

ShippingWCF API - ParcelOne

SOAP Services - Last Update 2019-11-21

Name	Description
IShippingWCF	<p>ParcelOne provides you with a WCF API for implementing an interface to ParcelOne shipping services into your own shop or, respectively, into the shops of your customers.</p> <p>Here you will find a complete documentation of the available API functions with descriptions of request methods, request parameters as well as request and response formats. If you would like to investigate code samples of client implementations in php or C#, you can find them under</p> <ul style="list-style-type: none"> for php (requires php 5.2 or higher): https://sandboxapi.awiwe.solutions/version4/shippingwcfssandbox/doc/ShippingWCF_Client_Sandbox_Example_php.zip for C# (requires Visual Studio 10 or higher): https://sandboxapi.awiwe.solutions/version4/shippingwcfssandbox/doc/ShippingWCF_Client_Sandbox_Example_CSharp.zip <p>As a communication platform we use WCF (Windows Communication Foundation).</p> <p>Authentification is provided by Basic Authentification secured over SSL. Further authentification is done via an API-Key to authenticate the provider of the client software accessing the API. Please, contact us if you would like to receive an API-Key and Sandbox Login data for your client development.</p>

Types

Name	Namespace	Description
Address	http://schemas.datacontract.org/2004/07/ShippingWCF	Address Data.
Amount	http://schemas.datacontract.org/2004/07/ShippingWCF	Amounts Information.
ArrayOfAmount	http://schemas.datacontract.org/2004/07/ShippingWCF	
ArrayOfCEP	http://schemas.datacontract.org/2004/07/ShippingWCF	
ArrayOfCEPSpecial	http://schemas.datacontract.org/2004/07/ShippingWCF	
ArrayOfCharges	http://schemas.datacontract.org/2004/07/ShippingWCF	
ArrayOfChargesPackageResult	http://schemas.datacontract.org/2004/07/ShippingWCF	
ArrayOfChargesResult	http://schemas.datacontract.org/2004/07/ShippingWCF	
ArrayOfClosedResult	http://schemas.datacontract.org/2004/07/ShippingWCF	
ArrayOfConsigner	http://schemas.datacontract.org/2004/07/ShippingWCF	
ArrayOfCustomer	http://schemas.datacontract.org/2004/07/ShippingWCF	
ArrayOfError	http://schemas.datacontract.org/2004/07/ShippingWCF	
ArrayOfIdentifyShipment	http://schemas.datacontract.org/2004/07/ShippingWCF	
ArrayOfIntDocContents	http://schemas.datacontract.org/2004/07/ShippingWCF	
ArrayOfIntDocData	http://schemas.datacontract.org/2004/07/ShippingWCF	
ArrayOfPrintDocumentsResult	http://schemas.datacontract.org/2004/07/ShippingWCF	
ArrayOfPrintRequest	http://schemas.datacontract.org/2004/07/ShippingWCF	
ArrayOfProduct	http://schemas.datacontract.org/2004/07/ShippingWCF	
ArrayOfProfile	http://schemas.datacontract.org/2004/07/ShippingWCF	
ArrayOfService	http://schemas.datacontract.org/2004/07/ShippingWCF	
ArrayOfShipment	http://schemas.datacontract.org/2004/07/ShippingWCF	
ArrayOfShipmentActionResult	http://schemas.datacontract.org/2004/07/ShippingWCF	
ArrayOfShipmentDocumentsResult	http://schemas.datacontract.org/2004/07/ShippingWCF	
ArrayOfShipmentPackage	http://schemas.datacontract.org/2004/07/ShippingWCF	
ArrayOfShipmentPackageResult	http://schemas.datacontract.org/2004/07/ShippingWCF	
ArrayOfShipmentResult	http://schemas.datacontract.org/2004/07/ShippingWCF	
ArrayOfShipmentService	http://schemas.datacontract.org/2004/07/ShippingWCF	
ArrayOfShipmentStatusResult	http://schemas.datacontract.org/2004/07/ShippingWCF	
ArrayOfShipmentTrackingResult	http://schemas.datacontract.org/2004/07/ShippingWCF	
ArrayOfTrackingResult	http://schemas.datacontract.org/2004/07/ShippingWCF	
CEP	http://schemas.datacontract.org/2004/07/ShippingWCF	Carrier Information.
CEPSpecial	http://schemas.datacontract.org/2004/07/ShippingWCF	CEPSpecials Information.
Charges	http://schemas.datacontract.org/2004/07/ShippingWCF	Charges Request Data.
ChargesPackageResult	http://schemas.datacontract.org/2004/07/ShippingWCF	Charges Parcel Results Data.
ChargesResult	http://schemas.datacontract.org/2004/07/ShippingWCF	Charges Result Data.
ClosedResult	http://schemas.datacontract.org/2004/07/ShippingWCF	Closed Shipments Result Data.

Consigner	http://schemas.datacontract.org/2004/07/ShippingWCF	Consigner Information.
Contact	http://schemas.datacontract.org/2004/07/ShippingWCF	Contact Request Data.
Customer	http://schemas.datacontract.org/2004/07/ShippingWCF	Customer Info.
Dimensions	http://schemas.datacontract.org/2004/07/ShippingWCF	Dimensions Information.
Error	http://schemas.datacontract.org/2004/07/ShippingWCF	Error Identification.
Format	http://schemas.datacontract.org/2004/07/ShippingWCF	Format Information.
IdentifyPackage	http://schemas.datacontract.org/2004/07/ShippingWCF	Parcel Identification.
IdentifyShipment	http://schemas.datacontract.org/2004/07/ShippingWCF	Shipment Identification.
IntDoc	http://schemas.datacontract.org/2004/07/ShippingWCF	International Documents Request Data.
IntDocContents	http://schemas.datacontract.org/2004/07/ShippingWCF	International Documents Contents Request Data.
IntDocData	http://schemas.datacontract.org/2004/07/ShippingWCF	International Documents Request Data.
Measurement	http://schemas.datacontract.org/2004/07/ShippingWCF	Measurements Information.
PrintDocumentsResult	http://schemas.datacontract.org/2004/07/ShippingWCF	Shipment Documents Print Result Data.
PrintRequest	http://schemas.datacontract.org/2004/07/ShippingWCF	Print Request Data.
Product	http://schemas.datacontract.org/2004/07/ShippingWCF	Product Information.
Profile	http://schemas.datacontract.org/2004/07/ShippingWCF	Profile for a mandator.
Service	http://schemas.datacontract.org/2004/07/ShippingWCF	Services Information.
ShipFrom	http://schemas.datacontract.org/2004/07/ShippingWCF	ShipFrom / Consigner Data.
Shipment	http://schemas.datacontract.org/2004/07/ShippingWCF	Shipment Request Data.
ShipmentActionResult	http://schemas.datacontract.org/2004/07/ShippingWCF	Shipment Action Results.
ShipmentDocumentsResult	http://schemas.datacontract.org/2004/07/ShippingWCF	Shipment Documents Results Data.
ShipmentPackage	http://schemas.datacontract.org/2004/07/ShippingWCF	Parcel Information.
ShipmentPackageResult	http://schemas.datacontract.org/2004/07/ShippingWCF	Shipment Parcel Results Data.
ShipmentResult	http://schemas.datacontract.org/2004/07/ShippingWCF	Shipment Results Data.
ShipmentService	http://schemas.datacontract.org/2004/07/ShippingWCF	Services Information.
ShipmentStatusResult	http://schemas.datacontract.org/2004/07/ShippingWCF	Shipment Status Results.
ShipmentTrackingResult	http://schemas.datacontract.org/2004/07/ShippingWCF	Shipment Trackings Results.
ShipTo	http://schemas.datacontract.org/2004/07/ShippingWCF	Shipto / Receiver Data.
TrackingResult	http://schemas.datacontract.org/2004/07/ShippingWCF	Shipment Tracking Result Data.
ArrayOfstring	http://schemas.microsoft.com/2003/10/Serialization/Arrays	

SOAP Services

IShippingWCF

ParcelOne provides you with a WCF API for implementing an interface to ParcelOne shipping services into your own shop or, respectively, into the shops of your customers.

Here you will find a complete documentation of the available API functions with descriptions of request methods, request parameters as well as request and response formats. If you would like to investigate code samples of client implementations in php or C#, you can find them under

- for php (requires php 5.2 or higher):
https://sandboxapi.awiwe.solutions/version4/shippingwcfssandbox/doc/ShippingWCF_Client_Sandbox_Example_php.zip
- for C# (requires Visual Studio 10 or higher):
https://sandboxapi.awiwe.solutions/version4/shippingwcfssandbox/doc/ShippingWCF_Client_Sandbox_Example_CSharp.zip

As a communication platform we use WCF (Windows Communication Foundation).

Authentication is provided by Basic Authentication secured over SSL. Further authentication is done via an API-Key to authenticate the provider of the client software accessing the API. Please, contact us if you would like to receive an API-Key and Sandbox Login data for your client development.

Operations

Name	Description
closeShipments (ClosedStamp, Mandator, Consigner, CEP, Format , Shipments)	Close Forward Shipments. (only marked as closed and returning a report. For DHL it does NOT perform the actual DHL closing, this will be done automatically by IntraShip once a day.)
getCEPs (Mandator, level, Countries)	Get available Carriers for a mandator, optionally filtered by countries list.
getCharges (ChargesData)	Get Shipment Charges.
getClosableShipments (Mandator, Consigner, CEP)	get Closable Forward Shipments. May be filtered by Mandantor, Consigner and Carrier.
getClosedStamps (Mandator, Consigner, CEP, StartDate, EndDate)	get ClosedStamps to have a list of all finished closings. May be filtered by Mandantor, Consigner and Carrier as well as StartDate and EndDate.
getCustomers ()	Get the availabe Customer's Information and status.
getDateTime ()	Get Server DateTime to compare with Client DateTime.

getProducts(Mandator, CEP, level, Countries)	Get available Products for a mandator and Carrier, optionally filtered by countries list.
getProfiles(level, Countries)	Get all available carriers for all Mandators, optionally filtered by countries list.
getServices(Mandator, CEP, Product, Countries)	Get available Services for a mandator,Carrier,product, optionally filtered by countries list.
getShipmentStatus(ShippingData)	Get Shipment Status.
getTrackings(ShippingData)	Get Shipment Trackings.
printDocuments(PrintRequests)	Print Shipment Documents, including DHL ExportDocuments. PrintInternationalDocuments function will return the CN23 also for DHL, if preferred.
printInternationalDocuments(PrintRequests)	Print International Shipment Documents, CN23 (2 documents per page) or CN22 (one document).
printLabel(PrintRequests)	Print Shipment Label.
registerIntDoc(InternationalDoc)	Register/Add International Document Data for Shipment.
registerShipments(ShippingData)	Register Forward and Return Shipments.
voidShipments(ShippingData)	Void / Cancel Shipments.

Remarks

In order to be able to test your code without interfering the production environment, we provide a SandBox-Mode as well as a Production-Mode. You can reach the Sandbox- or Test-Mode under ...

- **Service:** <https://sandboxapi.awiwe.solutions/version4/shippingwcfssandbox/ShippingWCF.svc>
- **WSDL:** <https://sandboxapi.awiwe.solutions/version4/shippingwcfssandbox/ShippingWCF.svc?wsdl>

You can reach the Production-Mode under ...

- **Service:** <https://productionapi.awiwe.solutions/version4/shippingwcf/ShippingWCF.svc>
- **WSDL:** <https://productionapi.awiwe.solutions/version4/shippingwcf/ShippingWCF.svc?wsdl>

Please, use the Sandbox mode to test your client application thoroughly before switching to Production mode. In Sandbox mode all data will be saved to a separate database on our server.

IShippingWCF\closeShipments(ClosedStamp, Mandator, Consigner, CEP, Format, Shipments)

Close Forward Shipments. (only marked as closed and returning a report. For DHL it does NOT perform the actual DHL closing, this will be done automatically by IntraShip once a day.)

Parameters

<i>ClosedStamp</i>	Type: string
<i>Mandator</i>	Type: string
<i>Consigner</i>	Type: string
<i>CEP</i>	Type: string
<i>Format</i>	Type: Format
<i>Shipments</i>	Type: ArrayOfIdentifyShipment

Return

Type: [ArrayOfClosedResult](#)

Remarks

You have several possibilities to influence the selection of shipments to be closed:

1. **ClosedStamp:** Provide the ClosedStamp to reprint a Closing Report. No unclosed shipments are touched. (See also function `getClosedStamps()`.)
2. **Mandator:** Only unclosed shipments for this Mandator are closed.
3. **Consigner:** Only unclosed shipments for this Consigner are closed.
4. **CEP:** Only unclosed shipments for this Carrier are closed.
5. **Format:** Format contains information on how to return the data. Only Format.Type is needed, possible values are PDF or CSV.
6. **Shipments:** You can provide a list of Shipments. Only those in your list will be closed.

The parameters Mandator, Consigner and CEP can be mixed and will be considered in an AND-relation.

Return value: Array of ClosedResult containing CEP,ClosedStamp and printDocumentsResult, will contain a separate printDocumentsResult for each Carrier. The closable shipment data may be returned as a PDF file as Base64-encoded string or as a CSV file as Base64-encoded string. The fields in CSV:
ShipmentID;ShipmentRef;TrackingID;Mandator;Consigner;CEP;Product;Packages;Weight;Name1;Name2;Name3;Country;Zip;City;Street;Houseno

Example

Here is how you use this:

```
ClosedResult[] closeShipments(string ClosedStamp, string Mandator, string Consigner, string CEP, Format Format, identifyShipment[] Shipments);
```

Faults

- No exceptions, just the returned data may be empty.

IShippingWCF\getCEPs(Mandator, level, Countries)

Get available Carriers for a mandator, optionally filtered by countries list.

Parameters

Mandator Type: string

level

 Type: int

Countries

 Type: [ArrayOfstring](#)

Return

Type: [ArrayOfCEP](#)

Remarks

Parameters:

1. **Mandator:** Mandator
2. **level:** 0 = all levels returned; 1 = only 1 level returned (only carriers), 2 = only 2 levels returned (carrier and products info), >=3 = also services per product returned
3. **Countries:** optional String array of Alpha-2-code Country abbreviations (ISO 3166-1) to be checked against available carriers and products. Only carriers and products are returned that are available for ALL countries specified. If empty or null, no Countries check is taking place.

Example

Here is how you use this:

```
CEP[] getCEPs(string Mandator, int level, String[] Countries);
```

Faults

- No exceptions, just the returned data may be empty.

IShippingWCF\getCharges([ChargesData](#))

Get Shipment Charges.

Parameters

ChargesData

 Type: [ArrayOfCharges](#)

Return

Type: [ArrayOfChargesResult](#)

Remarks

Parameter:

1. **ChargesData:** You can provide a list of Shipments to get Shipment Charges info for them.

Return value: Array of ChargesResult containing ShipmentCharges, PackageCharges if available and Totalcharges as well as ShipmentActionResult containing ShipmentID, ShipmentRef and TrackingID for identification of shipments, Success flag (1=successfull, -1=not successfull), and Array of Errors (Errorno, Message and StatusCode) in case not being successfull

Example

Here is how you use this:

```
ChargesResult[] getCharges(Charges[] ChargesData);
```

Faults

- No exceptions, just the returned data may contain errors per shipment.

IShippingWCF\getClosableShipments(Mandator, Consigner, CEP)

get Closable Forward Shipments. May be filtered by Mandantor, Consigner and Carrier.

Parameters

Mandator
Type: string*Consigner*
Type: string*CEP*
Type: string**Return**Type: [ArrayOfprintDocumentsResult](#)**Remarks**

You have several possibilities to influence the selection of shipments to be returned as closable:

1. **Mandator:** Only unclosed shipments for this Mandator are returned.
2. **Consigner:** Only unclosed shipments for this Consigner are returned.
3. **CEP:** Only unclosed shipments for this Carrier are returned.

The parameters Mandator, Consigner and CEP can be mixed and will be considered in an AND-relation.

Return value: Array of printDocumentsResult, will contain a separate printDocumentsResult for each Carrier. The closable shipment data will be returned as a CSV file as Base64-encoded string. The fields in CSV:
ShipmentID;ShipmentRef;TrackingID;Mandator;Consigner;CEP;Product;Packages;Weight;Name1;Name2;Name3;Country;Zip;City;Street;Houseno

Example

Here is how you use this:

```
printDocumentsResult[] getClosableShipments(string Mandator, string Consigner, string CEP);
```

Faults

- No exceptions, just the returned CSV data may be empty.

IShippingWCF\getClosedStamps(Mandator, Consigner, CEP, StartDate, EndDate)

get ClosedStamps to have a list of all finished closings. May be filtered by Mandator, Consigner and Carrier as well as StartDate and EndDate.

Parameters*Mandator*
Type: string*Consigner*
Type: string*CEP*
Type: string*StartDate*
Type: string*EndDate*
Type: string**Return**Type: [ArrayOfstring](#)**Remarks**

You have several possibilities to influence the selection of ClosedStamps to be returned:

1. **Mandator:** Only ClosedStamps for this Mandator are returned.
2. **Consigner:** Only ClosedStamps for this Consigner are returned.
3. **CEP:** Only ClosedStamps for this Carrier are returned.
4. **StartDate:** Only ClosedStamps equal or after this StartDate are returned.
5. **EndDate:** Only ClosedStamps equal or before this EndDate are returned.

All parameters can be mixed and will be considered in an AND-relation.

Return value: Array of strings of ClosedStamps. A ClosedStamp consists of DateTimeStamp followed by the CEP, like e.g. "20150520134557UPS".

Example

Here is how you use this:

```
String[] getClosedStamps(string Mandator, string Consigner, string CEP, string StartDate, string EndDate);
```

Faults

- No exceptions, just the returned string array may be empty.

IShippingWCF\getCustomers()

Get the available Customer's Information and status.**Return**

Type: [ArrayOfCustomer](#)

Remarks

Return value: Array of Customer, containing CustomerID + MandatorID, Default flag, Shortname, StatusID and Status as well as Array of Consigners for this Mandator.

Possible statuses: ID 0 = Customer active, ID 1 = Customer blocked, ID 2 = Customer anulled

Example

Here is how you use this:

```
Customer[] getCustomers();
```

Faults

- No exceptions, just the returned data may be empty.

IShippingWCF\getDateTime()**Get Server DateTime to compare with Client DateTime.****Return**

Type: dateTime

Example

Here is how you use this:

```
DateTime GetDateTime();
```

Faults

- No exceptions.

IShippingWCF\getProducts(Mandator, CEP, level, [Countries](#))**Get available Products for a mandator and Carrier, optionally filtered by countries list.****Parameters***Mandator*

Type: string

CEP

Type: string

level

Type: int

Countries

Type: [ArrayOfstring](#)

Return

Type: [ArrayOfProduct](#)

Remarks

Parameters:

1. **Mandator:** Mandator
2. **CEP:** Carrier
3. **level:** 0 = all levels returned; 1 = only 1 level returned (only products), >=2 = 2 levels returned (products and services info)
4. **Countries:** optional String array of Alpha-2-code Country abbreviations (ISO 3166-1) to be checked against available carriers and products. Only carriers and products are returned that are available for ALL countries specified. If empty or null, no Countries check is taking place.

Example

Here is how you use this:

```
Product[] getProducts(string Mandator, string CEP, int level, String[] Countries);
```

Faults

- No exceptions, just the returned data may be empty.

IShippingWCF\getProfiles(*level*, [Countries](#))

Get all available carriers for all Mandators, optionally filtered by countries list.

Parameters

level

Type: int

Countries

Type: [ArrayOfstring](#)

Return

Type: [ArrayOfProfile](#)

Remarks

Parameters:

1. **level:** 0 = all levels returned; 1 = only 1 level returned (only carriers), 2 = only 2 levels returned (carrier and products info), >=3 = also services per product returned
2. **Countries:** optional String array of Alpha-2-code Country abbreviations (ISO 3166-1) to be checked against available carriers and products. Only carriers and products are returned that are available for ALL countries specified. If empty or null, no Countries check is taking place.

Example

Here is how you use this:

```
Profile[] getProfiles(int level, String[] Countries);
```

Faults

- No exceptions, just the returned data may be empty.

IShippingWCF\getServices(*Mandator*, *Carrier*, *Product*, [Countries](#))

Get available Services for a mandator,Carrier,product, optionally filtered by countries list.

Parameters

Mandator

Type: string

Carrier

Type: string

Product

Type: string

Countries

Type: [ArrayOfstring](#)

Return

Type: [ArrayOfService](#)

Remarks

Parameters:

1. **Mandator:** Mandator
2. **Carrier:** Carrier
3. **Product:** Product
4. **Countries:** optional String array of Alpha-2-code Country abbreviations (ISO 3166-1) to be checked against available carriers and products. Only carriers and products are returned that are available for ALL countries specified. If empty or null, no Countries check is taking place.

Example

Here is how you use this:

```
Service[] getServices(string Mandator, string Carrier, string Product, String[] Countries);
```

Faults

- No exceptions, just the returned data may be empty.

IShippingWCF\getShipmentStatus([ShippingData](#))

Get Shipment Status.

Parameters*ShippingData*Type: [ArrayOfIdentifyShipment](#)**Return**Type: [ArrayOfShipmentStatusResult](#)**Remarks**

Parameter:

- 1. **ShippingData:** You can provide a list of Shipments to get Shipment Status info for them.

Return value: Array of ShipmentStatusResult containing ShipmentStatusCode and ShipmentStatus, LastTrackingNo and LastCEP containing DestinationCarrier data as far as known as well as ShipmentActionResult containing ShipmentID, ShipmentRef and TrackingID for identification of shipments, Success flag (1=successfull, -1=not successfull), and Array of Errors (Errorno, Message and StatusCode) in case not being successfull

Possible returned status values so far: Code 10 = Shipped; Code 90 = Closed, Status will be returned according to client language provided.

Example

Here is how you use this:

```
ShipmentStatusResult[] getShipmentStatus(identifyShipment[] ShippingData);
```

Faults

- No exceptions, just the returned data may contain errors per shipment.

IShippingWCF\getTrackings([ShippingData](#))**Get Shipment Trackings.****Parameters***ShippingData*Type: [ArrayOfIdentifyShipment](#)**Return**Type: [ArrayOfShipmentTrackingResult](#)**Remarks**

Parameter:

- 1. **ShippingData:** You can provide a list of Shipments to get Shipment Tracking info for them.

Return value: Array of ShipmentTrackingsResult containing TrackingCode and TrackingStati as well as ShipmentActionResult containing ShipmentID, ShipmentRef and TrackingID for identification of shipments, Success flag (1=successfull, -1=not successfull), and Array of Errors (Errorno, Message and StatusCode) in case not being successfull

Tracking Data will be returned according to client language provided.

Example

Here is how you use this:

```
ShipmentTrackingResult[] getTrackings(identifyShipment[] ShippingData);
```

Faults

- No exceptions, just the returned data may contain errors per shipment.

IShippingWCF\printDocuments([PrintRequests](#))

Print Shipment Documents, including DHL ExportDocuments. PrintInternationalDocuments function will return the CN23 also for DHL, if preferred.

Parameters*PrintRequests*Type: [ArrayOfPrintRequest](#)**Return**Type: [ArrayOfprintDocumentsResult](#)**Remarks**

Parameter - Array of PrintRequest:

1. **DocType:** So far does not have to be specified, for future use.
2. **Format:** Only Format.Type has to be filled, other values for future use. Available values for Format.Type only "PDF".
3. **IDPackage:** PackageRefField to identify field of PackageReference, possible values: PackageID, PackageRef or PackageTrackingID. PackageRefValue to transfer value of this field. If empty, only Shipment is identified.
4. **IDShipment:** ShipmentRefField to identify field of ShipmentReference, possible values: ShipmentID, ShipmentRef or TrackingID. ShipmentRefValue to transfer value of this field.

Return value - Array of printDocumentResult:

1. **ActionResult:** ShipmentActionResult: Success, Errors, ShipmentID, ShipmentRef and TrackingID for Identification of Shipment
2. **DocType:** Possible values to be returned so far: "HighValueReport", "CODTurnInPage", "DHL ExportDocuments".
3. **Document:** Base64-encoded string of PDF document
4. **Format:** Format.Type = PDF. Size so far A4 for PDF. Other values are irrelevant so far.
5. **PackageID:** to identify Package by PackageID
6. **PackageRef:** to identify Package by PackageRef
7. **PackageTrackingID:** to identify Package by PackageTrackingID

Example

Here is how you use this:

```
printDocumentsResult[] printDocuments(PrintRequest[] PrintRequests);
```

Faults

- No exceptions, just the returned data may contain errors per request.
-

IShippingWCF\printInternationalDocuments([PrintRequests](#))

Print International Shipment Documents, CN23 (2 documents per page) or CN22 (one document).

Parameters

PrintRequests

Type: [ArrayOfPrintRequest](#)

Return

Type: [ArrayOfprintDocumentsResult](#)

Remarks

Parameter - Array of PrintRequest:

1. **DocType:** not relevant so far, will be returned as "CN22" or "CN23" in any case.
2. **Format:** Only Format.Type has to be filled, other values for future use. Available values for Format.Type may be either "GIF" or "PDF". Format.Size may be set to "CN22" to return CN22 instead of CN23. If more than 7 detail lines, CN23 will be returned anyway.
3. **IDPackage:** PackageRefField to identify field of PackageReference, possible values: PackageID, PackageRef or PackageTrackingID. PackageRefValue to transfer value of this field. If empty, only Shipment is identified.
4. **IDShipment:** ShipmentRefField to identify field of ShipmentReference, possible values: ShipmentID, ShipmentRef or TrackingID. ShipmentRefValue to transfer value of this field.

Return value - Array of printDocumentResult:

1. **ActionResult:** ShipmentActionResult: Success, Errors, ShipmentID, ShipmentRef and TrackingID for Identification of Shipment
2. **DocType:** returned as "CN22" or "CN23".
3. **Document:** Base64-encoded string of CN22/CN23, either PDF or GIF
4. **Format:** Format.Type = GIF or PDF. Size so far = A4 for GIF, A4/A6 (CN22) for PDF. Other values are irrelevant so far.
5. **PackageID:** to identify Package by PackageID
6. **PackageRef:** to identify Package by PackageRef
7. **PackageTrackingID:** to identify Package by PackageTrackingID

Example

Here is how you use this:

```
printDocumentsResult[] printInternationalDocuments(PrintRequest[] PrintRequests);
```

Faults

- No exceptions, just the returned data may contain errors per request.
-

IShippingWCF\printLabel([PrintRequests](#))

Print Shipment Label.

Parameters

PrintRequests

Type: [ArrayOfPrintRequest](#)

Return

Type: [ArrayOfprintDocumentsResult](#)**Remarks**

Parameter - Array of PrintRequest:

1. **DocType:** not relevant so far, will be returned as "Label" in any case.
2. **Format:** Only Format.Type has to be filled, other values for future use. Available values for Format.Type may be either "GIF" or "PDF".
3. **IDPackage:** PackageRefField to identify field of PackageReference, possible values: PackageID, PackageRef or PackageTrackingID. PackageRefValue to transfer value of this field. If empty, only Shipment is identified.
4. **IDShipment:** ShipmentRefField to identify field of ShipmentReference, possible values: ShipmentID, ShipmentRef or TrackingID. ShipmentRefValue to transfer value of this field.

Return value - Array of printDocumentResult:

1. **ActionResult:** ShipmentActionResult: Success, Errors, ShipmentID, ShipmentRef and TrackingID for Identification of Shipment
2. **DocType:** returned as "Label".
3. **Document:** Base64-encoded string of label, either PDF or GIF
4. **Format:** Format.Type = GIF or PDF. Size so far = A6 for GIF, A4 for PDF. Other values are irrelevant so far.
5. **PackageID:** to identify Package by PackageID
6. **PackageRef:** to identify Package by PackageRef
7. **PackageTrackingID:** to identify Package by PackageTrackingID

Example

Here is how you use this:

```
printDocumentsResult[] printLabel(PrintRequest[] PrintRequests);
```

Faults

- No exceptions, just the returned data may contain errors per request.
-

IShippingWCF\registerIntDoc([InternationalDoc](#))**Register/Add International Document Data for Shipment.****Parameters***InternationalDoc*Type: [ArrayOfIntDocData](#)**Return**Type: [ArrayOfprintDocumentsResult](#)**Remarks**

Parameter - Array of IntDocData:

1. **IDShipment:** to identify Shipment for adding IntDocData: ShipmentRefField to identify field of ShipmentReference, possible values: ShipmentID, ShipmentRef or TrackingID. ShipmentRefValue to transfer value of this field.
2. **IDPackage:** optionally: PackageRefField to identify field of PackageReference, possible values: PackageID, PackageRef or PackageTrackingID. PackageRefValue to transfer value of this field. If empty, only Shipment is identified.
3. **InternationalDoc:** The International Document Data, see also description of function: registerShipments

Return value - Array of printDocumentResult:

1. **ActionResult:** ShipmentActionResult: Success, Errors, ShipmentID, ShipmentRef and TrackingID for Identification of Shipment
2. **DocType:** returned as "CN22" or "CN23".
3. **Document:** Base64-encoded string of CN22/CN23, either PDF or GIF
4. **Format:** Format.Type = GIF or PDF. Format.Size = CN22 or CN23.
5. **PackageID:** to identify Package by PackageID
6. **PackageRef:** to identify Package by PackageRef
7. **PackageTrackingID:** to identify Package by PackageTrackingID

Example

Here is how you use this:

```
printDocumentsResult[] registerIntDoc(IntDocData[] InternationalDoc);
```

Faults

- No exceptions, just the returned data may be empty.
-

IShippingWCF\registerShipments([ShippingData](#))**Register Forward and Return Shipments.****Parameters***ShippingData*Type: [ArrayOfShipment](#)

Return

Type: [ArrayOfShipmentResult](#)

Remarks

Parameter - Array of Shipment:

1. **CEPID:** required - Carrier specification, possible values so far: UPS, PA1, DHL.
2. **ConsignerID:** required - Consigner ID at ParcelOne , usually "1", if only one consigner.
3. **CostCenter:** optional - only for client reference.
4. **DocumentFormat:** required - only Format.Type = "PDF" needed.
5. **LabelFormat:** required - only Format.Type = either "GIF" or "PDF" needed.
6. **MandatorID:** required - Mandator ID at ParcelOne , usually "1", if only one mandator.
7. **Other1:** optional - only for client reference.
8. **Other2:** optional - only for client reference.
9. **Other3:** optional - only for client reference.
10. **Packages:** required - ShipmentPackage data, see definition below.
11. **PrintDocuments:** required - Flag to indicate, if Documents (for DHL also Export Documents) should be returned with this request (0=no, 1=yes).
12. **PrintLabel:** required - Flag to indicate, if Label should be returned with this request (0=no, 1=yes).
13. **ProductID:** required - Product specification, in accordance with Carrier specification..
14. **ReturnCharges:** optional - to return Charges (0=no (default), 1=yes).
15. **MaxCharges:** optional - to specify maximum amount charge, over maximum will refuse the shipment.
16. **ReturnShipmentIndicator:** required - 0 = Forward Shipment, all values > 0 = Return Shipment. For UPS the following are available: 2-Print and Mail Return Label by UPS; 3-Return Service 1-Attempt; 5-Return Service 3-Attempt; 8-Electronic Return Label by URL; 9-Print Return Label. For DHL and Parcel One so far not available.
17. **Services:** optional - ShipmentService data, see definition below.
18. **ShipToData:** required - ShipTo Data - (= ShipFrom Data with Return Shipments), see definition below.
19. **ShipFromData:** optional - ShipFrom Data - only for special purposes, usually ConsignerID has to be provided. Requires special agreement with ParcelOne .
20. **ShipmentID:** required - ShipmentID, if Shipping successfull, this will be overwritten/assigned.
21. **ShipmentRef:** required - ShipmentRef for identification of shipment by client.
22. **Software:** required - Software specification of client, if possible with version number.

Parameter ShipmentPackage - Package:

1. **IntDocData:** optional/required - see definition below
2. **PackageDimensions:** optional/required - Dimensions: Height, Length, Width, Measurement="cm".
3. **PackageID:** optional - will be overwritten if shipment successfull.
4. **PackageRef:** optional - to identify package by client.
5. **PackageType:** required for UPS - UPS: Package Type = "02" for "Package".
6. **PackageVolume:** optional/required - Measurement: Unit = "ltr", Value = volume in ltr.
7. **PackageWeight:** required - Measurement: Unit = "kg", Value = weight in kg.
8. **Remarks:** optional for Forward Shipments, required for UPS Return Shipments = Contents Description, like: "goods".
9. **Services:** optional - Services per Package, see definition below.

Parameter IntDocData- IntDoc:

1. **Certificate:** 0 = no, 1=yes.
2. **CertificateNo:** optional - Certificate No.
3. **Comments:** optional - will be overwritten if shipment successfull.
4. **ConsignerCustomsID:** optional - to identify package by client.
5. **ContentsDesc:** required - Array of IntDocContents, see Definition below.
6. **Currency:** optional, default=EUR - Currency for values.
7. **Date:** optional, default=today - Date on CN23.
8. **Explanation:** optional - only for client reference.
9. **Invoice:** 0 = no, 1=yes.
10. **InvoiceNo:** optional - Invoice No.
11. **InternationalDocumentFormat:** required - only Format.Type = "PDF" or "GIF" needed. Format.Size may be set to "CN22" to return CN22 instead of CN23.
12. **ItemCategory:** optional - Item Category, possible values:1-Gift, 2-Documents, 3-Commercial Sample, 4-Returned Goods, 5-Other
13. **License:** 0 = no, 1=yes.
14. **LicenseNo:** optional - License No.
15. **OfficeOfOrigin:** optional - Office Of Origin.
16. **Postage:** optional - Postage Amount.
17. **PrintInternationalDocuments:** required - 0=Save Int.Doc.Data only or 1=return also printable CN23.
18. **ShipToRef:** optional - Receiver Reference on CN23.
19. **TotalValue:** optional - Total Value amount.
20. **TotalWeightkg:** optional - Total Weight Amount in kg.
21. **NonDeliveryInstruction:** optional (on some versions of CN23: Values: Treat as abandoned, Return to sender) - not used yet, for future use.
22. **ServiceLevel:** optional (on some versions of CN23: Values: PRI, ...) - not used yet, for future use.
23. **ValidatedForExport:** optional (on some versions of CN23: Values: 0=NO, 1=YES) - not used yet, for future use.

Parameter ContentsDesc - Array of IntDocContents:

1. **Contents:** Contents description.
2. **ItemValue:** Value of Item.
3. **NetWeight:** NetWeight of item.
4. **Origin:** Origin of Item: Alpha-2-code Country abbreviation (ISO 3166-1).
5. **Quantity:** Quantity of Item.
6. **TariffNumber:** TariffNumber of Item.

Parameter Services - Array of ShipmentService:

1. **Parameters:** optional - so far not in use, for future use.
2. **ServiceID:** required - Service ID, e.g. NN for COD, WERT for insurance, SA for Saturday Delivery, etc.
3. **Value:** optional - Currency and Value specification, e.g. for Insurance or COD.

Parameter CEPSpecials - Array of CEPSpecial:

1. **Key:** required - for DHL so far: Key: PRINTONLYIFCODEABLE, Value: 1 - will return label only in case LEITCODE could be found to avoid Surcharge for not codable addresses
2. **Value:** optional - value according to requirements of CEP

Parameter - ShipTo:

1. **BranchID:** optional - Branch or Pickup ID of Carrier. If PUDO-Shipping, then PUDO-Number must be provided here.
2. **CEPCustID:** optional - Carrier Customer ID of Receiver, not in use yet, for future use.
3. **CustomsID:** optional - Customs ID of client.
4. **Name1:** required - First name line.
5. **Name2:** optional - second name line.
6. **Name3:** optional - third name line.
7. **PrivateAddressIndicator:** optional/required - 1=Private Address, 0=B2B-Address.
8. **Reference:** optional - Receiver Reference.

9. **SalesTaxID:** optional - Sales Tax ID.
10. **ShipmentAddress:** required - ShipTo address, see definition below.
11. **ShipmentContact:** required - ShipTo contact, see definition below.

Parameter ShipmentAddress - Address:

1. **City:** required - City.
2. **Country:** required - Alpha-2-code Country abbreviation (ISO 3166-1).
3. **District:** optional - District, often printed under StreetNo and Streetno.
4. **PostalCode:** optional - ZIPCode.
5. **State:** required - State, often printed under PostalCode and City.
6. **Street:** optional/required - Street.
7. **Streetno:** optional/required - Streetno.

Parameter ShipmentContact - Contact:

1. **AttentionName:** optional - Attention name.
2. **BirthDate:** optional - Birth date, not used yet.
3. **Company:** optional - Company.
4. **ContactPerson:** optional - Contact person.
5. **Email:** required - email of receiver.
6. **Fax:** optional - fax.
7. **FirstName:** optional - firstname.
8. **LastName:** optional - lastname.
9. **Mobile:** optional - mobile phone number.
10. **Phone:** optional - phone number.
11. **Salutation:** optional - salutation.

Return value - Array of ShipmentResult:

1. **ActionResult:** ShipmentActionResult: Success, Errors, ShipmentID, ShipmentRef and TrackingID for Identification of Shipment
2. **Charges:** Array of Amount (Currency,Value and Description) - to return price information.
3. **DocumentsAvailable:** Number of documents that have been produced for this shipment.
4. **DocumentsResults:** Array of ShipmentDocumentsResult: Charges, DocType, Base64-encoded string of GIF or PDF document, Format, PackageID for Package Identification, ShipmentID and TrackingID for Shipment Identificaiton, DocumentID for Document Identification (e.g. ParcelOne Retoure Trackno)
5. **InternationalDocumentsAvailable:** Number of International documents that have been produced for this shipment.
6. **InternationalDocumentsNeeded:** Values: -1 = has not been checked, 0 = not needed, 1 = needed. Returned independently of whether Customs Data were provided or not.
7. **LabelsAvailable:** Number of labels that have been produced for this shipment.
8. **LabelURL:** URL to download Label and Documents, for UPS Return Shipments with Download-URL to include in email for receiver.
9. **PackageResults:** Array of ShipmentPackageResult: Charges for Package, DocType, Format, Label, TrackingURL, PackageID and PackageRef for Package Identification, ShipmentID and TrackingID for Shipment Identificaiton

Example

Here is how you use this:

```
ShipmentResult[] registerShipments(Shipment[] ShippingData);
```

Faults

- No exceptions, just the returned data may contain errors per request.

IShippingWCF\voidShipments([ShippingData](#))

Void / Cancel Shipments.

Parameters

ShippingData

Type: [ArrayOfIdentifyShipment](#)

Return

Type: [ArrayOfShipmentActionResult](#)

Remarks

Parameter:

1. **ShippingData:** You can provide a list of Shipments. Only those in your list will be voided/cancelled if possible.

Return value: Array of ShipmentActionResult containing ShipmentID, ShipmentRef and TrackingID for identification of shipments, Success flag (1=successfull, -1=not successfull), and Array of Errors (Errorno, Message and StatusCode) in case not being successfull

Example

Here is how you use this:

```
ShipmentActionResult[] voidShipments(identifyShipment[] ShippingData);
```

Faults

- No exceptions, just the returned data may contain errors per shipment.

Types

Address Type

Address Data.

Members

Member Name	Type	Required	MaxLength	Description
City	string	Yes	40	City.
Country	string	Yes	2	Alpha-2-code Country abbreviation (ISO 3166-1).
District	string	No	40	District or any information to be printed under Postal Code / City.
PostalCode	string	No	15	Postal Code, required for most countries, but there are exceptions.
State	string	No	30	State Information, printed under Postal Code / City and District.
Street	string	No	40	Street with or without Streetno, in most cases required.
Streetno	string	No	15	Streetno, may also be included in Street field.

Xml Representation

```
<Address>
<City>...</City>
<Country>...</Country>
<District>...</District>
<PostalCode>...</PostalCode>
<State>...</State>
<Street>...</Street>
<Streetno>...</Streetno>
</Address>
```

Amount Type

Amounts Information.

Members

Member Name	Type	Required	MaxLength	Description
Currency	string	No	3	3-char-Abbreviation of Currency, e.g. EUR for Euro (Default).
Description	string	No		Description of Amount.
Value	string	Yes	Decimal(10,2)	Actual Amount, Number Format: InvariantInfo.

Xml Representation

```
<Amount>
<Currency>...</Currency>
<Description>...</Description>
<Value>...</Value>
</Amount>
```

ArrayOfAmount Array

Array Element

Element Name	Type
Amount	Amount

Xml Representation

```
<ArrayOfAmount>
<Amount>...</Amount>
<Amount>...</Amount>
<Amount>...</Amount>
...
</ArrayOfAmount>
```

ArrayOfCEP Array

Array Element

Element Name	Type
CEP	CEP

Xml Representation

```
<ArrayOfCEP>
  <CEP>...</CEP>
  <CEP>...</CEP>
  <CEP>...</CEP>
  ...
</ArrayOfCEP>
```

ArrayOfCEPSpecial Array**Array Element**

Element Name	Type
CEPSpecial	CEPSpecial

Xml Representation

```
<ArrayOfCEPSpecial>
  <CEPSpecial>...</CEPSpecial>
  <CEPSpecial>...</CEPSpecial>
  <CEPSpecial>...</CEPSpecial>
  ...
</ArrayOfCEPSpecial>
```

ArrayOfCharges Array**Array Element**

Element Name	Type
Charges	Charges

Xml Representation

```
<ArrayOfCharges>
  <Charges>...</Charges>
  <Charges>...</Charges>
  <Charges>...</Charges>
  ...
</ArrayOfCharges>
```

ArrayOfChargesPackageResult Array**Array Element**

Element Name	Type
ChargesPackageResult	ChargesPackageResult

Xml Representation

```
<ArrayOfChargesPackageResult>
  <ChargesPackageResult>...</ChargesPackageResult>
  <ChargesPackageResult>...</ChargesPackageResult>
  <ChargesPackageResult>...</ChargesPackageResult>
  ...
</ArrayOfChargesPackageResult>
```

ArrayOfChargesResult Array**Array Element**

Element Name	Type
ChargesResult	ChargesResult

Xml Representation

```
<ArrayOfChargesResult>
  <ChargesResult>...</ChargesResult>
  <ChargesResult>...</ChargesResult>
```

```
<ChargesResult>...</ChargesResult>
...
</ArrayOfChargesResult>
```

ArrayOfClosedResult Array

Array Element

Element Name	Type
ClosedResult	ClosedResult

Xml Representation

```
<ArrayOfClosedResult>
  <ClosedResult>...</ClosedResult>
  <ClosedResult>...</ClosedResult>
  <ClosedResult>...</ClosedResult>
  ...
</ArrayOfClosedResult>
```

ArrayOfConsigner Array

Array Element

Element Name	Type
Consigner	Consigner

Xml Representation

```
<ArrayOfConsigner>
  <Consigner>...</Consigner>
  <Consigner>...</Consigner>
  <Consigner>...</Consigner>
  ...
</ArrayOfConsigner>
```

ArrayOfCustomer Array

Array Element

Element Name	Type
Customer	Customer

Xml Representation

```
<ArrayOfCustomer>
  <Customer>...</Customer>
  <Customer>...</Customer>
  <Customer>...</Customer>
  ...
</ArrayOfCustomer>
```

ArrayOfError Array

Array Element

Element Name	Type
Error	Error

Xml Representation

```
<ArrayOfError>
  <Error>...</Error>
  <Error>...</Error>
  <Error>...</Error>
  ...
</ArrayOfError>
```

ArrayOfIdentifyShipment Array

Array Element

Element Name	Type
identifyShipment	identifyShipment

Xml Representation

```
<ArrayOfIdentifyShipment>
  <identifyShipment>...</identifyShipment>
  <identifyShipment>...</identifyShipment>
  <identifyShipment>...</identifyShipment>
  ...
</ArrayOfIdentifyShipment>
```

ArrayOfIntDocContents Array

Array Element

Element Name	Type
IntDocContents	IntDocContents

Xml Representation

```
<ArrayOfIntDocContents>
  <IntDocContents>...</IntDocContents>
  <IntDocContents>...</IntDocContents>
  <IntDocContents>...</IntDocContents>
  ...
</ArrayOfIntDocContents>
```

ArrayOfIntDocData Array

Array Element

Element Name	Type
IntDocData	IntDocData

Xml Representation

```
<ArrayOfIntDocData>
  <IntDocData>...</IntDocData>
  <IntDocData>...</IntDocData>
  <IntDocData>...</IntDocData>
  ...
</ArrayOfIntDocData>
```

ArrayOfprintDocumentsResult Array

Array Element

Element Name	Type
printDocumentsResult	printDocumentsResult

Xml Representation

```
<ArrayOfprintDocumentsResult>
  <printDocumentsResult>...</printDocumentsResult>
  <printDocumentsResult>...</printDocumentsResult>
  <printDocumentsResult>...</printDocumentsResult>
  ...
</ArrayOfprintDocumentsResult>
```

ArrayOfPrintRequest Array

Array Element

Element Name	Type
PrintRequest	PrintRequest

Xml Representation

```
<ArrayOfPrintRequest>
  <PrintRequest>...</PrintRequest>
  <PrintRequest>...</PrintRequest>
  <PrintRequest>...</PrintRequest>
  ...
</ArrayOfPrintRequest>
```

ArrayOfProduct Array**Array Element**

Element Name	Type
Product	Product

Xml Representation

```
<ArrayOfProduct>
  <Product>...</Product>
  <Product>...</Product>
  <Product>...</Product>
  ...
</ArrayOfProduct>
```

ArrayOfProfile Array**Array Element**

Element Name	Type
Profile	Profile

Xml Representation

```
<ArrayOfProfile>
  <Profile>...</Profile>
  <Profile>...</Profile>
  <Profile>...</Profile>
  ...
</ArrayOfProfile>
```

ArrayOfService Array**Array Element**

Element Name	Type
Service	Service

Xml Representation

```
<ArrayOfService>
  <Service>...</Service>
  <Service>...</Service>
  <Service>...</Service>
  ...
</ArrayOfService>
```

ArrayOfShipment Array**Array Element**

Element Name	Type
Shipment	Shipment

Xml Representation

```
<ArrayOfShipment>
  <Shipment>...</Shipment>
  <Shipment>...</Shipment>
  <Shipment>...</Shipment>
  ...
</ArrayOfShipment>
```

ArrayOfShipmentActionResult Array

Array Element

Element Name	Type
ShipmentActionResult	ShipmentActionResult

Xml Representation

```
<ArrayOfShipmentActionResult>
  <ShipmentActionResult>...</ShipmentActionResult>
  <ShipmentActionResult>...</ShipmentActionResult>
  <ShipmentActionResult>...</ShipmentActionResult>
  ...
</ArrayOfShipmentActionResult>
```

ArrayOfShipmentDocumentsResult Array

Array Element

Element Name	Type
ShipmentDocumentsResult	ShipmentDocumentsResult

Xml Representation

```
<ArrayOfShipmentDocumentsResult>
  <ShipmentDocumentsResult>...</ShipmentDocumentsResult>
  <ShipmentDocumentsResult>...</ShipmentDocumentsResult>
  <ShipmentDocumentsResult>...</ShipmentDocumentsResult>
  ...
</ArrayOfShipmentDocumentsResult>
```

ArrayOfShipmentPackage Array

Array Element

Element Name	Type
ShipmentPackage	ShipmentPackage

Xml Representation

```
<ArrayOfShipmentPackage>
  <ShipmentPackage>...</ShipmentPackage>
  <ShipmentPackage>...</ShipmentPackage>
  <ShipmentPackage>...</ShipmentPackage>
  ...
</ArrayOfShipmentPackage>
```

ArrayOfShipmentPackageResult Array

Array Element

Element Name	Type
ShipmentPackageResult	ShipmentPackageResult

Xml Representation

```
<ArrayOfShipmentPackageResult>
  <ShipmentPackageResult>...</ShipmentPackageResult>
  <ShipmentPackageResult>...</ShipmentPackageResult>
  <ShipmentPackageResult>...</ShipmentPackageResult>
```

...
 </ArrayOfShipmentPackageResult>

ArrayOfShipmentResult Array

Array Element

Element Name	Type
ShipmentResult	ShipmentResult

Xml Representation

```
<ArrayOfShipmentResult>
  <ShipmentResult>...</ShipmentResult>
  <ShipmentResult>...</ShipmentResult>
  <ShipmentResult>...</ShipmentResult>
  ...
</ArrayOfShipmentResult>
```

ArrayOfShipmentService Array

Array Element

Element Name	Type
ShipmentService	ShipmentService

Xml Representation

```
<ArrayOfShipmentService>
  <ShipmentService>...</ShipmentService>
  <ShipmentService>...</ShipmentService>
  <ShipmentService>...</ShipmentService>
  ...
</ArrayOfShipmentService>
```

ArrayOfShipmentStatusResult Array

Array Element

Element Name	Type
ShipmentStatusResult	ShipmentStatusResult

Xml Representation

```
<ArrayOfShipmentStatusResult>
  <ShipmentStatusResult>...</ShipmentStatusResult>
  <ShipmentStatusResult>...</ShipmentStatusResult>
  <ShipmentStatusResult>...</ShipmentStatusResult>
  ...
</ArrayOfShipmentStatusResult>
```

ArrayOfShipmentTrackingResult Array

Array Element

Element Name	Type
ShipmentTrackingResult	ShipmentTrackingResult

Xml Representation

```
<ArrayOfShipmentTrackingResult>
  <ShipmentTrackingResult>...</ShipmentTrackingResult>
  <ShipmentTrackingResult>...</ShipmentTrackingResult>
  <ShipmentTrackingResult>...</ShipmentTrackingResult>
  ...
</ArrayOfShipmentTrackingResult>
```

ArrayOfTrackingResult Array

Array Element

Element Name	Type
TrackingResult	TrackingResult

Xml Representation

```
<ArrayOfTrackingResult>
  <TrackingResult>...</TrackingResult>
  <TrackingResult>...</TrackingResult>
  <TrackingResult>...</TrackingResult>
  ...
</ArrayOfTrackingResult>
```

CEP Type

Carrier Information.

Members

Member Name	Type	Required	MaxLength	Description
CEPID	string	Yes	3	Carrier abbreviation (UPS, PA1 for Parcel One, ...).
CEPLongname	string	No	30	Carrier Longname.
Default	int	No	0 or 1	Default Carrier information.
Products	ArrayOfProduct	No		Products available for this carrier

Xml Representation

```
<CEP>
  <CEPID>...</CEPID>
  <CEPLongname>...</CEPLongname>
  <Default>...</Default>
  <Products>...</Products>
</CEP>
```

CEPSpecial Type

CEPSpecials Information.

Members

Member Name	Type	Required	MaxLength	Description
Key	string	No	20	for DHL so far: Key: PRINTONLYIFCODEABLE, Value: 1 - will return label only in case LEITCODE could be found to avoid Surcharge for not codable addresses.
Value	string	No	20	the needed value as string representation

Xml Representation

```
<CEPSpecial>
  <Key>...</Key>
  <Value>...</Value>
</CEPSpecial>
```

Charges Type

Charges Request Data.

Members

Member Name	Type	Required	MaxLength	Description
CEPID	string	Yes	3	Carrier abbreviation (UPS, PA1 for Parcel One, ...).
ConsignerID	string	Yes	20	Consigner Abbreviation.
IDShipment	identifyShipment	Yes		to identify Shipments to process.
MandatorID	string	Yes	20	Mandator Abbreviation.

Packages	ArrayOfShipmentPackage	Yes		Array of Packages to calculate.
PrivateAddressIndicator	int	No		Private Address = 1, B2B = 0
ProductID	string	Yes	20	Product Abbreviation
ReturnShipmentIndicator	int	No		Return Shipment = 1; default = 0 = Forward Shipment
Services	ArrayOfShipmentService	No		Array of Shipment Services of this shipment.
ShipToAddress	Address	Yes		ShipTo Data.

Xml Representation

```
<Charges>
  <CEPID>...</CEPID>
  <ConsignerID>...</ConsignerID>
  <IDShipment>...</IDShipment>
  <MandatorID>...</MandatorID>
  <Packages>...</Packages>
  <PrivateAddressIndicator>...</PrivateAddressIndicator>
  <ProductID>...</ProductID>
  <ReturnShipmentIndicator>...</ReturnShipmentIndicator>
  <Services>...</Services>
  <ShipToAddress>...</ShipToAddress>
</Charges>
```

ChargesPackageResult Type

Charges Parcel Results Data.

Members

Member Name	Type	Required	MaxLength	Description
Charges	ArrayOfAmount	No		Charges for each package in shipments.
PackageID	string	No	20	Package Identification.
PackageRef	string	No	20	Parcel Reference Field of Client.
ShipmentID	string	No	20	Shipment ID.
TrackingID	string	No	20	Tracking number

Xml Representation

```
<ChargesPackageResult>
  <Charges>...</Charges>
  <PackageID>...</PackageID>
  <PackageRef>...</PackageRef>
  <ShipmentID>...</ShipmentID>
  <TrackingID>...</TrackingID>
</ChargesPackageResult>
```

ChargesResult Type

Charges Result Data.

Members

Member Name	Type	Required	MaxLength	Description
ActionResult	ShipmentActionResult	No		Success, Errors and Shipment ID Fields.
PackageResults	ArrayOfChargesPackageResult	No		Charges on package level.
Remarks	string	No		Charges Remarks.
ShipmentCharges	ArrayOfAmount	No		Charges on shipment level.
TotalCharges	Amount	No		Total Charges for Shipments.

Xml Representation

```
<ChargesResult>
  <ActionResult>...</ActionResult>
  <PackageResults>...</PackageResults>
  <Remarks>...</Remarks>
  <ShipmentCharges>...</ShipmentCharges>
  <TotalCharges>...</TotalCharges>
</ChargesResult>
```

ClosedResult Type

Closed Shipments Result Data.

Members

Member Name	Type	Required	MaxLength	Description
CEP	string	No	3	Closed result for this carrier
ClosedStamp	string	No	20	Stamp, consisting of DateTime (YYYYMMDDHHMMSS) and Carrier information, like 20150528135632PA1.
printDocumentResult	printDocumentsResult	No		Closing Report as Base64-encoded string.

Xml Representation

```
<ClosedResult>
  <CEP>...</CEP>
  <ClosedStamp>....</ClosedStamp>
  <printDocumentResult>...</printDocumentResult>
</ClosedResult>
```

Consigner Type

Consigner Information.

Members

Member Name	Type	Required	MaxLength	Description
ConsignerID	string	Yes	20	Consigner Abbreviation.
Default	int	No	0 or 1	Default Consigner (1=true).
Shortname	string	No	20	Consigner Short name.

Xml Representation

```
<Consigner>
  <ConsignerID>....</ConsignerID>
  <Default>....</Default>
  <Shortname>...</Shortname>
</Consigner>
```

Contact Type

Contact Request Data.

Members

Member Name	Type	Required	MaxLength	Description
AttentionName	string	No	50	Attention name.
BirthDate	string	No	Date	Birth Date, not used yet.
Company	string	No	50	Company Information.
ContactPerson	string	No	50	Contact Person
Email	string	No	40	Email.
Fax	string	No	40	Fax.
FirstName	string	No	30	Firstname.
LastName	string	No	30	Lastname.
Mobile	string	No	40	Mobile Phone Number.
Phone	string	No	40	Phone Number.
Salutation	string	No	40	not in use yet.

Xml Representation

```
<Contact>
  <AttentionName>...</AttentionName>
  <BirthDate>...</BirthDate>
  <Company>...</Company>
  <ContactPerson>...</ContactPerson>
  <Email>...</Email>
  <Fax>...</Fax>
  <FirstName>...</FirstName>
```

```
<LastName>...</LastName>
<Mobile>...</Mobile>
<Phone>...</Phone>
<Salutation>...</Salutation>
</Contact>
```

Customer Type

Customer Info.

Members

Member Name	Type	Required	MaxLength	Description
Consigners	ArrayOfConsigner	No		Available Consigners for this customer.
CustomerID	string	Yes	20	Customer Number at ParcelOne .
Default	int	No	0 or 1	Default Mandator (1=true).
MandatorID	string	No	20	Mandator ID
Shortname	string	No	20	Mandator Short name.
Status	string	No		ID 0 = Customer active, ID 1 = Customer blocked, ID 2 = Customer anulled.
StatusID	int	No		ID 0 = Customer active, ID 1 = Customer blocked, ID 2 = Customer anulled.

Xml Representation

```
<Customer>
  <Consigners>...</Consigners>
  <CustomerID>...</CustomerID>
  <Default>...</Default>
  <MandatorID>...</MandatorID>
  <Shortname>...</Shortname>
  <Status>...</Status>
  <StatusID>...</StatusID>
</Customer>
```

Dimensions Type

Dimensions Information.

Members

Member Name	Type	Required	MaxLength	Description
Height	string	No	Decimal (10,2).	Height of Package, NumberFormat InvariantInfo.
Length	string	No	Decimal (10,2)	Length of Package, NumberFormat InvariantInfo.
Measurement	string	No		Measurement, so far only cm for centimeter.
Width	string	No	Decimal (10,2)	Width of Package, NumberFormat InvariantInfo.

Xml Representation

```
<Dimensions>
  <Height>...</Height>
  <Length>...</Length>
  <Measurement>...</Measurement>
  <Width>...</Width>
</Dimensions>
```

Error Type

Error Identification.

Members

Member Name	Type	Required	MaxLength	Description
ErrorNo	string	No		Error No., usually 4 digits.
Message	string	No		Error Message.
StatusCode	string	No		Error Type, usually Hard or Soft.

Xml Representation

```
<Error>
  <ErrorNo>...</ErrorNo>
  <Message>...</Message>
  <StatusCode>...</StatusCode>
</Error>
```

Format Type

Format Information.

Members

Member Name	Type	Required	MaxLength	Description
Height	string	No		Height for Print Document, not in use yet.
Orientation	int	No	0 or 1	0=Portrait, 1=Landscape, not in use yet.
Size	string	No		Predefined size, like A4, A6, in use so far "CN22" for International Documents.
Type	string	No	10	Output format, like PDF or GIF.
Unit	string	No		Unit of Height and Width parameter, not in use yet.
Width	string	No		Output width, not in use yet.

Xml Representation

```
<Format>
  <Height>...</Height>
  <Orientation>...</Orientation>
  <Size>...</Size>
  <Type>...</Type>
  <Unit>...</Unit>
  <Width>...</Width>
</Format>
```

identifyPackage Type

Parcel Identification.

Members

Member Name	Type	Required	MaxLength	Description
PackageRefField	string	No		optional field of PackageReference, possible values: PackageID, PackageRef or PackageTrackingID.
PackageRefValue	string	No		PackageRefValue to transfer value of this field. If empty, only Shipment is identified.

Xml Representation

```
<identifyPackage>
  <PackageRefField>...</PackageRefField>
  <PackageRefValue>...</PackageRefValue>
</identifyPackage>
```

identifyShipment Type

Shipment Identification.

Members

Member Name	Type	Required	MaxLength	Description
ShipmentRefField	string	No		possible values: ShipmentID, ShipmentRef or TrackingID.
ShipmentRefValue	string	No		String representation of Value of this field.

Xml Representation

```
<identifyShipment>
  <ShipmentRefField>...</ShipmentRefField>
  <ShipmentRefValue>...</ShipmentRefValue>
</identifyShipment>
```

IntDoc Type

International Documents Request Data.

Members

Member Name	Type	Required	MaxLength	Description
Certificate	int	No	0 or 1	Check Certificate Box on CN23, 1= true.
CertificateNo	string	No	25	Certificate No to print on CN23.
Comments	string	No	100	Comments to print on CN23.
ConsignerCustomsID	string	No	20	Consigner's CustomsID to print on CN23, if not filled here, taken from Consigner Data in cbc-System.
ContentsDesc	ArrayOfIntDocContents	No		Contents lines for CN23.
Currency	string	No	3	3-character representation of Currency to use on CN23, default: EUR for Euro.
Date	string	No	Date	Date on CN23, if not filled today's date is used.
Explanation	string	No	100	Explanation to be printed on CN23.
InternationalDocumentFormat	Format	No		Output Format may be GIF or PDF, Size will always be A4.
Invoice	int	No	0 or 1	Check Invoice box on CN23.
InvoiceNo	string	No	20	Invoice No to print on CN23.
ItemCategory	int	No	1,2,3,4,5	Item Category, possible values:1-Gift, 2-Documents, 3-Commercial Sample, 4-Returned Goods, 5-Other.
License	int	No	0 or 1	Check License Box on CN23.
LicenseNo	string	No	25	License No to print on CN23.
NonDeliveryInstruction	string	No		not in use yet, for future use.
OfficeOfOrigin	string	No	100	Office of Origin to print on CN23.
Postage	string	No	Decimal (10,2)	Postage to include in CN23. NumberFormat: InvariantInfo
PrintInternationalDocuments	int	No	0 or 1	Print CN23 flag (1=true).
ServiceLevel	string	No		not in use yet, for future use.
ShipToRef	string	No	80	ShipTo Reference on CN23, if not filled, Email and phone of ShipTo address will be used.
TotalValue	string	No	Decimal (12,2)	Total Value to print on CN23, if not filled, Total will be calculated (Contents values plus Postage). NumberFormat: InvariantInfo
TotalWeightkg	string	No	Decimal (12,3)	Total Weight in kg on CN23, if not filled, will be calculated from Contents Weights. NumberFormat: InvariantInfo
ValidatedForExport	int	No	0 or 1	not in use yet, for future use (1=YES, 0=NO).

Xml Representation

```
<IntDoc>
  <Certificate>....</Certificate>
  <CertificateNo>...</CertificateNo>
  <Comments>...</Comments>
  <ConsignerCustomsID>...</ConsignerCustomsID>
  <ContentsDesc>...</ContentsDesc>
  <Currency>...</Currency>
  <Date>...</Date>
  <Explanation>...</Explanation>
  <InternationalDocumentFormat>...</InternationalDocumentFormat>
  <Invoice>...</Invoice>
  <InvoiceNo>...</InvoiceNo>
  <ItemCategory>...</ItemCategory>
  <License>...</License>
  <LicenseNo>...</LicenseNo>
  <NonDeliveryInstruction>...</NonDeliveryInstruction>
  <OfficeOfOrigin>...</OfficeOfOrigin>
  <Postage>...</Postage>
  <PrintInternationalDocuments>...</PrintInternationalDocuments>
  <ServiceLevel>...</ServiceLevel>
  <ShipToRef>...</ShipToRef>
  <TotalValue>...</TotalValue>
  <TotalWeightkg>...</TotalWeightkg>
  <ValidatedForExport>...</ValidatedForExport>
</IntDoc>
```

IntDocContents Type

International Documents Contents Request Data.

Members

Member Name	Type	Required	MaxLength	Description
Contents	string	No	60	Contents Description.
ItemValue	string	No	Decimal (12,2)	ItemValue, NumberFormat: InvariantInfo.
NetWeight	string	No	Decimal (12,3)	NetWeight of Item, NumberFormat: InvariantInfo.
Origin	string	No	40	if possible, Alpha-2-code Country abbreviation (ISO 3166-1).
Quantity	int	No	Int	Quantity of Item.
TariffNumber	string	No	30	TariffNumber of Item.

Xml Representation

```
<IntDocContents>
  <Contents>....</Contents>
  <ItemValue>...</ItemValue>
  <NetWeight>...</NetWeight>
  <Origin>...</Origin>
  <Quantity>...</Quantity>
  <TariffNumber>...</TariffNumber>
</IntDocContents>
```

IntDocData Type

International Documents Request Data.

Members

Member Name	Type	Required	MaxLength	Description
IDPackage	identifyPackage	No		to identify Package, if not filled, only Shipment will be identified.
IDShipment	identifyShipment	Yes		to identify Shipments to process.
InternationalDoc	IntDoc	No		International Documents Data to add to Shipments/Packages.

Xml Representation

```
<IntDocData>
  <IDPackage>...</IDPackage>
  <IDShipment>...</IDShipment>
  <InternationalDoc>...</InternationalDoc>
</IntDocData>
```

Measurement Type

Measurements Information.

Members

Member Name	Type	Required	MaxLength	Description
Unit	string	No		can be different
Value	string	No	Decimal	Value of Measurement. NumberFormat: InvariantInfo.

Xml Representation

```
<Measurement>
  <Unit>...</Unit>
  <Value>...</Value>
</Measurement>
```

printDocumentsResult Type

Shipment Documents Print Result Data.

Members

Member Name	Type	Required	MaxLength	Description
ActionResult	ShipmentActionResult	No		Success, Errors and Shipment ID Fields.
DocType	string	No	25	Description of Document returned.
Document	string	No		Document as Base64-encoded string.

Format	Format	No		to identify returned Format of Document.
PackageID	string	No	20	Parcel No.
PackageRef	string	No	20	Parcel Reference Field of Client.
PackageTrackingID	string	No	20	Parcel Tracking Number.

Xml Representation

```
<printDocumentsResult>
  <ActionResult>...</ActionResult>
  <DocType>...</DocType>
  <Document>...</Document>
  <Format>...</Format>
  <PackageID>...</PackageID>
  <PackageRef>...</PackageRef>
  <PackageTrackingID>...</PackageTrackingID>
</printDocumentsResult>
```

PrintRequest Type

Print Request Data.

Members

Member Name	Type	Required	MaxLength	Description
DocType	string	No	25	Identification of Document, not required yet.
Format	Format	No		to define Format to return, possible values, dependent on requested document: PDF or GIF.
IDPackage	identifyPackage	No		to identify Parcel, if not filled, only Shipment is identified.
IDShipment	identifyShipment	No		to identify Shipment to process.

Xml Representation

```
<PrintRequest>
  <DocType>...</DocType>
  <Format>...</Format>
  <IDPackage>...</IDPackage>
  <IDShipment>...</IDShipment>
</PrintRequest>
```

Product Type

Product Information.

Members

Member Name	Type	Required	MaxLength	Description
Default	int	No	0 or 1	Default Product for Carrier - (1=true).
ProductID	string	No	20	Product abbreviation, e.g. 11 for UPS-Standard, eco for PA1-Eco.
ProductName	string	No	20	Product Name.
Services	ArrayOfService	No		Available services for this carrier/product-combination.

Xml Representation

```
<Product>
  <Default>...</Default>
  <ProductID>...</ProductID>
  <ProductName>...</ProductName>
  <Services>...</Services>
</Product>
```

Profile Type

Profile for a mandator.

Members

Member Name	Type	Required	MaxLength	Description
CEPs	ArrayOfCEP	No		Available CEPs.
Consigners	ArrayOfConsigner	No		available Consigners for this Mandator.

Default	int	No	0 or 1	Default Mandator (1=true).
MandatorID	string	No	20	Mandator ID.
MandatorShortname	string	No	20	Mandator Short name.

Xml Representation

```
<Profile>
  <CEPs>...</CEPs>
  <Consigners>...</Consigners>
  <Default>...</Default>
  <MandatorID>...</MandatorID>
  <MandatorShortname>...</MandatorShortname>
</Profile>
```

Service Type

Services Information.

Members

Member Name	Type	Required	MaxLength	Description
AmountNeeded	int	No	0 or 1	Amout spec required (1=true) for this service, e.g. for COD or Insurance.
CurrencyNeeded	int	No	0 or 1	Currency required (1=true) for this service, 3-character-Abbreviaton of Currency, e.g. EUR for Euro..
Default	int	No	0 or 1 or 2	0=Optional (Service may be booked), 1=Standard (Service may be shown as a default choice), 2=Forced (Service will be added by the API anyway if not chosen already)
ServiceID	string	No	10	Service ID, e.g. NN for COD, SA for SaturdayDelivery, depends on available Services for Carrier.
ServiceName	string	No	30	Service Description, language-sensitive.

Xml Representation

```
<Service>
  <AmountNeeded>...</AmountNeeded>
  <CurrencyNeeded>...</CurrencyNeeded>
  <Default>...</Default>
  <ServiceID>...</ServiceID>
  <ServiceName>...</ServiceName>
</Service>
```

ShipFrom Type

ShipFrom / Consigner Data - only to be used with special agreement!

Members

Member Name	Type	Required	MaxLength	Description
CustomsID	string	No	20	Consigner's customsID.
Name1	string	No	50	First line on Shipment label.
Name2	string	No	50	2nd line on shipment label.
Name3	string	No	50	3rd line on shipment label
Reference	string	No	20	ShipFrom Reference by Client.
SalesTaxID	string	No	20	Sales TaxID of Consigner,not in use yet.
ShipmentAddress	Address	No		ShipFrom Address.
ShipmentContact	Contact	No		ShipFrom Contact Data, information like phone and/or email required for some carriers.

Xml Representation

```
<ShipFrom>
  <CustomsID>...</CustomsID>
  <Name1>...</Name1>
  <Name2>...</Name2>
  <Name3>...</Name3>
  <Reference>...</Reference>
  <SalesTaxID>...</SalesTaxID>
  <ShipmentAddress>...</ShipmentAddress>
  <ShipmentContact>...</ShipmentContact>
</ShipFrom>
```

Shipment Type

Shipment Request Data.

Members

Member Name	Type	Required	MaxLength	Description
CEPID	string	Yes	20	Carrier Abbreviation, e.g. UPS or PA1 for Parcel One.
CEPSpecials	ArrayOfCEPSpecial	No		for DHL so far: Key: PRINTONLYIFCODEABLE, Value: 1 - will return label only in case LEITCODE could be found to avoid Surcharge for not codable addresses.
ConsignerID	string	Yes	20	to identify Consigner to ship from.
CostCenter	string	No	20	for client use.
DocumentFormat	Format	No		specify Format for Documents to be returned, e.g. UPS HighValueReport, only value possible: PDF.
LabelFormat	Format	No		Label Format, only possible values of Type: GIF or PDF, Size-Parameter is not used yet.
MandatorID	string	Yes	20	Mandator ID for shipment.
MaxCharges	Amount	No		You can specify an amount of Charge as a maximum for the shipment (Amount of Zero = no action). In case of exaggerating the shipment will be declined with an error message. (So far charges are only returned for UPS.)
Other1	string	No	30	Additional Information of any kind.
Other2	string	No	30	Additional Information of any kind.
Other3	string	No	30	Additional Information of any kind.
Packages	ArrayOfShipmentPackage	Yes		Parcel Specifications.
PrintDocuments	int	No	0 or 1.	Return Documents, like UPS HighValueReport, with Shipment? Default=0=false.
PrintLabel	int	No	0 or 1	Return Label with shipment, default=0=false.
ProductID	string	Yes	20	Product for Carrier, e.g. 11 for UPS Standard, eco for Parcel One-eco.
ReturnCharges	int	No	0 or 1	to return the shipment charges (so far only works for UPS), default=0=false, 1 = true.
ReturnShipmentIndicator	int	No	0 or intvalue.	required - 0 = Forward Shipment, all values > 0 = Return Shipment. For UPS the following are available: 2-Print and Mail Return Label by UPS; 3-Return Service 1-Attempt; 5-Return Service 3-Attempt; 8-Electronic Return Label by URL; 9-Print Return Label.
Services	ArrayOfShipmentService	No		Services for Shipment.
ShipFromData	ShipFrom	No		ShipFrom-Data, to use only with special agreement with ParcelOne , usually ConsignerID is used.
ShipToData	ShipTo	Yes		ShipTo-Data.
ShipmentID	string	No	numeric (int64)	If shipping successfull, ShipmentID will be overwritten/assigned, has to be numeric.
ShipmentRef	string	No	20	Shipment Client reference, for later identification.
Software	string	Yes	20	Software and version used.

Xml Representation

```
<Shipment>
  <CEPID>...</CEPID>
  <CEPSpecials>....</CEPSpecials>
  <ConsignerID>....</ConsignerID>
  <CostCenter>....</CostCenter>
  <DocumentFormat>....</DocumentFormat>
  <LabelFormat>....</LabelFormat>
  <MandatorID>....</MandatorID>
  <MaxCharges>...</MaxCharges>
  <Other1>...</Other1>
  <Other2>...</Other2>
  <Other3>...</Other3>
  <Packages>...</Packages>
  <PrintDocuments>...</PrintDocuments>
  <PrintLabel>...</PrintLabel>
  <ProductID>...</ProductID>
  <ReturnCharges>...</ReturnCharges>
  <ReturnShipmentIndicator>...</ReturnShipmentIndicator>
  <Services>...</Services>
  <ShipFromData>...</ShipFromData>
  <ShipToData>...</ShipToData>
  <ShipmentID>...</ShipmentID>
  <ShipmentRef>...</ShipmentRef>
  <Software>...</Software>
</Shipment>
```

ShipmentActionResult Type

Shipment Action Results.

Members

Member Name	Type	Required	MaxLength	Description
Errors	ArrayOfError	No		Array of Error Information.
ShipmentID	string	No	int64	ShipmentID assigned with succesfull shippings.
ShipmentRef	string	No	20	Shipment Reference field provided by client for identification.
Success	int	No	-1,0,1	-1=Error, 0=Warnings, 1=Successfull.
TrackingID	string	No	20	Tracking ID provided by carrier.

Xml Representation

```
<ShipmentActionResult>
  <Errors>...</Errors>
  <ShipmentID>...</ShipmentID>
  <ShipmentRef>...</ShipmentRef>
  <Success>...</Success>
  <TrackingID>...</TrackingID>
</ShipmentActionResult>
```

ShipmentDocumentsResult Type

Shipment Documents Results Data.

Members

Member Name	Type	Required	MaxLength	Description
Charges	ArrayOfAmount	No		not in use yet.
DocType	string	No	25	DocumentType for description of document returned.
Document	string	No		Document returned as Base64-encoded string.
DocumentID	string	No		Document Identification returned, e.g. ParcelOne Retoure Tracknumber.
Format	Format	No		to return provided DocumentType (PDF,GIF), DocumentSize (A4,A6), etc.
PackageID	string	No	20	Parcel ID assigned at Shipping.
ShipmentID	string	No	Int64	ShipmentID assigned at Shipping.
TrackingID	string	No	20	TrackingID assigned by Carrier at Shipping.

Xml Representation

```
<ShipmentDocumentsResult>
  <Charges>...</Charges>
  <DocType>...</DocType>
  <Document>...</Document>
  <Format>...</Format>
  <PackageID>...</PackageID>
  <ShipmentID>...</ShipmentID>
  <TrackingID>...</TrackingID>
</ShipmentDocumentsResult>
```

ShipmentPackage Type

Parcel Information.

Members

Member Name	Type	Required	MaxLength	Description
IntDocData	IntDoc	No		International Documents Data for Parcel (for Shipment send with first parcel).
PackageDimensions	Dimensions	No		to specify Length, Width, Height, required for some carriers.
PackageID	string	No	20	will be reassigned at successfull shipping.
PackageRef	string	No	20	Client's reference to parcel.
PackageType	string	No	10	Letter, Parcel, ... - required for UPS - UPS: Package Type = "02" for "Package".
PackageVolume	Measurement	No		Volume for Parcel in liters, required for some carriers.
PackageWeight	Measurement	No		ParcelWeight in kg, required for most carriers.
Remarks	string	No	100	Client's comments.
Services	ArrayOfShipmentService	No		Specify Services at package level.

Xml Representation

```
<ShipmentPackage>
  <IntDocData>...</IntDocData>
```

```

<PackageDimensions>...</PackageDimensions>
<PackageID>...</PackageID>
<PackageRef>...</PackageRef>
<PackageType>...</PackageType>
<PackageVolume>...</PackageVolume>
<PackageWeight>...</PackageWeight>
<Remarks>...</Remarks>
<Services>...</Services>
</ShipmentPackage>

```

ShipmentPackageResult Type

Shipment Parcel Results Data.

Members

Member Name	Type	Required	MaxLength	Description
Charges	ArrayOfAmount	No		not in use yet.
DocType	string	No	25	will always be "Label".
Format	Format	No		to return provided Format (Type=GIF or PDF, Size=A4 or A6).
Label	string	No		64base-encoded string of Label.
PackageID	string	No	20	Parcel number assigned at successfull shipping.
PackageRef	string	No	20	Parcel reference provided by client for identification.
ShipmentID	string	No	int64	Shipment ID assigned at succesfull shipping.
TrackingID	string	No	20	Tracking ID provided by carrier at successfull shipping.
TrackingURL	string	No	120	TrackingURL of respective Carrier.

Xml Representation

```

<ShipmentPackageResult>
  <Charges>...</Charges>
  <DocType>...</DocType>
  <Format>...</Format>
  <Label>...</Label>
  <PackageID>...</PackageID>
  <PackageRef>...</PackageRef>
  <ShipmentID>...</ShipmentID>
  <TrackingID>...</TrackingID>
</ShipmentPackageResult>

```

ShipmentResult Type

Shipment Results Data.

Members

Member Name	Type	Required	MaxLength	Description
ActionResult	ShipmentActionResult	No		Shipment Identification fields, Errors, Success.
Charges	ArrayOfAmount	No		Charges Information if ReturnCharges has been selected.
DocumentsAvailable	int	No		Number of available Documents for this shipment.
DocumentsResults	ArrayOfShipmentDocumentsResult	No		Documents returned if specified, e.g. UPS HighValueReport.
InternationalDocumentsAvailable	int	No		Number of International Documents available for this shipment.
InternationalDocumentsNeeded	int	No	1	Values: -1 = has not been checked (with ParcelOne will be always checked), 0 = not required, 1= required. This information is returned independently of whether InternationalDocuments-Data have been provided or not!
InternationalDocumentsResults	ArrayOfShipmentDocumentsResult	No		CN23 returned if specified.
LabelsAvailable	int	No		Number of labels available for this shipment.
LabelURL	string	No	200	LabelURL, so far only used for UPS Return Shipments with ReturnShipmentIndicator "8".
PackageResults	ArrayOfShipmentPackageResult	No		Parcel data returned.
TotalCharges	Amount	No		Total Charges Information if ReturnCharges has been selected.

Xml Representation

```

<ShipmentResult>
  <ActionResult>...</ActionResult>
  <Charges>...</Charges>
  <DocumentsAvailable>...</DocumentsAvailable>
  <DocumentsResults>...</DocumentsResults>

```

```

<InternationalDocumentsAvailable>...</InternationalDocumentsAvailable>
<InternationalDocumentsNeeded>...</InternationalDocumentsNeeded>
<InternationalDocumentsResults>...</InternationalDocumentsResults>
<LabelURL>...</LabelURL>
<LabelsAvailable>...</LabelsAvailable>
<PackageResults>...</PackageResults>
<TotalCharges>...</TotalCharges>
</ShipmentResult>

```

ShipmentService Type

Services Information.

Members

Member Name	Type	Required	MaxLength	Description
Parameters	string	No	100	not in use yet.
ServiceID	string	No	20	Service Abbreviation, depending on Carrier/Product, e.g. NN for COD, SA for SaturdayDelivery.
Value	Amount	No		Amount and Currency required for some Services like COD and Insurance.

Xml Representation

```

<ShipmentService>
  <Parameters>...</Parameters>
  <ServiceID>...</ServiceID>
  <Value>...</Value>
</ShipmentService>

```

ShipmentStatusResult Type

Shipment Status Results.

Members

Member Name	Type	Required	MaxLength	Description
ActionResult	ShipmentActionResult	No		Shipment Identification fields, Errors, Success.
ShipmentStatus	string	No	20	Shipment Status, Code 10 = Shipped; Code 90 = Closed, Status will be returned according to client language provided.
ShipmentStatusCode	string	No		Code 10 = Shipped; Code 90 = Closed, Status will be returned according to client language provided.
LastTrackingNo	string	No		Tracking Code (often USPS-No.) of last known Carrier nearest to destination carrier.
LastCEP	string	No		Last known Carrier nearest to destination carrier..

Xml Representation

```

<ShipmentStatusResult>
  <ActionResult>...</ActionResult>
  <ShipmentStatus>...</ShipmentStatus>
  <ShipmentStatusCode>...</ShipmentStatusCode>
  <LastTrackingNo>...</LastTrackingNo>
  <LastCEP>...</LastCEP>
</ShipmentStatusResult>

```

ShipmentTrackingResult Type

Shipment Trackings Results.

Members

Member Name	Type	Required	MaxLength	Description
ActionResult	ShipmentActionResult	No		Shipment Identification fields, Errors, Success.
Trackings	ArrayOfTrackingResult	No		Array of Tracking Data (DateTime, StatusCode, Status) for every Shipment inquired.

Xml Representation

```

<ShipmentTrackingResult>
  <ActionResult>...</ActionResult>
  <Trackings>...</Trackings>
</ShipmentTrackingResult>

```

ShipTo Type

Shipto / Receiver Data.

Members

Member Name	Type	Required	MaxLength	Description
BranchID	string	No	20	Branch or Pickup ID of Carrier. If PUDO-Shipping, then PUDO-Number must be provided here.
CEPCustID	string	No	20	Carrier Customer ID of Receiver, e.g. UPS Customer No. for cheaper Shipping. Not in use yet.
CustomsID	string	No	20	Receiver's customsID.
Name1	string	Yes	50	First line on shipment label.
Name2	string	No	50	2nd line on shipment label.
Name3	string	No	50	3rd line on shipment label.
PrivateAddressIndicator	int	Yes	0 or 1	Private Address of Receiver? (1=true) - B2B or B2C.
Reference	string	No	20	ShipTo Reference provided by client.
SalesTaxID	string	No	20	Sales TaxID of Receiver, not in use yet.
ShipmentAddress	Address	Yes		ShipTo Address.
ShipmentContact	Contact	No		ShipTo Contact Data, information like phone and/or email required for some carriers.

Xml Representation

```
<ShipTo>
  <BranchID>...</BranchID>
  <CEPCustID>...</CEPCustID>
  <CustomsID>...</CustomsID>
  <Name1>...</Name1>
  <Name2>...</Name2>
  <Name3>...</Name3>
  <PrivateAddressIndicator>...</PrivateAddressIndicator>
  <Reference>...</Reference>
  <SalesTaxID>...</SalesTaxID>
  <ShipmentAddress>...</ShipmentAddress>
  <ShipmentContact>...</ShipmentContact>
</ShipTo>
```

TrackingResult Type

Shipment Tracking Result Data.

Members

Member Name	Type	Required	MaxLength	Description
TrackingDateTime	string	No		Tracking Event DateTime.
TrackingStatus	string	No	Char unlimited	Tracking Event Status Description.
TrackingStatusCode	string	No	20	Tracking Event Status Code Number.

Xml Representation

```
<TrackingResult>
  <TrackingDateTime>...</TrackingDateTime>
  <TrackingStatus>...</TrackingStatus>
  <TrackingStatusCode>...</TrackingStatusCode>
</TrackingResult>
```

ArrayOfstring Array

Array Element

Element Name	Type
string	string

Xml Representation

```
<ArrayOfstring>
  <string>...</string>
  <string>...</string>
```

1/3/26, 8:29 AM

ShippingWCF API - ParcelOne

```
<string>...</string>
...
</ArrayOfstring>
```
