



PF Concept

Gateway Web Service

Gateway order and other web services

Version 2.1.0

October 03, 2023



Index

1. Introduction.....	3
1.1. What you need to know	3
1.2. Support	3
2. Setup and Invoke	4
2.1. Setup client.....	4
2.2. Transmit orders and inventory requests	4
2.3. Receive user authentication	4
2.4. JSON descriptions and samples	4
2.5. Invoke Web Service	4
2.6. Response messages and rejection of an order.....	5
3. Call reference	5
3.1. Header Item Object	5
3.2. GET Inventory	6
3.3. POST Order	6
3.4. Artwork.....	6
3.5. Order Confirmation	7
3.6. Order Status	7
3.7. Advanced Shipping Notification (ASN)	7
APPENDIX	8
POST Order response messages.....	8
Header Item Object	12
GET Inventory request - Inbound	13
GET Inventory request -sample.....	13
GET Inventory response – Outbound	14
GET Inventory response JSON-sample	15
POST Order request – Inbound	16
POST Order request Blank JSON-sample	20
POST Order request Decorated JSON-sample	21
POST Order response – Outbound	23
POST Order response JSON-sample	23
POST Order confirmation message – Outbound.....	24
POST Order confirmation message JSON-sample	26
POST Order status message – Outbound	28
POST Order status message JSON-sample	29
POST ASN message – Outbound	30
POST ASN message JSON-sample	33



1. Introduction

The Web Service is a set of operations accessible by a standardized web protocol. REpresentational State Transfer (REST) is used to exchange structured information in the implementation of Web Services between the network of PF Concept (PFC) and its client. This protocol relies on JSON. Information Set for its message format, and usually relies on other Application Layer protocols, most notably HTTP, for messages negotiation and transmission.

1.1. What you need to know

The present document addresses IT personnel who aim to establish web service integration of their enterprise tools with PF Concept ERP-system. As aforementioned, in order to access the Web Service, REST protocol is used. IT personnel should be familiar with JSON and have a good understanding of Web Services.

We have implemented two approaches of Messages Delivery Services (MDS) in order to exchange information between the serves of PFC and its client:

- **Client calling PF Web Services:** Client can call PF Web Service to use certain services.
Example: Send order, Get inventory, etc.;
- **PFC calling client Web Services:** PFC will make calls to client's web service and push data.
Example: Order Status Notification (OSN), Advance Shipment Notification (ASN), etc.

1.2. Support

If you need support, please contact our PF Concept Gateway Servicedesk by sending an email to api.support@pfconcept.com.



2. Setup and Invoke

This chapter will explain how a client can setup and invoke the PF Concept Gateway Web Service.

2.1. Setup client

What needs to be installed is a machine with working Internet connection and deploy a REST stub generation framework on it.

2.2. Transmit orders and inventory requests

The Web Service definition and scheme files are required for developing applications with the PF Concept Gateway Web Services. In the table below you can find the URI (Uniform Resource Identifier) REST Protocol. Only files in JSON format are supported. For Content type please use application/json. SendOrder should be sent by POST method, while getinventory will use GET method.

Message type	Location
REST	Send Order:
	Live: https://wsa.pfconcept.com/RestGateway/rest/RestGatewayService/order
	Test: https://wsa.pfconcept.com/test/RestGateway/rest/RestGatewayService/order
	Get Inventory:
	Live: https://wsa.pfconcept.com/RestGateway/rest/RestGatewayService/inventory
	Test: https://wsa.pfconcept.com/test/RestGateway/rest/RestGatewayService/inventory

Table 1|Transmission overview

2.3. Receive user authentication

Calling for any operations requires authentication. For the http calls, **basic authentication** will be used. The user authentication (username and password) will be supplied by our Technical Support Team.

Note that separate credentials will be shared for test and production environment. Please make sure that during the testing phase you use the credentials provided for test environment. When moving to production you'll need to adjust to the credentials provided for production environment.

2.4. JSON descriptions and samples

You can find your JSON sample in the appendix of this document for any service you want to consume. Adopt and customize them to fit your needs.

You can also find Swagger documentation here: <https://wsa.pfconcept.com/test/RestGateway/>.

2.5. Invoke Web Service

Once you have customized your request according to your needs, send it to the URI as described in above table and invoke the PF Concept Web Service.



2.6. Response messages and rejection of an order

For every order request sent, you'll get back a response message. This message can either confirm or reject the order.

1. SUCCESS MESSAGE

If the order has been correctly processed, you will get a success response. In the message you'll get status = "success" with response code "202"

2. REJECTION MESSAGE

If the order is rejected, you will get a rejection response. In the message you'll get status = "error" with one on the following response codes, depending on the type of the error:

- a. 401 : Unauthorized -> SenderID/password not correct
- b. 400 : Bad request -> Request JSON is invalid/incomplete/invalid data etc
- c. 415 : Unsupported Media Type -> e.g. If request body is not JSON
- d. 500 : Server error -> Server side error

When an order is not accepted by our Gateway for any of the below describes reasons, you'll get an error response in reply to the order request message you sent.

In the Appendix of the document you can find a complete [overview](#) of all the possible response messages.

3. REJECTION EMAIL

In addition to the above mentioned rejection response, we have another level of validation for possible issues that can't be determined in an earlier stage:

- PO Number already existing
- Artfile is missing on the artwork location

When an order is not accepted by our Gateway for any of the above described reasons, this will trigger an email notification. This message will give an indication of what is causing the error. The message will be sent to the email address that you have supplied to us.

Please note that when you receive this error it means that the order is rejected and it will not be processed until a new attempt is made to send the order.

3. Call reference

3.1. Header Item Object

Header is a common component used in some push/pull transaction held between PF Concept and client servers. The header is used for identification of the message and for general information about the request.

In the Appendix of the document you can find a full [description](#) of the header Item Object section.



3.2. GET Inventory

Inventory returns stock information based on what is provided in the filter in the request sent. Business Unit, Item Number, SKU, Inventory Since Date and Inventory Type needs to be passed in the REST request, which in return shall respond with Business Unit, Item Number, SKU, UOM, Available To Sell Qty, Location, next Qty, next Qty date and Inventory Type.

In the Appendix of the document you can find a full [description](#) of the GET Inventory message.

3.3. POST Order

POST Order is the operation you can call to send an Order message using REST. Upon success, the call will return a confirmation message to the client server.

Please note that it is not possible to combine stock items with items from the WorldSource or PFM segment in one order. A separate order should always be sent for both those types. The segment for each item can be found in our datafeeds.

In the Appendix of the document you can find a full [description](#) of the POST Order message.

3.4. Artwork

For decorated orders the artwork has to be sent to the PF Concept artwork server. Dependent on the purpose a specific type of the artwork needs to be sent. It is possible to send various ArtworkTypes for various purposes:

PROOF	<i>E-proof which shows the position of the logo on the product.</i>
RAW	<i>Logo file. Artist will make required adjustments to the image, if needed.</i>

Besides the above we also have the possibility to deliver production ready artwork. This option can only be used in agreement with PF Concept. Please contact us for more information.

The artwork can be sent as various file types, depending on the chosen ArtworkType:

ArtworkType	Possible file types								
	EPS	AI	PDF	JPG	PNG	GIF	TXT	CDR	DST
PROOF	✓	✓	✓	✓	✓	✓			
RAW	✓	✓	✓	✓	✓		✓	✓	

There are 2 ways to send art files to PF:

- 1 – VIA URL – Placing the artwork on an accessible location on the net and providing the URLs inside the JSON order
- 2 – VIA HTTP CALL – Sending files via an HTTP call and providing file names inside the JSON order.

ART VIA URL

We can read the upload location from the json file and store it on our server. This has our preference. You just have to indicate the complete URL of the image in field "UrlArtFile". The files will be copied and placed on our server.



ART VIA HTTP CALL

Is it possible to send the images through an http call. In order to do this, you can create your own script. Examples will be provided if needed. You can also upload the images through an upload program that we can provide. File names must be unique.

3.5. Order Confirmation

Order confirmation response pushes REST message from PFC database system to client server after receiving a new order. The JSON contains an order confirmation that consist in a recap of the order sent.

Concerning security, we use by default standard base authentication. If you need anything different, please let us know. We let you know if we can handle this.

In the Appendix of the document you can find a full [description](#) of the Order Status message.

3.6. Order Status

Order status response pushes REST message from PFC database system to client server on a periodic basis. The JSON contains status of the sales order record being purchased by the client.

Concerning security, we use by default standard base authentication. If you need anything different, please let us know. We let you know if we can handle this.

The frequency of sending messages is currently several times a day from 8:15 to 16:15 every update made after 16:15 will be send in the first message of the following morning. When the status of the order changes, a new message will be sent to the customer through the gateway.

The status sent can be:

- Stalled
- Processing
- Partial shipped
- Shipped
- Completed
- Cancelled

In the Appendix of the document you can find a full [description](#) of the Order Status message.

3.7. Advanced Shipping Notification (ASN)

Advanced Shipment Notification REST message is pushed from PFC database system to the client server. Delivery Location of the shipment is sent via this REST message which consists of Tracking ID, Carrier, Service, Customer PO Number, Item, Reference, and Delivery Address information.

Concerning security, we use by default standard base authentication. If you need anything different, please let us know. We let you know if we can handle this.

Please note that once this setting is enabled for your Gateway account you will no longer receive these messages via email.

In the Appendix of the document you can find a full [description](#) of the ASN message.



APPENDIX

POST Order response messages

Status	Response code	Message	Description
success	202	Order accepted	Order has been accepted at gateway but the processing has not been completed
error	401	Authentication failed	SenderId/Password not correct or nothing sent on Authorization header
error	400	Expected: Header/SKUs/OrderType/PurchaseOrderNumber/Currency/Shipment	one of these node is missing in the request JSON
error	400	messageId is required.	messageId is missing or blank
error	400	timestamp is required.	timestamp missing or blank
error	400	Invalid time-stamp: <timestamp>. Should be yyyy-mm-ddThh:mm:ss.mmm	timestamp is invalid
error	400	Invalid data type for Header (expected: object)	
error	400	OrderType is required	
error	400	PurchaseOrderNumber is required	
error	400	Currency is required	
error	400	PurchaseOrderTotal is required	
error	400	ProcessingPriority is required	
error	400	skuReferenceID is invalid or empty.	
error	400	Duplicate skuReferenceID: <skuReferenceID>	duplicate <skuReferenceID> is sent for different SKUs
error	400	SKUID is missing or invalid data type. [skuReferenceID: <skuReferenceID>]	node missing in json or empty or wrong data type
error	400	Unitprice is required [skuReferenceID: <skuReferenceID>]	
error	400	Quantity is required [skuReferenceID: <skuReferenceID>]	
error	400	DecorationReferenceIDs entry is invalid or empty [skuReferenceID: <skuReferenceID>]	
error	400	Wrong data type for SKUs, expected: Array	SKUs node should be of type array []
error	400	decorationReferenceID is invalid or empty.	



Status	Response code	Message	Description
error	400	Duplicate decorationReferenceID: <decorationReferenceID>.	
error	400	PMSColors is required [decorationReferenceID: <decorationReferenceID >]	
error	400	ConfigurationID is required [decorationReferenceID: <decorationReferenceID >]	
error	400	NumberOfColors is required [decorationReferenceID: <decorationReferenceID >]	
error	400	ArtWidth is required [decorationReferenceID: <decorationReferenceID >]	
error	400	ArtHeight is required [decorationReferenceID: <decorationReferenceID >]	
error	400	ArtworkReferenceIDs entry is invalid or empty [decorationReferenceID: <decorationReferenceID >]	
error	400	ArtworkReferenceIDs is missing or invalid [decorationReferenceID: <decorationReferenceID >]	
error	400	artworkReferenceID is invalid	
error	400	Duplicate artworkReferenceID: <artworkReferenceID>	
error	400	Either ArtworkFileName or UrlArtFile is expected [artworkReferenceID: <artworkReferenceID>]	
error	400	ArtworkType is required [artworkReferenceID: <artworkReferenceID>]	
error	400	Shipment information missing or incomplete	No "Shipments" node or not sent as array or empty or "shipContact" missing or shipContact > shipAddress missing
error	400	shipmentRefID is invalid or empty	
error	400	Shipment Service is required. [shipmentReferenceID: <shipmentReferenceID>]	
error	400	shipContact Name is required. [shipmentReferenceID: <shipmentReferenceID>]	
error	400	City is required. [shipmentReferenceID: <shipmentReferenceID>]	



Status	Response code	Message	Description
error	400	PostalCode is required. [shipmentReferenceID: <shipmentReferenceID>]	
error	400	Country is required. [shipmentReferenceID: <shipmentReferenceID>]	
error	400	Multiple shipments are not allowed at this moment.	
error	400	Invalid decorationReferenceID: <decorationReferenceID> on SKU data.[skuReferenceID: <skuReferenceID>]	decorationReferenceID is specified on SKU but decoration data is missing
error	400	Decoration data without link to any SKU. [decorationReferenceID: <decorationReferenceID>]	
error	400	Invalid artworkReferenceID: <artworkReferenceID> on decoration data.[decorationReferenceID: <decorationReferenceID>]	
error	400	Art data without link to any Decoration. [artworkReferenceID: <artworkReferenceID>]	
error	500	Header or Order or SKU or Shipment info couldn't be fetched. + <errormessage>	Server side error, contact
error	400	Unknown country in shipAdress: <Country>	
error	400	Invalid Sender ID: <senderId>	Unlikely, this was a double check on later stage
error	500	Customer record not found <senderId>	Debtor link not found or debtor data missing in gateway
error	400	Invalid SKU ID: <SKUID>	
error	400	Manufactured item <SKUID> ordered quantity <Quantity> less then minimum order quantity <minOrdQty>	
error	400	Worldsource item <SKUID> ordered quantity <Quantity> less then minimum order quantity (50)	
error	400	Item <SKUID> May not be sold in country <Countrycode> (Country)	
error	400	Combination of items from PFM or WS with other segment(s) is not allowed: <SKUID>(<segment>);<SKUID>(<segment>)	
error	400	Decoration information is missing	if OrderType is 'Decorated' but no decoration data on request
error	400	Artwork information is missing	if OrderType is 'Decorated' but no artwork data on request



Status	Response code	Message	Description
error	400	Invalid Configuration: <ConfigurationID > for <SKUID> dc <DC>.	
error	400	Item's default decoration not Found for item: <SKUID>	For partner setup with default deco
error	400	Invalid Method ID: <methodID>	For those sending method/orientation/location in configuration id
error	400	Invalid Orientation ID: <orientationID>	For those sending method/orientation/location in configuration id
error	400	Invalid Location ID: <locationID>	For those sending method/orientation/location in configuration id



Header Item Object

Node	Data type	Occurrence	Explanation
Header	Container	Required minOccurs: 1 maxOccurs: 1	Common message header that will be on every inbound/outbound message. Specified by following child elements: messageId, timestamp, isTest, senderId, receiverId, originalMessageId, errorCode, errorMessage
Header .messageId	String maxLength:36	Required minOccurs: 1 maxOccurs: 1	Contains unique identification for the REST message. E.g. 123abc-cba31-22331-bcabcc
Header .timestamp	String maxLength: 28	Required minOccurs: 1 maxOccurs: 1	ISO datetime the message was generated. E.g. 2020-01-09 T11:54:46.839Z
Header .isTest	Boolean	Required minOccurs: 1 maxOccurs: 1	The purpose of having this flag is to have a test order in the production. The setting currently doesn't trigger anything, it's for reference only.
Header .senderId	String maxLength: 16	Optional minOccurs: 0 maxOccurs: 1	Identifier code of the sender to identify who is the sender. The senderId will be configured and confirmed by PF.
Header .receiverId	String maxLength: 2	Required minOccurs: 1 maxOccurs: 1	Identifier code for PF Concept. The receiverId for PF Concept is "PF".
Header .originalMessageId	String maxLength: 36	Optional	Identifier code for the message. Not applicable for sending requests. PF Concept will send back the message received confirmation with the MessageId the request was received as originalMessageId.
Header .errorCode	String maxLength: 20	Optional	Only applicable in response to request in case the message cannot be processed.
Header .errorMessage	String maxLength: 100	Optional	Description of the error code.



GET Inventory request - Inbound

Argument	Type	Occurrence	Meaning
Inventory	Container	Required minOccurs: 1 maxOccurs: 1	The inventory request implements a filter mechanism. All fields should be applied together, as if they represent the WHERE clause of a SQL statement. Any combination of filter options can be sent by the client and respond the data that applies to those criteria. Filter is specified by following child elements: businessUnit, itemNumber, SKU, InventorySinceDate, InventoryType
BusinessUnit	String	Optional minOccurs: 0 maxOccurs: 1	Business unit can be provided to indicate the business division from whom the inventory data needs to be fetched. E.g if the inventory information is to be fetched from PF Concept then 'PF', or if the inventory information is to be fetched from PF UK stock then 'PFUK', or if the inventory information is to be fetched from Leed's then 'LE' (only for US based customers), etc.
ItemNumber	String	Optional minOccurs: 0 maxOccurs: 1	High-level grouping of SKU (Model Code)
SKU	String	Optional minOccurs: 0 maxOccurs: 1	Stock Keeping Unit (SKU) number, a unique identifier for each distinct product and service that can be purchased in business. E.g. '11941100'

GET Inventory request -sample

<https://wsa.pfconcept.com/RestGateway/rest/RestGatewayService/inventory?BusinessUnit=PF&ItemNumber=&SKU=10000200&messageId=b03efbd2-28ab-0291-a114-c58b60a94514×tamp=2022-11-11T12:12:12&receiverId=PF>



GET Inventory response – Outbound

Argument	Type	Occurrence	Meaning
InventoryResponse	Container	Required minOccurs: 1 maxOccurs: 1	Inventory response is the answer to the inventory request which is sent to the requestor. Inventory Response is specified by following child elements: Header (Ref. section 3.1) Inventory Level
InventoryLevel	Container	Required minOccurs: 1 maxOccurs: 1	Inventory Level specified by below child elements: businessUnit, itemNumber, SKU, uom availableToSellQTY, location, nextQTY, nextQTYDate, inventoryType
InventoryLevel .businessUnit	String	Required minOccurs: 1 maxOccurs: 1	Business division from where the inventory data has been sent in response.
InventoryLevel .itemNumber	String	Required minOccurs: 1 maxOccurs: 1	High-level grouping of SKU.
InventoryLevel .sku	String	Required minOccurs: 1 maxOccurs: 1	Stock Keeping Unit (SKU) number, a unique identifier for each distinct product and service that can be purchased in business. E.g. '33SO1012'
InventoryLevel .uom	String	Required minOccurs: 1 maxOccurs: 1	Unit of Measure, e.g. ounces, feet, pcs, etc.
InventoryLevel .availableToSellQTY	Decimal	Required minOccurs: 1 maxOccurs: 1	The number of pieces available to be sold.
InventoryLevel .location	String	Required minOccurs: 1 maxOccurs: 1	The warehouse or site where the pieces reside.
InventoryLevel .nextQTY	Decimal	Optional minOccurs: 0 maxOccurs: 1	The quantity available to sell on the next inbound shipment that has one or more pieces available to sell.
InventoryLevel .nextQTYDate	Date	Optional minOccurs: 0 maxOccurs: 1	The date of the next inbound shipment that has one or more pieces available to sell.
InventoryLevel .inventoryType	String	Optional minOccurs: 0 maxOccurs: 1	Not used



GET Inventory response JSON-sample

```
{
  "InventoryResponse": {
    "Header": {
      "messageId": "f9b8a92c-99e2-139b-a614-c96530ed2097",
      "timestamp": "2023-02-02T11:01:33.920Z",
      "isTest": true,
      "senderId": "PF",
      "receiverId": "XXXXXX",
      "originalMessageId": "b03efbd2-28ab-0291-a114-c58b60a94514",
      "errorCode": "",
      "errorMessage": ""
    },
    "InventoryLevel": [
      {
        "businessUnit": "PF",
        "itemNumber": "100002",
        "sku": "10000200",
        "uom": "EACH",
        "availableToSellQTY": 60356.0,
        "location": "Poland",
        "nextQTY": 2.0,
        "nextQTYDate": "2023-01-30T00:00:00.000Z",
        "InventoryType": ""
      }
    ]
  }
}
```



POST Order request – Inbound

Node	Type	Occurrence	Explanation
	Container	Required minOccurs: 1 maxOccurs: 1	Order container contains following child elements: Header (Ref. Appendix 'Header section'), OrderType, PurchesaOrderNumber, Currency, RequestedInHandsDate, CouponCode, REMaks, PurchaseOrderTotal, ProcessingPriority, Shipments, Skus, Decorations, Artworks.
OrderType	String	Required	One of the values from the list below: BLANK, BLANK SAMPLE, DECORATED
PurchaseOrderNumber	String	Required	Unique order number supplied by sender (client).
Currency	String	Required	Currency code. E.g. EUR
RequestedInHandsDate	Datetime	Optional	N/A.
CouponCode	String	Optional	N/A.
Remarks	String	Optional	N/A.
PurchaseOrderTotal	Decimal	Required	Total order Value. This field must be valued with a number higher than 0. No other checks are made on this field.
ProcessingPriority	String	Required	STANDARD by default.
Shipments	Container	Required minOccurs: 1 maxOccurs: 1	Shipments contains goods delivery information. The child nodes are: Shipment
	Container	Required minOccurs: 1 maxOccurs: 1	The child nodes under shipment are: ShipmentReferenceID, Service, shipContact



Node	Type	Occurrence	Explanation
shipmentReferenceID	Integer	Required	shipmentReferenceID is attribute of Shipment element and expected value is unique numeric value. We suggest to use 1 or 0. Only one shipmentReferenceID is allowed
Service	String	Required	We support STANDARD only*. The code is STD, which needs to be entered here as Shipment service. For Express service please contact us for the possibilities.
shipContact	Container	Required minOccurs: 1 maxOccurs: 1	Contact has delivery address information. The child nodes are: Name, Email, Phone, Fax, Address
shipContact .Name	String	Required	Person's Name or Company name of the delivery address. Optional to add "to attention of" contact after semi colon.
shipContact .Attention	String	Optional	
shipContact .Email	String	Optional	Email address can be put here.
shipContact .Phone	String	Optional	Phone number of delivery address can be put here.
shipAddress	Container	Required minOccurs: 1 maxOccurs: 1	Address contains the delivery Address details. The child nodes are: Address1, Address2, Address3, City, State, PostalCode, Country
shipAddress .Address1	String	Required minOccurs: 1 maxOccurs: 1	Address (street name - house number)
shipAddress .Address2	String	Optional minOccurs: 1 maxOccurs: 1	In case address1 does not cover full address, you can put remaining part of address in address2.
shipAddress .Address3	String	Optional minOccurs: 1 maxOccurs: 1	Rest of the address if any.



Node	Type	Occurrence	Explanation
shipAddress .City	String	Required minOccurs: 1 maxOccurs: 1	City
shipAddress .State	String	Optional minOccurs: 1 maxOccurs: 1	State
shipAddress .PostalCode	String	Required minOccurs: 1 maxOccurs: 1	Postal Code or Zip Code
shipAddress .Country	String	Required	Country Code is iso-2 format. E.g. 'NL' for Netherlands, 'DE' for Germany, etc.
SKUs	Container	Required minOccurs: 1 maxOccurs: 1	SKUs contain the list of SKUs to be purchased by the client. The child nodes are: SKU
	Container	Required minOccurs: 1 maxOccurs: unbound	There can be multiple SKUs in the order. The child nodes are: skuReferenceID, isItemProof, SKUID, UnitPrice, Quantity, DecorationReferenceIDs
SKU .skuReferncelD	Integer	Required	skuReferenceID is attribute of SKU and expected value is unique numeric value like 1, 2, 3
SKU .isItemProof	Boolean	Required	false
SKU .SKUID	String	Required	PF SKU (item) code
SKU .UnitPrice	Decimal	Required	Unit price for the SKU. May not be 0. Optional (not recommended) to build a price check in agreement with PF Concept.
SKU .ParentSKUReferenceID	Decimal	Optional	Used for REPEAT ORDERS. For repeating a decorated order, the original SO number of the order to be repeated should be indicated here. Order type for REPEAT orders should be BLANK and no artwork is needed
SKU .Quantity	Decimal	Required	Order Quantity for the SKU. UOM is the attribute of it and expected value is "EACH".



Node	Type	Occurrence	Explanation
SKU .DecorationReferenceID	Integer	Optional	DecorationReferenceID is used for decorated order only. It reference to Decorations section to which the SKU is linked for decoration method/location/orientation and art files associated with the decorations.
Decorations	Container	Optional minOccurs: 0 maxOccurs: 1	This is used only for Decorated Order, not for Blank order. The decorations methodology which is linked to SKU with DecorationReferenceIDs are defined here.
Decorations .Decoration	Container	Required minOccurs: 1 maxOccurs: Unbound	There can be multiple decorations linked to one SKU. The child nodes are: DecorationReferenceID, allowArtSizeToMax, ConfigurationID, NumberOfColors, PMSColors, ArtworkReferenceIDs, ArtWidth, ArtHeight
Decoration .DecorationReferenceID	Integer	Optional	Used to match the DecorationReferenceID in SKU
Decoration .allowArtSizeToMax	Boolean	Optional	
Decoration .ConfigurationID	String	Required	Configuration ID is the combination of Decoration ImpMethodCode and ImpLocationCode. E.g. for embroidery (code 4) on the front the Configuration ID is 4-front. This information is available for you in our datafeeds.
Decoration .NumberOfColors	Integer	Required	Number of colours to be used for decoration
Decoration .PMSColors	Container	Required	PMS Codes of the colours to be used for decoration
Decoration .ArtworkReferenceIDs	Container	Required	Child node is ArtworkReferenceID



Node	Type	Occurrence	Explanation
Decoration .ArtworkReferenceID	Integer	Required	ArtworkReferenceID is used for decorated order only. It reference to artworks section to which the particular decoration is linked to.
Decoration .ArtWidth	Integer	Required	Width of the artwork
Decoration .ArtHeight	Integer	Required	Height of the artwork
Artworks	Container	Optional minOccurs: 0 maxOccurs: 1	This is used only for Decorated Order, not for Blank order. Art files related information which is linked to decorations are defined here.
Artworks .Artwork	Container	Mandatory minOccurs: 1 maxOccurs: unbounded	There can be multiple art files linked to one artwork. The child nodes are: ArtworkReferenceId, ArtworkFileName, UrlArtFile, ArtworkType, artworkReferenceID which is already defined under Decoration is attribute of Artwork and expected value is unique numeric value like 1, 2, 3.
Artwork .ArtworkReferenceId	String	Required	is already defined under Decoration Expected value is unique numeric value like 1, 2,
Artwork .ArtworkFileName	String	Required	Art file name should be unique name. The naming of the artfile name must be [SenderId]_[unique filename]
Artwork .UrlArtFile	String	Optional	It contains the URL of the artwork
Artwork .ArtworkType	String	Required	Art file Type. Type can be one of 3 types defined: RAW or PROOF

POST Order request Blank JSON-sample

```
{
  "Header":{
    "messageId":"l6roldvvhx9d7eb7exbjkjonrcuhveo3nnx",
    "timestamp":"2022-02-28T13:28:07+01:00",
    "isTest":false,
    "receiverId":"PF"
  },
  "OrderType":"BLANK",
  "PurchaseOrderNumber":"PO-MA-TEST",
}
```



```
"Currency": "EUR",
"RequestedInHandsDate": "2022-11-11",
"CouponCode": "Coupon code",
"Remarks": "GW v4 test remarks",
"PurchaseOrderTotal": 692.0833,
"ProcessingPriority": "STANDARD",

"Shipments": [
  {
    "shipmentReferenceID": "0",
    "Service": "STD",
    "shipContact": {
      "Name": "Contact",
      "Attention": "Att-name",
      "Email": "email@address.com",
      "Phone": "06-12345678",
      "shipAddress": {
        "Address1": "Torenwacht 21",
        "Address3": "",
        "City": "Leiderdorp",
        "State": "Noord-Holland",
        "PostalCode": "2353 DB",
        "Country": "NL"
      }
    }
  }
],
"SKUs": [
  {
    "skuReferenceID": "0",
    "isItemProof": false,
    "SKUID": "10000200",
    "UnitPrice": 0.87,
    "Quantity": 300
  },
  {
    "skuReferenceID": "1",
    "isItemProof": false,
    "SKUID": "10000201",
    "UnitPrice": 0.87,
    "Quantity": 250
  }
]
}
```

POST Order request Decorated JSON-sample

```
{
  "Header": {
    "messageId": "l6roldvvbvx9d7eb7exbjkjonrcuhveo3nnx",
    "timestamp": "2022-02-28T13:28:07+01:00",
    "isTest": false,
    "receiverId": "PF"
  },
  "OrderType": "DECORATED",
  "PurchaseOrderNumber": "PO-MA-TEST",
  "Currency": "EUR",
  "RequestedInHandsDate": "2022-11-11",
  "CouponCode": "Coupon code",
  "Remarks": "GW v4 test remarks",
  "PurchaseOrderTotal": 692.0833,
}
```



```
"ProcessingPriority":"STANDARD",

"Shipments":[
  {
    "shipmentReferenceID":"0",
    "Service":"STD",
    "shipContact":{
      "Name":"Contact",
      "Attention":"Att-name",
      "Email":"email@address.com",
      "Phone":"06-12345678",
      "shipAddress":{
        "Address1":"Torenwacht 21",
        "Address3":"",
        "City":"Leiderdorp",
        "State":"Noord-Holland",
        "PostalCode":"2353 DB",
        "Country":"NL"
      }
    }
  }
],
"SKUs":[
  {
    "skuReferenceID":"0",
    "isItemProof":false,
    "SKUID":"10000200",
    "UnitPrice":0.87,
    "Quantity":300,
    "DecorationReferenceIDs":[
      "0",
      "1"
    ]
  },
  {
    "skuReferenceID":"1",
    "isItemProof":false,
    "SKUID":"10000201",
    "UnitPrice":0.87,
    "Quantity":250,
    "DecorationReferenceIDs":[
      "1"
    ]
  }
],
"Decorations":[
  {
    "decorationReferenceID":"0",
    "allowArtSizeToMax":true,
    "ConfigurationID":"1-front",
    "NumberOfColors":2,
    "PMSColors":[
      "376 C",
      "1235 C"
    ],
    "ArtworkReferenceIDs":[
      "0",
      "1"
    ],
    "ArtWidth":30,
    "ArtHeight":70
  },
  {

```



```
"decorationReferenceID":"1",
"allowArtSizeToMax":false,
"ConfigurationID":"1-front",
"NumberOfColors":2,
"PMSColors":[
  "376 C",
  "1235 C"
],
"ArtworkReferenceIDs":[
  "1"
],
"ArtWidth":30,
"ArtHeight":70
}
],
"Artworks":[
  {
    "artworkReferenceID":"0",
    "ArtworkFileName": "",
    "UrlArtFile": "https://www.artworkserverxxx.com/Visual1.pdf",
    "ArtworkType": "PROOF"
  },
  {
    "artworkReferenceID":"1",
    "ArtworkFileName": "",
    "UrlArtFile": "https://www.artworkserverxxx.com/Artwork1.pdf ",
    "ArtworkType": "RAW"
  }
]
}
```

POST Order response – Outbound

Node	Type	Occurrence	Explanation
	Container	Required minOccurs: 1 maxOccurs: 1	Order container contains following child elements: yourMessageId, Status, message
yourMessageId	String	Required minOccurs: 1 maxOccurs: 1	ID of the Order sent
Status	String	Required minOccurs: 1 maxOccurs: 1	Confirmation of Order REST message after receiving it successfully.
message	String	Required minOccurs: 1 maxOccurs: 1	Additional message to explain reason for Status

POST Order response JSON-sample

```
{
  "yourMessageId": "16roldvvhbx9d7eb7exjbkjonrcuhveo3nnx",
  "status": "success",
  "message": "Order accepted."
}
```



POST Order confirmation message – Outbound

Node	Type	Occurrence	Explanation
	Container	Required minOccurs: 1 maxOccurs: 1	Order confirmation container contains the following child elements: messageId, timestamp, receiverID, orderNumber, customerNumber, customerName, date, reference, currency, subTotal, totalAmount, payment, lines, delivery, vat.
messageId	String	Required minOccurs: 1 maxOccurs: 1	ID of the message
timestamp	String	Required minOccurs: 1 maxOccurs: 1	ISO datetime the message was generated. E.g. 2020-01-09 T11:54:46.839Z
receiverID	String	Required minOccurs: 1 maxOccurs: 1	Identifier code of the receiver.
orderNumber	Integer	Required minOccurs: 1 maxOccurs: 1	PF Sales Order number
customerNumber	Integer	Required minOccurs: 1 maxOccurs: 1	PF code of the customer
customerName	String	Required minOccurs: 1 maxOccurs: 1	Name of the customer
date	String	Required minOccurs: 1 maxOccurs: 1	Creation date of the order
reference	String	Required minOccurs: 1 maxOccurs: 1	PO Number of the order (purchaseOrderNumber)
currency	String	Required minOccurs: 1 maxOccurs: 1	Currency
subTotal	Decimal	Required minOccurs: 1 maxOccurs: 1	Total amount of the order without VAT.
totalAmount	Decimal	Required minOccurs: 1 maxOccurs: 1	Total amount of the order including VAT.
payment	String	Required minOccurs: 1 maxOccurs: 1	Payment term



Node	Type	Occurrence	Explanation
lines	Container	Required minOccurs: 1 maxOccurs: 1	Lines container.
	Container	Required minOccurs: 1 maxOccurs: unbound	There can be multiple lines in the order. Child nodes are: item, description, quantity, price, discount, lineTotal.
.item	String	Required minOccurs: 1 maxOccurs: 1	Sku of the item ordered
.description	String	Required minOccurs: 1 maxOccurs: 1	Description of the item ordered
.quantity	Integer	Required minOccurs: 1 maxOccurs: 1	Item quantity ordered
.price	Decimal	Required minOccurs: 1 maxOccurs: 1	Unit price of the item ordered
.discount	Decimal	Required minOccurs: 1 maxOccurs: 1	Discount applied on the item price
.lineTotal	Decimal	Required minOccurs: 1 maxOccurs: 1	Total amount per line
Delivery	Container	Required minOccurs: 1 maxOccurs: 1	Delivery details. Child nodes are: order, pickingTicket, address, city, zipcode, name, attn, cost.
delivery.order	Integer	Required minOccurs: 1 maxOccurs: 1	PF Order Number
delivery.pickingTicket	Integer	Required minOccurs: 1 maxOccurs: 1	Picking Ticket Number
delivery.address	String	Required minOccurs: 1 maxOccurs: 1	Delivery address
Delivery.city	String	Required minOccurs: 1 maxOccurs: 1	Delivery city
delivery.zipcode	String	Required minOccurs: 1 maxOccurs: 1	Delivery zipcode



Node	Type	Occurrence	Explanation
delivery.name	String	Required minOccurs: 1 maxOccurs: 1	Delivery name
delivery.attn	String	Required minOccurs: 1 maxOccurs: 1	Delivery contact name
delivery.cost	Decimal	Required minOccurs: 1 maxOccurs: 1	Delivery cost
Vat	Container	Required minOccurs: 1 maxOccurs: 1	VAT applied
	Container	Required minOccurs: 1 maxOccurs: unbound	There can be multiple VAT in the order. Child nodes are: text, base, percentage, total.
.text	String	Required minOccurs: 1 maxOccurs: 1	VAT description
.base	Integer	Required minOccurs: 1 maxOccurs: 1	VAT base
.percentage	Integer	Required minOccurs: 1 maxOccurs: 1	VAT percentage
.total	Integer	Required minOccurs: 1 maxOccurs: 1	VAT total amount

POST Order confirmation message **JSON-sample**

```
{
  "messageId": "a8aa56e5-a62f-23a3-af14-7bbb1c064d1e",
  "timestamp": "2023-07-07T12:19:32.699Z",
  "receiverId": "CUSTOMER_ID",
  "orderNumber": 1234567,
  "customerNumber": 3009999,
  "customerName": "CUSTOMER NAME",
  "date": "30/06/23",
  "reference": "TEST_BLANK",
  "currency": "EUR",
  "subTotal": 13.08,
  "totalAmount": 15.70,
  "payment": "Payment Terms",
  "lines": [
    {
      "item": "12065790",
```



```
"description": "Item Description 1",
"quantity": 1,
"price": 3.14,
"discount": 2.00,
"lineTotal": 3.08
},
{
  "item": "",
  "description": "Delivery charges",
  "quantity": 0,
  "price": 0,
  "discount": 0,
  "lineTotal": 10.00
}
],
"delivery": {
  "order": 1234567,
  "pickingTicket": 0,
  "address": "Sunset Boulevard 99",
  "city": "Amsterdam",
  "zipcode": "XXXXXX",
  "name": "John Doe",
  "reference": "XXXXXXXXXX",
  "attn": "",
  "cost": 10.00
},
"vat": [
  {
    "text": "VAT",
    "base": 13.08,
    "percentage": 20.00,
    "total": 2.62
  }
]
}
```



POST Order status message – Outbound

Node	Type	Occurrence	Explanation
StatusChangedNotification	Container	Required minOccurs: 1 maxOccurs: 1	OrderStatusResponse specified by following child elements: messageID, TimeStamp, senderID, receiverID, businessUnit, PO Number, statusCode, statusInformation, expectedShipDate
StatusChangedNotification .messageID	String	Required minOccurs: 1 maxOccurs: 1	ID of the message
StatusChangedNotification .TimeStamp	String	Required minOccurs: 1 maxOccurs: 1	ISO datetime the message was generated. E.g. 2020-01-09 T11:54:46.839Z
StatusChangedNotification .senderID	String	Required minOccurs: 1 maxOccurs: 1	ID code of the sender. Always "PF"
StatusChangedNotification .receiverID	String	Required minOccurs: 1 maxOccurs: 1	Identifier code of the receiver.
StatusChangedNotification .businessUnit	String	Required minOccurs: 1 maxOccurs: 1	The sender's ID which means when PFC is sending the order response businessUnit would always be PF.
StatusChangedNotification .pONumber	String	Required minOccurs: 1 maxOccurs: 1	Customer PO Number reference.
StatusChangedNotification .StatusCode	String	Required minOccurs: 1 maxOccurs: 1	Status code of the order shall be sent via this element. The values can be either one of the following: Stalled, Processing, Partial shipped, Shipped, Completed, Cancelled
StatusChangedNotification .statusInformation	String	Optional minOccurs: 0 maxOccurs: 1	Optional details related to status code.
StatusChangedNotification .expectedShipDate	Date	Optional minOccurs: 0 maxOccurs: 1	Expected shipment date of the order shall be sent via this element.



POST Order status message JSON-sample

```
{  
  "StatusChangedNotification": {  
    "messageId": "bce9a456-674a-26bc-a614-4e5968b003a5",  
    "timestamp": "2023-02-01T15:58:02.982Z",  
    "senderId": "PF",  
    "receiverId": "XXXXXXXXXX",  
    "businessUnit": "PF",  
    "poNumber": "PO123123123",  
    "statusCode": "COMPLETED",  
    "statusInformation": "",  
    "expectedShipDate": "2022-06-30T00:00:00.000Z"  
  }  
}
```



POST ASN message – Outbound

Node	Type	Occurrence	Explanation
ShipmentNotification	Container	Required minOccurs: 1 maxOccurs: 1	Shipment Notification contains shipment information about the client's order. The child elements used are: messageID, TimeStamp, senderID, receiverID, isTest, Deliveries
ShipmentNotification .messageID	String	Required minOccurs: 1 maxOccurs: 1	ID of the message
ShipmentNotification .TimeStamp	String	Required minOccurs: 1 maxOccurs: 1	ISO datetime the message was generated. E.g. 2020-01-09 T11:54:46.839Z
ShipmentNotification .senderID	String	Required minOccurs: 1 maxOccurs: 1	ID code of the sender. Always "PF"
ShipmentNotification .receiverID	String	Required minOccurs: 1 maxOccurs: 1	Identifier code of the receiver.
ShipmentNotification .isTest	Boolean	Required minOccurs: 1 maxOccurs: 1	The purpose of having this flag is to have a test order in the production. The setting currently doesn't trigger anything, it's for reference only.
ShipmentNotification .Deliveries	Container	Required minOccurs: 1 maxOccurs: 1	The element is specified by below child element: DeliveryLocation
Deliveries .DeliveryLocation	Container	Required minOccurs: 1 maxOccurs: unbounded	There can be multiple Delivery Location elements in Deliveries. The element is specified by below child elements: BusinessUnit, Carrier, CustomerPONumber, IsOrderComplete, Item, PrimaryTransportationMode, Reference, Service, ShipDate, TrackingID, TrackingURL, Address (Ref. Section 3.3).
DeliveryLocation .BusinessUnit	String	Required minOccurs: 1 maxOccurs: 1	Business Unit. Always PF
DeliveryLocation .Carrier	String	Required minOccurs: 1 maxOccurs: 1	Freight carrier name that delivers the shipments. E.g. FedEx, UPS, etc.



Node	Type	Occurrence	Explanation
DeliveryLocation .CustomerPONumber	String	Required minOccurs: 1 maxOccurs: 1	Customer Reference Number.
DeliveryLocation .IsOrderComplete	Boolean	Optional	If order status is completed then true, else false.
DeliveryLocation .service	String	Required minOccurs: 1 maxOccurs: 1	Type of service
DeliveryLocation .shipDate	String	Required minOccurs: 1 maxOccurs: 1	Date of shipment
DeliveryLocation .primaryTransportationMode	String	Required minOccurs: 1 maxOccurs: 1	The vehicle used for transport
DeliveryLocation .trackAndTrace	Container	Required minOccurs: 1 maxOccurs: unbounded	Contains information tracking. The element is specified by below child elements: trackingId, trackingURL
trackAndTrace .trackingId	String	Required minOccurs: 1 maxOccurs: 1	Tracking identity number of the shipment.
trackAndTrace .TrackingURL	String	Required minOccurs: 1 maxOccurs: 1	URL of the tracking of the shipment identity number of the shipment.
DeliveryLocation .Item	Container	Required minOccurs: 1 maxOccurs: unbounded	Contains item information about the shipment. There can be multiple Item elements in the Delivery Location element. The Item element is specified by below child element: DeliveryLocationItem
Item .DeliveryLocationItem	Container	Required minOccurs: 1 maxOccurs: unbounded	Contains item information about the shipment. There can be multiple DeliveryLocationItem elements in the Item element. The DeliveryLocationItem element is specified by below child elements: BoxID, CountryOfOrigin, HTSCode, MIDCode, Quantity, SKU, UOM
DeliveryLocationItem .HTSCode	String	Required minOccurs: 1 maxOccurs: 1	Tariff code.



Node	Type	Occurrence	Explanation
DeliveryLocationItem .CountryOfOrigin	String	Required minOccurs: 1 maxOccurs: 1	Country of origin of the shipment.
DeliveryLocationItem .SKU	String	Required minOccurs: 1 maxOccurs: 1	Stock Keeping Unit (SKU) number, a unique identifier for each distinct product and service that can be purchased in business. E.g. '33SO1012'
DeliveryLocationItem .Quantity	String	Required minOccurs: 1 maxOccurs: 1	Quantity order for item being purchased.
DeliveryLocationItem .UOM	String	Required minOccurs: 1 maxOccurs: 1	Unit of Measurement, e.g. pcs, each, etc.
DeliveryLocationItem .BoxID	String	Required minOccurs: 1 maxOccurs: 1	ID of the carton
DeliveryLocationItem .MiDCode	String	Optional minOccurs: 0 maxOccurs: 1	MID code.
DeliveryLocation .Reference	Container	Optional	Reference is an optional container that may occur only once in the delivery location container. Reference is specified by below child elements: ShipmentId, PickTicket
Reference .ShipmentId	String	Required minOccurs: 1 maxOccurs: 1	Contains the shipment ID stored in PFC's database system.
Reference .PickTicket	String	Required minOccurs: 1 maxOccurs: 1	Contains the picking ticket number of the shipment stored in PFC's database system.
DeliveryLocation .Address	Container	Required minOccurs: 1 maxOccurs: 1	Address contains the delivery Address details. The child nodes are: Address1, Address2, Address3, City, Country, PostalCode, State
Address .Address1	String	Required minOccurs: 1 maxOccurs: 1	Address (street name - house number)



Node	Type	Occurrence	Explanation
Address .Address2	String	Optional minOccurs: 1 maxOccurs: 1	In case address1 does not cover full address, you can put remaining part of address in address2.
Address .Address3	String	Optional minOccurs: 1 maxOccurs: 1	Rest of the address if any
Address .City	String	Required minOccurs: 1 maxOccurs: 1	City
Address .State	String	Optional minOccurs: 1 maxOccurs: 1	State.
Address .PostalCode	String	Required minOccurs: 1 maxOccurs: 1	Postal Code or Zip Code
Address .Country	String	Optional minOccurs: 1 maxOccurs: 1	Country Code is iso-2 format. E.g. 'NL' for Netherlands, 'DE' for Germany, etc.

POST ASN message JSON-sample

```
{
  "ShipmentNotification":{
    "messageId":"eeabf801-2b4b-26ab-a614-7f5818ba22e2",
    "timestamp":"2023-02-01T14:44:02.818Z",
    "senderId":"PF",
    "receiverId":"XXXXXXX",
    "isTest":false,
    "deliveries":[
      {
        "deliveryLocation":{
          "businessUnit":"PF",
          "carrier":"UPS",
          "customerPONumber":"PO123123",
          "isOrderComplete":true,
          "service":"std",
          "shipDate":"2022-07-12T00:00:00.000Z",
          "primaryTransportationMode":"TRUCK",
          "trackAndTrace":[
            {
              "trackingId":"123456",
              "trackingURL":"https://www.forwarder.com/options"
            }
          ],
          "item":[
            {
```



```
"HTSCode":"841459959000000000000000",
"countryOfOrigin":"CN",
"SKU":"10070406",
"quantity":10,
"UOM":"EACH",
"boxId": "",
"MIDCode": ""
},
{
  "HTSCode":"961700000000000000000000",
  "countryOfOrigin":"CN",
  "SKU":"10049402",
  "quantity":27,
  "UOM":"EACH",
  "boxId": "",
  "MIDCode": ""
}
],
"reference":{
  "shipmentID":"987654",
  "pickTicket":"789456"
},
"address":{
  "address1":"VIA GARIBALDI 154",
  "address2": "",
  "address3": "",
  "city":"ROME",
  "state": "",
  "postalCode":"00600",
  "country":"IT"
}
}
}
]
}
}
```