	M
Command	GetLocations	M
Location Object		
Country	Ship to country (2 letter ISO code)	M
Zip	Ship to zip code	M

Example Request

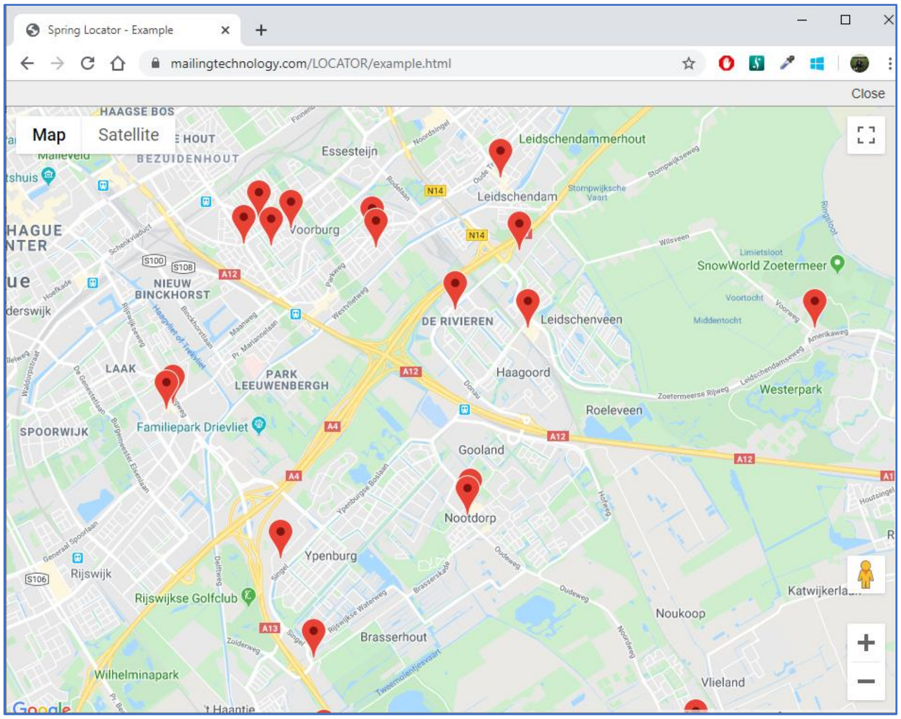
```
{
  "Apikey": "<your-apikey-here>",
  "Command": "GetLocations",
  "Location": {
    "Country": "NL",
    "Zip": "2132WT"
  }
}
```

Example Response

```
{
  "ErrorLevel": 0,
  "Location": [
    {
      "Id": "176227",
      "Service": "CLLCT",
      "Carrier": "PostNL",
      "Name": "Primera Sanders",
      "Address1": "Polderplein 3",
      "Address2": null,
      "City": "Hoofddorp",
      "State": null,
      "Zip": "2132BA",
      "CountryCode": "NL",
      "Latitude": 52.30329367,
      "Longitude": 4.69476214,
      "BusinessHours": "Mon: 11:00-18:00, Tue: 9:00-18:00, Wed: 9:00-18:00, Thu: 9:00-18:00,
Fri: 9:00-21:00, Sat: 9:00-18:00, Sun: 12:00-17:00"
    },
    …
  ]
}
```

2.8.1 GetLocations widget

There is a widget available for Collect locations which can be integrated in a website.



Widget url structure:

https://mailingtechnology.com/LOCATOR/?country=***<country-code>***&zip=***<zipcode>***

Example widget url:

https://mailingtechnology.com/LOCATOR/?country=NL&zip=2595AK

The widget can be called (opened in iframe) from a web page with the required country and zip-code transferred in the url. When the consumer selects a location the data is sent to the page that called the widget using the postMessage() method, used to send cross-domain messages. The web page should accept and parse this message listening window 'message' event.

The widget sends JSON-encoded location's data like name, address, id and Spring service code to the parent page. The web page should parse it and at least get Service and Id. These are used to create a parcel in XBS.

See the page source of an example widget integration:

https://mailingtechnology.com/LOCATOR/example.html

2.9 OrderShipmentInfo command

This resource can be used to get information on a potential shipment before running the OrderShipment command. The main use case of this command is to display a Tax and Duties calculation in the checkout basket of a web shop. This is especially useful for customers that are using a DDP service. Please note that returning Tax and Duties information is a pay-per-request service and needs to be activated for the account. When your account is activated for Tax and Duties Calculation (DDP services) the value "DDP" can be specified for request key Shipment.CustomsDuty. The Tax and Duties information is contained in the Shipment.DdpInfo block of the response. Some notes on the response:

- calculatedTotal = productRawTotal + totalTax + totalDuty + ShippingValue
- hs6codeDescription is the official description of the provided HsCode

The OrderShipmentInfo also returns the final mile carrier service selected when a routed service like TRCK, SIGN and UNTR is specified. See Shipment.Service and Shipment.Carrier in the response. This part of the command does not require activation and is free to use.

The command has the exact same structure as the OrderShipment command in Chapter 2.1. Running the OrderShipmentInfo not generate a label but instead return shipment information. The request parameter Shipment.ShippingValue is only used in the OrderShipmentInfo command. This is the shipping costs charged to the receiver of the parcel.

Body Parameters

The request body contains the same body parameters as the OrderShipment command. See chapter 2.1 for a description of the body parameters. Please note that in the Shipment.Products list the following fields are required for Tax & Duty calculation:

- Shipment.Products[x].Description
- Shipment.Products[x].HsCode
- Shipment.Products[x].OriginCountry
- Shipment.Products[x].Quantity
- Shipment.Products[x].Value
- Shipment.Products[x].Weight

Example Request

```
{
  "Apikey": "{{XBSkey}}",
  "Command": "OrderShipmentInfo",
  "Shipment": {
    "LabelFormat": "PDF",
    "ShipperReference": "MyDDPshipment02",
    "Service": "TRCK",
    "Weight": "0.85",
    "WeightUnit": "kg",
    "Length": "20",
    "Width": "10",
    "Height": "10",
    "DimUnit": "cm",
    "Value": "25",
    "Currency": "EUR",
    "CustomsDuty": "DDP",
    "Description": "CDs",
    "DeclarationType": "SaleOfGoods",
    "ShippingValue": 5,
    "ConsignorAddress": {
      "Name": "Jan Janssen",
      "Company": "Webshop JAN",
      "Vat": "123456789000",
      "Eori": "GB123456789000"
    },
    "ConsigneeAddress": {
      "Name": "John Doe",
      "Company": "",
      "AddressLine1": "Brushfield St 1",
      "City": "London",
      "Zip": "E1 6AN",
      "Country": "GB",
      "Phone": "+44-123456789",
      "Email": "john@doe.com"
    },
    "Products": [
      {
        "Description": "CD: The Postal Service - Give Up",
        "HsCode": "852349",
        "OriginCountry": "NL",
        "Quantity": "2",
        "Value": "20",
        "Weight": "0.8"
      }
    ]
  }
}
```

Example Response

```
{
  "ErrorLevel": 0,
  "Shipment": {
    "ShipperReference": "Reference_011",
    "DisplayId": "123450000",
    "Service": "RM48S",
    "Carrier": "Royal Mail Tracked 48 - Sign",
    "DdpInfo": {
      "requestId": "5520bcc4-6548-41c4-8194-63a04c5fbde1",
      "landedCost": {
        "requestId": "fbe09fad-dca4-499a-b4a7-ba240583f28e",
        "sourceCountry": "ES",
        "sourceCurrency": "EUR",
        "destinationCountry": "GB",
        "destinationIP": null,
        "destinationCurrency": "GBP",
        "convertCurrency": true,
        "currencyExchange": "EUR:GBP",
        "exchangeRate": 0.9075492625,
        "rateLastUpdated": "2020-12-29T09:09:57.314Z",
        "vwUnitsMetric": true,
        "shippingValue": {
          "sourceValue": 6,
          "destinationValue": 5.45
        },
        "products": [
          {
            "hs6code": "852349",
            "commodityCode": "8523499000",
            "value": {
              "sourceValue": 20,
              "destinationValue": 18.15
            },
            "description": "CD: THE POSTAL SERVICE - GIVE UP",
            "percentageDutyRate": 0,
            "volumetricDutyRate": 0,
            "itemDutyRate": 0,
            "additionalDutyRate": null,
            "taxRate": 0.2,
            "dutyValue": {
              "sourceValue": 0,
              "destinationValue": 0
            },
            "taxValue": {
              "sourceValue": 5.2,
              "destinationValue": 4.72
            },
            "productTotal": {
              "sourceValue": 31.2,
              "destinationValue": 28.32
            },
            "hs6codeDescription": "electric chicago blues"
          }
        ],
        "productRawTotal": {
          "sourceValue": 20,
          "destinationValue": 18.15
        },
        "calculatedTotal": {
          "sourceValue": 31.2,
          "destinationValue": 28.32
        },
        "totalTax": {
          "sourceValue": 5.2,
          "destinationValue": 4.72
        },
        "totalDuty": {
          "sourceValue": 0,
          "destinationValue": 0
        },
        "dutyDeminimisValue": {
          "sourceValue": 148.75,
          "destinationValue": 135
        },
        "taxDeminimisValue": {
          "sourceValue": 16.53,
          "destinationValue": 15
        },
        "isError": false,
        "errorCode": 0,
        "notes": "",
        "generatedDate": "2020-12-29T10:09:54.221Z"
      }
    }
  }
}
```

.

2.10 GetServices command

This resource can be used to list the services that have been enabled for your account. In the API response the **AllowedServices** shows the list of services that have been enabled for your account and which can be used in the OrderShipment command. **AllowedSpringClear** returns whether Spring Clear has been enabled for your account. **List** gives a description of all services in our system.

Body Parameters

Name field	Description	Mandatory
Apikey	Apikey provided by Spring	M
Command	GetServices	M

Example Request

```
{
  "Apikey": "<your-apikey-here>",
  "Command": "GetServices"
}
```

Example Response

```
{
  "ErrorLevel": 0,
  "Services": {
    "AllowedServices": [
      "TRCK",
      "SIGN",
      "UNTR",
      "CLLCT",
      "BACK"
    ],
    "AllowedSpringClear": true,
    "List": {
      "TRCK": "TRACKED",
      "SIGN": "SIGNATURED",
      "UNTR": "UNTRACKED",
      "CLLCT": "SPRING COLLECT SERVICE",
      "EXPR": "EXPRESS",
      "IMPRT": "ORDER IMPORT",
      "BACK": "SPRING BACK RETURNS",
      "PPLEU": "PostNL Parcel EU",
      "PPND": "PostNL Parcel Benelux - No sign",
      "PPNDS": "PostNL Parcel Benelux - Sign",
      "PPHD": "PostNL Parcel Benelux - No sign, no neighbor",
      "PPHDS": "PostNL Parcel Benelux - Sign, no neighbor",
      "PPLUP": "PostNL Parcel Benelux - UPU",
      "PPLGE": "PostNL Parcel GlobalPack EMS",
      "PPLGU": "PostNL Parcel GlobalPack UPU",
      "PPLEP": "PostNL Parcel EPG",
      "PPNEU": "PostNL Parcel EPG NonEU",
      "PPLLW": "PostNL Lightweight China",
      "PPLCS": "PostNL Collect Service",
      "PPTT": "PostNL Packet Tracked",
      "PPTR": "PostNL Packet Registered",
      "PPNT": "PostNL Packet Non Tracked",
      "PPBBT": "PostNL Packet Boxable Bag & Trace",
      "PPBT": "PostNL Packet Bag & Trace",
      "PPBTT": "PostNL Packet Boxable Tracked",
      "PPBNT": "PostNL Packet Boxable Non Tracked",
      "RM24": "Royal Mail Tracked 24 - No sign",
      "RM24S": "Royal Mail Tracked 24 - Sign",
      "RM48": "Royal Mail Tracked 48 - No sign",
      "RM482": "Royal Mail Tracked 48 - No sign 2",
      "RM48S": "Royal Mail Tracked 48 - Sign",
      "SEND": "Sending Mainland",
      "SEND2": "Sending Islands",
      "ITCR": "Italian Post Crono",
      "ITCRX": "Italian Post Crono Express",
      "DPDDE": "DPD DE",
      "HEHDS": "Hermes - Sign",
      "HEDCS": "Hermes Germany Collect Service",
      "CPHD": "Colis Prive - No sign",
      "CPHDS": "Colis Prive - Sign",
      "SCST": "Spring Com Standard - No sign",
      "SCSTS": "Spring Com Standard - Sign",
      "SCEX": "Spring Com Express - No sign",
      "SCEXS": "Spring Com Express - Sign",
      "UPGRS": "USA Parcel Grnd - Sign",
      "UPGR": "USA Parcel Grnd - No sign",
      "UPEXS": "USA Parcel Exp - Sign",
      "UPEX": "USA Parcel Exp - No sign",
      "UPMAS": "USA Parcel Max - Sign",
      "UPMA": "USA Parcel Max - No sign",
      "UDGS": "USA Parcel Ground DG - Sign",
      "UPDG": "USA Parcel Ground DG - No sign",
      "UPDGS": "USA Parcel Plus Ground DG - Sign",
      "UPPDG": "USA Parcel Plus Ground DG - No sign",
      "PACHD": "Packeta",
      "MABNT": "MailAlliance Boxable Non Tracked",
      "ATEHD": "Austrian Post - Eurodis Home Delivery",
      "BACKT": "Spring BACK Tracked",
      "NOLABEL": "NO LABEL"
    }
  }
}
```

2.11 CreateGroup command

This resource can be used to make groups and generate group labels. Successfully creating groups means material can be processed quicker in the Spring GDS operations. The group labels need to be attached to the container. Technical requirement for this integration is that the Weight provided in the OrderShipment command is accurate. An operational requirement for this integration is that you need be able to tell which parcel goes in which bag / pallet. The groups created via this API command can be viewed in the XBS portal. If you need to make changes to a group you can remove the group with the VoidGroup command and create the group again. Making changes to a group is possible until the group is processed in the Spring GDS operations.

Body Parameters

Name field	Description	Mandatory
Apikey	Apikey provided by Spring	M
Command	CreateGroup	M
	Group Object	
LabelFormat	**PDF** (default) **PNG** **ZPL300** – ZPL 300 dpi **ZPL200** – ZPL 203 dpi **ZPL** – alias for ZPL300 **EPL** – EPL 203 dpi	O
GroupShipperReference	Reference number for the group. Only unique values accepted.	O
ReceptacleType	**BG** – Bag (default) **PB** – Pallet box **IB** – IPC Pallet box	O
ReceptacleWeightGrams	Weight of receptable in grams.	O
Boxable	If all parcel conform to boxable dimensions and there are commercial agreements for a boxable group on the service and destination combination the value **Y** can be used. Accepted values: **Y** or **N**. Default is **N**	O
Mixed	Can be used for Mixed destinations. Accepted values: **Y** or **N**. Default is **N**	O
Priority	If there are commercial agreements for a non-priority / economy group for a service and destination combination the value **N** can be used. Accepted values: **Y** or **N**. Default is **Y**.	O
Items	Array / list of parcel barcodes included in the group. TrackingNumber in OrderShipment command.	M

Example Request

```
{
  "Command": "CreateGroup",
  "Apikey": "<your-apikey-here>",
  "Group": {
    "LabelFormat": "PDF",
    "GroupShipperReference": "GroupReference-5645",
    "ReceptacleType": "BG",
    "ReceptacleWeightGrams": 200,
    "Boxable": "N",
    "Mixed": "N",
    "Priority": "Y",
    "Items": [
      "UD906023786NL",
      "UD906023738NL",
      "UD906023707NL",
      "UD906023769NL",
      "UD906023755NL",
      "UD906023772NL",
      "UD906023790NL"
    ]
  }
}
```

Example Response

```
{
  "ErrorLevel": 0,
  "Group": {
    "Type": "Bag",
    "Carrier": "PostNL Packet Non Tracked",
    "Boxable": "N",
    "GroupTrackingNumber": "PTLISAAUNNBGESUK0900000001010",
    "GroupShipperReference": "GroupReference-5645",
    "ReceptacleType": "BG",
    "Priority": "Y",
    "ReceptacleWeightGrams": 200,
    "TotalWeight": 3.222,
    "TotalWeightUnit": "kg",
    "Mixed": "N",
    "LabelFormat": "PDF",
    "Label": "<base64-encoded-string>"
  }
}
```

2.12 VoidGroup command

This resource can be used to delete a group created by the CreateGroup command. A response with ErrorLevel 0 means the group has successfully been deleted.

Body Parameters

Name field	Description	Mandatory
Apikey	Apikey provided by Spring	M
Command	VoidGroup	M
	Group Object	
GroupTrackingNumber	Only one element is allowed: GroupTrackingNumber or GroupShipperReference	M/O
GroupShipperReference	Only one element is allowed: GroupTrackingNumber or GroupShipperReference	M/O

Example Request

```
{
  "Command": "VoidGroup",
  "Apikey": "<your-apikey-here>",
  "Group": {
    "GroupTrackingNumber": "",
    "GroupShipperReference": "GroupReference-5645"
  }
}
```

Example Response

```
{
    "ErrorLevel": 0,
    "Group": {
        "GroupTrackingNumber": "",
        "GroupShipperReference": " GroupReference-5645"
    }
}
```

3 Tracking Event Codes List

The following table gives a description for the tracking event code. This table may be extended in the future.

Codes	Event description
0	PARCEL CREATED
20	ACCEPTED
21	IN TRANSIT
2101	IN TRANSIT - EXPORTED
25	END OF TRACKING UPDATES
31	DELIVERY EXCEPTION
40	IN CUSTOMS
41	CUSTOMS EXCEPTION
91	DELIVERY ATTEMPTED
92	DELIVERY AWAITING COLLECTION
93	DELIVERY SCHEDULED
100	DELIVERED
101	DELIVERED TO DESTINATION COUNTRY
111	LOST OR DESTROYED
11101	LOST
11102	DESTROYED
124	RETURN IN TRANSIT
12401	RETURN IN TRANSIT - REFUSED
12402	RETURN IN TRANSIT - UNDELIVERABLE
12403	RETURN IN TRANSIT - DAMAGED
12404	RETURN IN TRANSIT – NOT COLLECTED
12405	RETURN IN TRANSIT – ACCORDING TO AGREEMENT
125	RETURN RECEIVED